Sicherheit und Privatsphäre in Online Sozialen Netzwerken

Michael Dürr

Dissertation
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig–Maximilians–Universität München

vorgelegt von
Michael Dürr

Tag der Einreichung: 15. Oktober 2013

Sicherheit und Privatsphäre in Online Sozialen Netzwerken

Michael Dürr

Dissertation
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig–Maximilians–Universität München

vorgelegt von
Michael Dürr

1. Berichterstatter:	Prof. Dr. Claudia Linnhoff-Popien
2. Berichterstatter:	Prof. Dr. Stefan Fischer
Tag der Einreichung:	15. Oktober 2013
Tag der Disputation:	22. November 2013

Eidesstattliche Versicherung

(siehe Promotionsordnung vom 12.07.11, § 8, Abs. 2 Pkt. 5)

Hiermit erkläre ich an Eides statt, dass die Dissertation von
mir selbstständig, ohne unerlaubte Beihilfe angefertigt ist.

Michael Dürr

Danksagung

Mein ganz besonderer Dank gilt Frau Prof. Dr. Claudia Linnhoff-Popien, die mir durch ihre hervorragende und verständnisvolle Betreuung die Erstellung der vorliegenden Dissertation überhaupt erst ermöglichte. Herrn Prof. Dr. Stefan Fischer danke ich für die Übernahme des Koreferats und Herrn Prof. Dr. Ohlbach für sein Mitwirken als Vorsitzender der Prüfungskommission.
Weiter möchte ich mich bei allen Kollegen am Lehrstuhl für Mobile und Verteilte Systeme für unzählige anregende Diskussionen und Gespräche bedanken. Ein besonderer Dank gilt dabei Florian Dorfmeister, Markus Duchon, Florian Gschwandtner, Robert Lasowski, Marco Maier, Valentin Protschky, Kim Schindhelm, und Kevin Wiesner, die mir tatkräftig bei der Lösung von Fragestellungen und in zahlreichen inhaltlichen Diskussionen zur Seite standen.
Besonders bedanken möchte ich mich bei Nicole Comtesse, Claudia und Manuel Friebe und Steffen Lehmann, die immer ein offenes Ohr für meine Sorgen hatten und es mir nie nachgetragen haben, dass ich mich in den letzten Jahren nur sehr eingeschränkt um unsere gemeinsame Freundschaft gekümmert habe.
Sehr herzlich bedanke ich mich bei Dario Haselwarter, Christoph Hummel und Tobias Karpinski die mir mit zahlreichen gemeinsamen Aktionen in den Bergen dieser Welt, das Promovieren versüßten.
Mein größter Dank gebührt meiner Familie, die mich in guten wie in schlechten Zeiten mit ihren aufmunternden Worten und ihrer unendlichen Liebe unterstützt hat.

Zusammenfassung

Online Soziale Netzwerke (OSNs) repräsentieren das vorherrschende Medium zur computergestützten Kommunikation und Verbreitung persönlicher, geschäftlicher oder auch wissenschaftlicher Inhalte. Eine Reihe von Vorkommnissen in der jüngsten Vergangenheit hat gezeigt, dass die Bereitstellung privater Informationen in OSNs mit erheblichen Risiken für die Sicherheit und den Schutz der Privatsphäre seiner Nutzer verbunden ist. Gleiches gilt für die Bereiche Wirtschaft und Wissenschaft. Ursächlich dafür ist die zentralisierte Verwaltung der Nutzer und ihrer publizierten Inhalte unter einer singulären administrativen Domäne.
Mit Vegas präsentiert der erste Teil dieser Arbeit ein dezentrales OSN, das mit seiner restriktiven Sicherheitsarchitektur diesem Problem begegnet. Oberstes Ziel ist die technische Umsetzung des Rechts auf informationelle Selbstbestimmung. Dazu schränkt Vegas den Zugriff auf den sozialen Graphen und jeglichen Informationsaustausch auf die Nutzer des eigenen Egonetzwerks ein.
Neben der Möglichkeit zur Kommunikation und der Bereitstellung persönlicher Informationen erlauben einige OSNs auch das Browsen des sozialen Graphen und die Suche nach Inhalten anderer Nutzer. Um auch in sicheren und die Privatsphäre schützenden OSNs wie Vegas vom akkumulierten Wissen des sozialen Graphen zu profitieren, beschäftigt sich der zweite Teil dieser Arbeit mit der Entwicklung und Analyse intelligenter Priorisierungsstrategien zur Weiterleitung von Suchanfragen innerhalb dezentraler OSNs.
Im Kontext von OSNs werden neue Algorithmen und Protokolle zunächst simulativ evaluiert. Die Grundlage bildet in der Regel der Crawling-Datensatz eines OSNs. Offensichtlich ist das Crawling in sicheren und die Privatsphäre schützenden dezentralen OSNs wie Vegas nicht möglich. Um diesem Problem zu begegnen, beschäftigt sich der dritte Teil dieser Arbeit mit der Entwicklung eines generischen Modells zur künstlichen Erzeugung sozialer Interaktionsgraphen. Neben den strukturellen Besonderheiten zentralisierter und dezentraler Systeme wird erstmals auch das Interaktionsverhalten der Nutzer eines OSNs modelliert. Die Eignung des Modells wird auf der Grundlage gecrawlter sozialer Graphen evaluiert.

Abstract

Online Social Networks (OSNs) represent the dominating media for computer-aided communication and the distribution of personal, commercial, and scientific content. Recently a series of incidents has shown that, for its users, the provision of private information in an OSN can create considerable security and privacy risks. The same statement holds for the commercial and the scientific domain. The problem arises from a centralized organization of users and their published contents and its management through a single administrative domain.

To overcome this problem, the first part of this thesis introduces Vegas, a decentralized OSN which is based on a highly restrictive security architecture. The major goal of Vegas is to provide a technical implementation of the right for informational self-determination. Therefore Vegas restricts access to the social graph and the exchange of information to users of the own ego-network.

In addition to the possibility to communicate and to provide personal data, several OSNs allow for browsing the social graph and for searching content of other users. To benefit from the accumulated knowledge of the social graph in secure and privacy-preserving OSNs like Vegas, the second part of this thesis addresses the development and the analysis of intelligent prioritization strategies for query forwarding in decentralized OSNs.

In context of OSNs, the evaluation of new algorithms and protocols takes place through simulation which is based on crawling data of an OSN. Obviously crawling secure and privacy-preserving OSNs like Vegas is not possible. Therefore the third part of this thesis presents a generic model to synthesize social interaction graphs. Besides structural characteristics of centralized and decentralized OSNs, the model also considers the interaction behavior of its users. Its applicability is evaluated on the basis of social graph crawling data.

Inhaltsverzeichnis

1 Einleitung

Spätestens seit dem Einzug *sozialer Medien* (engl: *social media*) [110] im *World Wide Web* (WWW) vollzieht sich ein kontinuierlicher Wandel bei der Verbreitung und Darstellung persönlicher Informationen. Erfolgten vor der Jahrtausendwende Veröffentlichungen noch ausschließlich über statische Webseiten, Newsgroups oder Online-Foren, so beobachtet man gegenwärtig eine klare Dominanz der Publikation in *Online Sozialen Netzwerken* (*OSNs*, engl: *online social networks*). Laut Alexa Internet Inc. [233] zählen OSNs wie Facebook [246], LinkedIn [263] oder Twitter [276] neben Suchmaschinenriesen wie Google [256] und Yahoo [283] derzeit zu den 15 am stärksten frequentierten Angeboten im WWW.

Entgegen der rein inhaltsbezogenen Bereitstellung von Informationen forcieren OSNs die *ichbezogene* bzw. *egozentrische* Publikation persönlicher Inhalte. Das Individuum steht im Mittelpunkt und beeinflusst maßgeblich, wie schnell und zu welchem Grad sich neue Informationen über das WWW verbreiten [16].

OSNs unterstützen die egozentrische Darstellung durch eigene Profilen und durch Verknüpfungen in enger Beziehung zueinander stehender Profile. Außerdem bieten sie die Möglichkeit zur profilgetriebenen Recherche entlang dieser Verknüpfungen [34]. Stellt man die Gesamtheit der Nutzer und ihrer Beziehungen untereinander als Graphen dar, spricht man auch vom *sozialen Graphen* eines OSNs.

Die individuellen, gemeinschaftlichen und geschäftlichen Vorteile von OSNs sind vielfältig: sie bieten eine aktuelle Liste an Kontakten, eine Vielzahl an Möglichkeiten zur Kommunikation, Funktionen zum Publizieren, Teilen, Bewerten, Empfehlen und Suchen von Inhalten sowie zahlreiche Gelegenheiten zur Zusammenarbeit [34]. Der enorme Informationsbestand eines OSNs ermöglicht die Entwicklung und Evaluation einer Vielfalt innovativer Anwendungen für den privaten, den geschäftlichen, aber auch den wissenschaftliche Gebrauch [116, 92, 182].

Leider ist die Bereitstellung privater Informationen in OSNs auch mit erheblichen Risiken für die Sicherheit und den Schutz der Privatsphäre der Nutzer verbunden. Zahlreiche Publikationen berichten über die negativen Auswirkungen einer ungewollten Weitergabe und Auswertung solcher Informationen. Neben meist als lästig empfundenen Entwicklungen wie dem Versand personalisierter Werbung und kontextsensitiven Spams [36, 100] werden persönliche Informationen vermehrt auch für gezielte Mobbing-Angriffe im Internet (engl: *cyber mobbing*) missbraucht [181, 265].

Daneben existieren Sicherheitsattacken zur Offenlegung persönlicher Daten [120, 17, 103, 231], zur Deanonymisierung von Nutzern [218, 157, 12, 156, 173, 30], zum Diebstahl von Identitäten [138, 33, 32, 27] sowie zur Übernahme existierender Nutzerkonten [98, 102, 174, 99, 188, 107]. Derartige Angriffe können zu er-

heblichen persönlichen Nachteilen und zu einer schwerwiegenden Schädigung der betroffenen Personen führen [243, 264, 274].

Derzeit handelt es sich bei der Sicherheit persönlicher Informationen und dem Schutz der Privatsphäre um die dominierenden Themen bei der Entwicklung und der Analyse von OSNs. Als Kernproblem lässt sich die zentralisierte Verwaltung der Nutzerdaten unter einer singulären administrativen Domäne identifizieren. Das globale Wissen über alle Nutzer, deren Eigenschaften und ihre Vernetzung untereinander erlaubt einen uneingeschränkten Zugriff auf den gesamten sozialen Graphen. Die Auswertung und die Analyse des sozialen Graphen ermöglichen zahlreiche Sicherheitsangriffe auf individuelle Nutzer.

Es existieren bereits einige Vorschläge, die Sicherheit und den Schutz der Privatsphäre der Nutzer in OSNs zu verbessern. Der eine Teil beschäftigt sich mit der nachträglichen Integration zusätzlicher Sicherheitsmechanismen [137, 89, 199] in bereits etablierten OSNs. Diese Lösungen erfordern zwar keinerlei Migration zu einem anderen OSN. Die persönlichen Inhalte befinden sich jedoch weiterhin unter einer einzelnen administrativen Domäne. Beim anderen Teil handelt es sich jeweils um eine dezentrale Neuentwicklungen eines OSNs [189, 38, 55, 242, 86, 158]. Die meisten Systeme verwenden zur Organisation und Kommunikation ein P2P-Overlay. In ihrer Sicherheitsarchitektur unterscheiden sich die Ansätze jedoch stark.

Mit einer dezentralen Organisation begegnet jedes dieser Systeme dem Problem einer singulären administrativen Domäne. Leider bietet jedoch keines dieser Systeme neben der notwendigen Sicherheit persönlicher Inhalte auch einen ausreichenden Schutz der Privatsphäre. Gerade das Recht auf Anonymität der Nutzer rückt bei vielen Konzepten komplett in den Hintergrund.

Mit *Vegas* [292] präsentiert der erste Teil dieser Arbeit ein dezentrales OSN, das mit seiner restriktiven Sicherheitsarchitektur diese Lücke füllt. Oberstes Ziel ist die technische Umsetzung des Rechts auf *informationelle Selbstbestimmung*. Gemeint ist das Recht jedes einzelnen Nutzers, jederzeit selbst über die Veröffentlichung und Verwendung seiner privaten und persönlichen Daten zu bestimmen [54, 141]. Für eine adäquate Umsetzung schränkt Vegas den Zugriff auf den sozialen Graphen und jeglichen Informationsaustausch auf die Nutzer des eigenen Egonetzwerks ein. Für den Zugriff auf persönliche Informationen sind die Existenz einer direkten Beziehung und die selektive Freigabe der entsprechenden Inhalte zwingend erforderlich. Weder eine personenbezogene, noch eine globale Aggregation persönlicher Informationen sind in Vegas durchführbar.

Neben der Möglichkeit zur Kommunikation und der Bereitstellung persönlicher Informationen erlauben einige etablierte OSNs auch das Browsen sowie die Suche nach persönlichen Informationen und Inhalte beliebiger anderer Nutzer. Aufgrund der Einbeziehung sozialer Strukturen spricht man in diesem Kontext auch von *sozialer Suche* [7, 42].

Das Browsen steht offensichtlich im Widerspruch zur Zielsetzung dezentraler OSNs. Bleibt das Recht auf informationelle Selbstbestimmung gewahrt, stellt die Möglichkeit zur sozialen Suche auch für dezentrale OSNs eine interessante Funktionalität dar.

Die meisten dezentralen OSNs verwenden als Overlay ein strukturiertes P2P-Netzwerk [39, 159, 189, 56]. Die soziale Suche lässt sich direkt über das Routing-Verfahren solcher Overlays realisieren. Vegas hingegen zählt zu den unstrukturierten P2P-Netzwerken. Seine restriktive Sicherheitsarchitektur verbietet die aktive Einflussnahme auf die Topologie des Overlays bzw. des zugrunde liegenden sozialen Graphen. Die Entscheidung zur Weiterleitung einer sozialen Suchanfrage ist auf die Mitglieder des Egonetzwerks beschränkt. Für die Umsetzung einer sozialen Suche in Vegas scheidet die Mehrzahl entsprechender Routing-Verfahren [52, 192, 209, 48, 71, 9] daher aus.

Motiviert durch diese Einschränkung stellt die Entwicklung und Analyse intelligenter Strategien zur Priorisierung der Mitglieder des Egonetzwerks den zweiten Schwerpunkt dieser Arbeit dar. Im Vordergrund stehen Strategien auf der Basis *sozialer Zentralitätsmaße* [75, 143]. Diese dienen als Richtwert für den Einfluss eines Nutzers innerhalb eines sozialen Netzwerks.

Neben egozentrischen werden auch die entsprechenden *soziozentrischen* Zentralitätsmaße als Kandidaten für die Priorisierung von Nutzern untersucht. Während erstere ausschließlich auf dem Wissen des eigenen Egonetzwerks beruhen, berücksichtigen letztere das gesamte soziale Netzwerk. Die Resultate für soziozentrische Zentralitätsmaße dienen in erster Linie als Vergleichswert für die Ergebnisse der egozentrischen Analyse.

Zudem werden *erweiterte* egozentrische Zentralitätsmaße untersucht. Die Erweiterung besteht in der Einbeziehung von Informationen über Freunde *zweiter* (Freundesfreunde), bzw. *dritter Ordnung*. Dabei gilt es herauszufinden, ob und in welchem Ausmaß dieses zusätzliche Wissen eine Priorisierungsstrategie verbessern kann. Unter Annahme weniger restriktiver Anforderungen an den Schutz der Privatsphäre können solche Strategien durchaus gerechtfertigt sein.

Für Computernetzwerke erfolgt die Untersuchung neuer Algorithmen, Protokolle und Anwendungen im ersten Schritt meist in Form einer Simulation. Im Falle von OSNs dient als Simulationsgrundlage das Abbild seines sozialen Graphen. Die vorherrschende Methode zu dessen Herleitung besteht im *Crawling* [224]: Roboterprogramme (engl: *crawler*) bewegen sich nach einem vorgegebenen oder zufälligen Muster entlang der Verbindungen zwischen den Nutzern eines OSNs. Die dabei gewonnenen Informationen über die Nutzer und deren Verknüpfungen beschreiben schließlich den sozialen Graphen.

Für die Analyse der betrachteten Priorisierungsstrategien zur Weiterleitung von Suchanfragen ergibt sich hier ein entscheidendes Problem: Aufgrund der restriktiven Sicherheitsvorkehrungen und der P2P-Charakteristik von Vegas ist ein Crawling generell nicht möglich. Offensichtlich trifft dieses Problem auf die Mehrzahl dezentraler OSNs zu.

Eine alternative Vorgehensweise besteht in der Verwendung sozialer Graphen zentralisierter OSNs als Ersatz für die Simulation dezentraler Systeme. Dieser Ansatz findet im Rahmen dieser Arbeit bei der Analyse der Priorisierungsstrategien Anwendung.

Die Intention dezentraler OSNs lässt jedoch vermuten, dass sie sich in ihren struktu-

rellen Eigenschaften maßgeblich von zentralisierten Systemen unterscheiden. Das Sicherheitskonzept von Vegas erfordert z.B. eine starke Vertrauensbeziehung als Grundlage der Ausbildung neuer Kontakte. Der damit einhergehende Wegfall unzähliger flüchtiger Kontakte hat massiven Einfluss auf die gesamte Graphenstruktur. Da Effektivität und Effizienz eines Algorithmus maßgeblich von der Struktur des sozialen Graphen abhängen, sind Crawling-Daten zentralisierter OSNs nur bedingt als Ersatz für dezentrale Systeme geeignet.

Unabhängig von den strukturellen Unterschieden zentralisierter und dezentraler OSNs ergeben sich weitere Probleme durch die Verwendung gecrawlter sozialer Graphen. Beim Crawling handelt es sich in der Regel um einen sehr aufwendigen Prozess, der sich über mehrere Wochen erstrecken kann. Daher stellt das Resultat stets eine zeitlich verzerrte Momentaufnahme des sozialen Graphen dar. Die statistische Konfidenz der darauf basierenden Ergebnisse ist somit nicht gewährleistet [185].

Aus Angst vor der Deanonymisierung [157, 218, 173] einzelner Personen existiert nur eine geringe Anzahl publizierter und ausreichend anonymisierter Crawling-Datensätze [150, 217, 82, 200]. Die Mehrzahl reflektiert lediglich die strukturellen Eigenschaften sozialer Graphen. Für die Analyse neuer Algorithmen, Anwendungen und Protokolle spielen jedoch auch *interaktive* Eigenschaften eine wichtige Rolle. Gemeint sind zeitlich beschränkte Ereignisse zwischen den Knoten eines sozialen Graphen, die in irgendeiner Form eine Interaktion zwischen den Mitgliedern des entsprechenden OSNs beschreiben. In den meisten OSNs existiert beispielsweise eine Vielzahl an Beziehungen, die nahezu keine Kommunikation oder jegliche andere Art aktiver Interaktionen aufweisen. Diese Tatsache hat unter Umständen maßgeblichen Einfluss auf das Verhalten eingesetzter Algorithmen, Anwendungen und Protokolle.

Als Alternative zum Crawling zentralisierter OSNs bietet sich die künstliche Erzeugung sozialer Graphen an. Zum einen entfallen damit jegliche Bedenken bezüglich einer Deanonymisierung einzelner Nutzer, zum anderen stellt die Verwendung mehrerer künstlich erzeugter Graphen die statistische Konfidenz der gewonnenen Ergebnisse sicher.

Gegenwärtig existieren zahlreiche Vorschläge zur Modellierung komplexer Netzwerke [213, 19, 20, 21], die auch einige Eigenschaften sozialer Netzwerke reflektieren [164]. Dennoch eignen sich die einzelnen Modelle nur bedingt zur Erzeugung sozialer Graphen von OSNs. Zum einen beschränken sie sich auf rein strukturelle Eigenschaften. Sie bieten somit keine Möglichkeit, die Gewichtung einzelner Beziehungen entsprechend der Interaktionen eines Nutzers adäquat zu modellieren. Zum anderen mangelt es ihnen an der Fähigkeit, auf einzelne Eigenschaften gezielt Einfluss zu nehmen, um spezifische Charakteristika dezentraler OSNs zu modellieren.

Motiviert durch die Probleme beim Crawling und die Erzeugung sozialer Graphen liefert der dritte Teil dieser Arbeit ein generisches Modell zur Erzeugung sozialer *Interaktionsgraphen*. Dieses berücksichtigt sowohl statische als auch interakti-

ve Besonderheiten zentralisierter und dezentraler OSNs. Die Eignung des Modells wird auf der Grundlage gecrawlter sozialer Graphen evaluiert.
Die vorliegende Arbeit ist folgendermaßen strukturiert: Kapitel 2 beschäftigt sich mit den Grundlagen, Funktionen, Definitionen und Begrifflichkeiten im Zusammenhang mit sozialen Netzwerken. Zudem liefert es einen Überblick zu P2P-Systemen sowie die im Kontext dieser Arbeit relevanten graphen- und netzwerktheoretischen Definitionen. Nach der detaillierten Betrachtung der Anforderungen an ein sicheres und die Privatsphäre schützendes OSN werden in Kapitel 3 bereits existierende dezentrale Systeme auf die Erfüllung aufzustellender Anforderungen hin untersucht. Kern des Kapitels stellt die Konzeption und Evaluation des dezentralen OSNs Vegas dar. Kapitel 4 beschäftigt sich mit der Informationsverbreitung in OSNs. Der Aufstellung von Anforderung an die Umsetzung der sozialen Suche folgt eine detaillierte Betrachtung möglicher Routing-Verfahren für dezentrale OSNs. Die Konzeption unterschiedlicher Priorisierungsstrategien für das Weiterleiten von Suchanfragen stellt den Schwerpunkt dieses Kapitels dar. Alle Strategien werden auf der Basis unterschiedlicher künstlich erzeugter und gecrawlter sozialer Graphen evaluiert. Kapitel 5 geht auf die Anforderungen an ein Modell zur künstlichen Erzeugung sozialer Graphen ein. Nach einer ausführlichen Betrachtung existierender Ansätze erfolgt die Konzeption eines generischen Modells. Dieses wird auf seine Eignung zur Modellierung existierender sozialer Graphen untersucht. Abschließend liefert Kapitel 6 eine Zusammenfassung der Ergebnisse der Arbeit und präsentiert offene Fragestellungen für zukünftige Forschungsarbeiten.

2 Grundlagen Online Sozialer Netzwerke

Dieses Kapitel beschäftigt sich mit den Grundlagen, Funktionen, Definitionen und Begriffen im Zusammenhang mit sozialen Netzwerken. Neben der Einordnung sozialer Netzwerke sowie der detaillierten Definition und Charakterisierung von OSNs erfolgen ein Überblick und eine Klassifikation von P2P-Systemen. Sie dienen als Grundlage für die Umsetzung eines dezentralen OSNs.
Zudem werden die wichtigsten graphen- und netzwerktheoretischen Definitionen und Konzepte vorgestellt, die im gesamten Verlauf dieser Arbeit von Bedeutung sind.
Zunächst wird die Bedeutung der Begriffe Sicherheit und Privatsphäre im Kontext dieser Arbeit erläutert.

2.1 Sicherheit und Privatsphäre

Im Umgang mit OSNs spielen die *Sicherheit* persönlicher Inhalte und der Schutz der eigenen *Privatsphäre* eine zentrale Rolle. Im Rahmen dieser Arbeit werden mit dem Begriff der Sicherheit sowohl die *Informationssicherheit* als auch der *Datenschutz* in OSNs assoziiert.
Die Informationssicherheit umfasst den Schutz von Daten gegen Beschädigung, Löschung oder Veränderungen durch eigene Eingriffe, aber auch durch das Einwirken anderer. Im Hinblick auf OSNs geht es hier insbesondere um die Gewährleistung des authentifizierten und autorisierten Zugriffs auf persönliche Daten und die Sicherstellung ihrer Integrität.
Unter *Datenschutz* versteht man den Schutz persönlicher Daten vor Missbrauch. Gemeint ist der Schutz vor der Erfassung, der Kenntnisnahme und der Verarbeitung personenbezogener Daten durch Unbefugte. Datenschutz ist eng verknüpft mit dem Schutz des Rechts auf *informationelle Selbstbestimmung*. Es handelt sich um das Recht jedes Einzelnen, persönlich über die Preisgabe und die Verwendung der eigenen personenbezogenen Daten zu bestimmen [141].
Um Datenschutz bzw. den Schutz des Rechts auf informationelle Selbstbestimmung zu gewährleisten, muss ein OSN den Schutz der Privatsphäre seiner Nutzer sicherstellen. Allgemein versteht man unter Privatsphäre einen nichtöffentlichen Bereich, in dem ein Mensch unbehelligt von äußeren Einflüssen sein Recht auf freie Entfaltung der Persönlichkeit wahrnehmen kann [54]. Für den Schutz der Privatsphäre müssen technische Maßnahmen eingesetzt werden, die den Zugriff auf persönliche Inhalte individuell regulieren und die Anonymität der Nutzer sicherstellen. Ge-

schützt werden müssen personenbezogene Daten wie Name, Alter und Geschlecht, welche die eigene Identität charakterisieren, aber auch jegliche andere Art persönlicher Inhalte, wie private Fotos, Kommentare zu persönlichen Erfahrungen oder sexuellen Vorlieben, Bewertungen von Produkten und Dienstleistungen oder individuelle Meinungen über andere Personen.

2.2 Differenzierung sozialer Netzwerke

Gegenwärtig verwendet die breite Masse der Bevölkerung den Begriff *soziales Netzwerk* als Synonym für OSNs. Vom wissenschaftlichen Standpunkt aus gesehen repräsentieren OSNs als virtuelle bzw. digitale Vertreter jedoch nur eine spezielle Ausprägung sozialer Netzwerke.
Neben der Informatik finden soziale Netzwerke in einer Vielzahl anderer wissenschaftlicher Disziplinen Anwendung. Auch in der Soziologie, der Psychologie, der Statistik oder der Betriebswirtschaftslehre dienen sie zur Modellierung und Analyse lokaler und globaler Muster, zur Lokalisierung einflussreicher Akteure und zur Auswertung der Dynamik von Netzwerken.
Allgemein versteht man unter einem sozialen Netzwerk eine Menge von Akteuren und den zwischen ihnen definierten Beziehungen [211]. Um den Begriff zu konkretisieren, wird im Folgenden zwischen realen und virtuellen sozialen Netzwerken unterschieden. OSNs stellen dabei eine besondere Form virtueller sozialer Netzwerke dar. Ist die exakte Klassifizierung für einen bestimmten Aspekt nicht relevant oder betrifft eine Aussage sowohl reale als auch virtuelle soziale Netzwerke, wird auch weiterhin der allgemeine Begriff des sozialen Netzwerks verwendet.

Reale soziale Netzwerke In den Sozialwissenschaften bezeichnet der Begriff soziales Netzwerk ein Geflecht von Beziehungen, welches einzelne Individuen miteinander verbindet [211]. Beispiele für solche Beziehungen existieren zwischen Mitgliedern der eigenen Familie und der Verwandtschaft, in der häuslichen Nachbarschaft oder auch in der Arbeitswelt. Da es sich bei den Teilnehmern dieser sozialen Netzwerke um reale Personen und Beziehungen handelt, werden sie im Folgenden auch mit dem Begriff *reale soziale Netzwerke* (*RSNs*) referenziert.

Virtuelle soziale Netzwerke Für nahezu jedes reale soziale Netzwerke findet man auch eine digitale bzw. virtuelle Repräsentation. Ein Beispiel dafür sind Datenbanken zur Publikation wissenschaftlicher Arbeiten, auf deren Basis man *Coautor-*, *Zitations-* und *Kollaborationsnetzwerke* ableiten kann. Auch aus Filmdatenbanken wie der Internet Movie Database [260] lassen sich soziale Netzwerke extrahieren. Die Verbindungen zwischen den Knoten eines *Movie-Actor-Netzwerks* geben z.B. Auskunft darüber, welche Schauspieler bereits gemeinsam in einem Film mitgewirkt haben. OSNs stellen eine Teilmenge *virtueller sozialer Netzwerke* dar. Bekannte Vertreter sind Facebook [246], Google+ [255], Twitter [276], YouTube [284], Flickr [249] und Last.fm [262].

OSNs stehen im Mittelpunkt dieser Arbeit. Das folgende Kapitel geht im Detail auf Eigenschaften und Besonderheiten von OSNs ein.

2.3 Online Soziale Netzwerke

Derzeit existieren mehrere hundert OSNs [281], die auf unterschiedlichste Zielgruppen und Anwender spezialisiert sind. Zu den Funktionen eines OSNs zählen das Veröffentlichen und Teilen persönlicher Informationen und Erfahrungen in Form von Photos, Videoclips oder Texten, die Kommunikation mit Freunden über Textnachrichten oder Chats, das Suchen nach personalisierten Inhalten, nach Freunden mit gemeinsamen Interessen oder potentiellen Geschäftspartnern und Mitarbeitern, oder auch die Teilnahme an einer Interessengemeinschaft [23].
Grob lassen sich OSNs bezüglich ihrer konkreten Ausprägung als *profilbezogen* (z.B. Facebook und Google+), *geschäftsbezogen* (z.B. LinkedIn [263] und Xing [282]), *hobbybezogen* (z.B. Couchsurfing [240] und Care2 [236]) sowie *inhaltebezogen* (z.B. YouTube, Flickr und Last.fm) klassifizieren [34].
Aufgrund der zahlreichen unterschiedlichen Merkmale ist eine exakte Definition des Begriffs OSN eher schwierig. Allen OSNs ist jedoch gemein, dass sie dem Aufbau bzw. der Pflege sozialer Kontakte dienen und dass sie von den publizierten persönlichen Informationen der Nutzer sowie deren Beziehungen untereinander profitieren.

2.3.1 Allgemeine Definition

Eine weit verbreitete Definition für OSNs liefern Boyed und Ellison [34]. Im Rahmen dieser Arbeit dient sie als Grundlage für die Beschreibung der gemeinsamen Eigenschaften von OSNs.
Boyed und Ellison charakterisieren ein OSN als einen Dienst, der es seinen Mitgliedern ermöglicht,

1. ein öffentliches oder halböffentliches Nutzerprofil innerhalb eines abgeschlossenen Systems zu erstellen,

2. eine öffentlichen Liste anderer Nutzer zu publizieren, mit denen sie eine Verbindung unterhalten und

3. die Listen ihrer eigenen Verbindungen und der Verbindungen anderer in ihrem abgeschlossenen System zu betrachten und zu traversieren.

An dieser Stelle sei angemerkt, dass sich die Definition auf die *Möglichkeit* zur Ausübung dieser drei Funktionen bezieht. Aus Sicht des einzelnen Nutzers kann sich der Funktionsumfang stark reduzieren. Grund dafür sind vor allem Konfigurationsmöglichkeiten, die es einem Nutzer erlauben, die Sichtbarkeit persönlicher und privater Informationen für jeden einzelnen Kontakt individuell festzulegen.

2.3.2 Weitere Eigenschaften

Unabhängig von seiner konkreten Ausprägung bietet ein OSN verschiedene Methoden zur Kommunikation. Dabei handelt es sich z.B. um den Versand von Nachrichten ähnlich einer E-Mail, dem Austausch von Nachrichten in Echtzeit ähnlich einem Chat oder auch der öffentlichen Bereitstellung von Informationen wie dem Eintrag auf einer Pinnwand oder in einem Forum. Von OSN zu OSN kann sich das Angebot an Kommunikationsmöglichkeiten stark unterscheiden. Auch die Untermenge an Nutzern, die man über einen Kommunikationskanal erreicht, unterscheidet sich gegebenenfalls stark. In der Regel bietet ein OSN die Möglichkeit, Sicherheits- und Privatsphäreneinstellungen so zu modifizieren, dass der Zugriff auf veröffentlichte Nachrichten wie z.B. einen Pinnwand- oder Foreneintrag nur einem bestimmten Nutzerkreis erlaubt ist.

Neben dem Austausch von Nachrichten bieten OSNs die Möglichkeit zum Publizieren und Teilen persönlicher und öffentlicher Inhalte. Dabei handelt es sich z.B. um Links auf Webseiten, Blogeinträge, Fotos, Musikdateien oder Videos.

Viele der geteilten Inhalte sind nur für einen eingeschränkten Nutzerkreis interessant. Oftmals können sie aber auch für die breite Öffentlichkeit von Bedeutung sein. Daher bieten viele OSNs auch die Möglichkeit, explizit nach bestimmten Personen oder Inhalten zu suchen. Einige OSNs bieten mittlerweile die Funktionalität einer *sozialen Suchmaschine*. Der Unterschied zu herkömmlichen Suchmaschinen wie die von Google oder Yahoo besteht darin, dass sie soziale Faktoren bei der Filterung der Suchergebnisse berücksichtigen.

Eine weitere Gemeinsamkeit stellt die zentralisierte Verwaltung von Nutzerprofilen und persönlichen Inhalten dar. Anbieter wie Facebook oder Google besitzen eine globale Sicht auf den gesamten durch ihre Nutzer bereitgestellten Datenbestand. Anders als z.B. P2P-Netzwerke unterstützt ein zentralisiertes System die einfache Aufbereitung und Bereitstellung aller im OSN publizierten Informationen.

Die Interaktion mit einem OSN erfolgt über eine Webanwendung oder, z.B. bei Nutzung eines Smartphones, über eine gesonderte Applikation. Sie fungiert dabei ausschließlich als Schnittstelle zum OSN. Alle publizierten Inhalte und getätigten Suchanfragen werden zentral vom Dienstanbieter verwaltet bzw. durchgeführt.

Die kostenlose Nutzung zahlreicher OSNs wird überwiegend durch Werbung ermöglicht. Im Gegensatz zu anderen (kostenlosen) Webportalen wie z.B. GMX [253] oder Web.de [280] besitzen OSNs jedoch ein extremes Alleinstellungsmerkmal. Betreiber eines OSNs haben nicht nur Kenntnis von Profilen, Vorlieben und Verhaltensweisen ihrer Nutzer, sondern auch von deren sozialen Beziehungen untereinander. Werbung lässt sich nicht nur mit Hilfe der auf die eigene Person bezogenen Daten filtern, sondern auch unter Einbeziehung persönlicher Informationen befreundeter Nutzer personalisieren.

2.4 P2P-Netzwerke

Peer-to-Peer (*P2P*) ist ein Paradigma zur Kommunikation in Computernetzwerken. Es stellt die Grundlage zahlreicher internetbasierter Systeme dar. Anwendungen und Protokolle finden sich in den Bereichen File-Sharing wie z.B. bei Gnutella [254] oder BitTorrent [237], Internettelephonie wie z.B. bei Skype [273] oder auch P2P-Fernsehen wie z.B. Zattoo [285]. Aufgrund ihrer dezentralen Natur eignen sich P2P-Systeme auch als Grundlage für den Entwurf eines sichereren und die Privatsphäre schützenden OSNs.
Die Vorteile von P2P-Ansätzen gegenüber zentralisierten Systemen reichen von ihrer hohen Skalierbarkeit und Robustheit bis hin zu den stark reduzierten Kosten für deren Betrieb [8]. Einen Nachteil stellt die erhöhte Komplexität für die Sicherstellung der Integrität und Vertraulichkeit in der Kommunikation dar [58, 210]. Diese hängt jedoch stets von der konkreten Ausprägung eines P2P-Systems ab.
P2P-Netzwerke basieren auf einem virtuellen Netzwerk von Knoten und ihren Verbindungen [8]. Da die logische Struktur nicht vom zugrunde liegenden physischen Computernetzwerk abhängt, bezeichnet man P2P-Netzwerke auch als *Overlay-Netzwerke*.
P2P-Systeme unterscheiden sich in ihrer Topologie, ihrer Struktur und ihrem Grad der Dezentralisierung. Zudem kommen verschiedene Routing-Strategien und Lokalisierungsmechanismen zum Einsatz, um Suchanfragen bzw. Nachrichten zuzustellen und Inhalte zu verteilen. Zu den weiteren wichtigen Unterscheidungsmerkmalen zählen Skalierbarkeit, Robustheit, Betriebskosten, die Erreichbarkeit und Persistenz von Daten, Integrität und Vertraulichkeit, Fairness und die Möglichkeiten zur Verwaltung gemeinsamer Ressourcen [8].
Grundsätzlich klassifiziert man Overlay-Netzwerke entsprechend dem Grad ihrer Zentralisierung und Struktur [194].

2.4.1 Unstrukturierte Systeme

P2P-Systeme der ersten Generation wie Napster [267] und Gnutella zählen zur Klasse *unstrukturierter P2P-Netzwerke*. Man spricht von einem unstrukturierten P2P-System, wenn dieses keine deterministische Auflösung einer Suchanfrage garantiert. Da in einem unstrukturierten System Verbindungen im Overlay willkürlich entstehen, führt ein und dieselbe Suchanfrage unter Umständen zu sehr unterschiedlichen Antworten.
Inhalte werden in der Regel über einen Index lokalisiert. Die Indexierung erfolgt durch die Anwendung einer Hash-Funktion auf Eigenschaften wie der Dateigröße, dem Dateinamen oder Schlagwörtern mit Bezug zum Inhalt der Datei. Die meisten unstrukturierten Systeme unterstützen daher auch *unscharfe* Suchanfragen wie die Suche nach (einer Kombination von) Schlüsselwörtern [46]. Nur solange Suchanfragen lokal aufgelöst werden sind unstrukturierte Systeme auch skalierbar.
Ein Nachteil unstrukturierter P2P-Systeme liegt in ihrer willkürlichen Ausbildung von Verbindungen. Abhängig von der Netzwerktopologie können Suchanfragen

fehlschlagen, obwohl ein Inhalt im P2P-Netzwerk existiert. Unstrukturierte P2P-Systeme können daher die Lokalisierung eines Inhalts nicht garantieren.
Man unterscheidet zwischen *zentralisierten*, *dezentralisierten* und *hybriden* bzw. *hierarchischen* Systemen.

2.4.1.1 Zentralisierte Systeme

Um Suchanfragen aufzulösen, verwenden zentralisierte Systeme eine zentrale Instanz. Diese verwaltet einen zentralen Index aller im System veröffentlichten Inhalte. Beim Beitritt eines neuen Knotens wird der zentrale Index um den Index des neuen Knotens erweitert.
Sucht ein Knoten nach einem bestimmten Inhalt, sendet dieser seine Suchanfrage an die zentrale Instanz. Diese antwortet mit ein oder mehreren Adressen der Knoten, die den gesuchten Inhalt bereitstellen. Der daraufhin ablaufende Datentransfer läuft ausschließlich zwischen den beteiligten Knoten ab.
Suchanfragen können in zentralisierten Systemen sehr schnell und zuverlässig aufgelöst werden. Zentralisierte Systeme besitzen jedoch ein Skalierungsproblem. Die zentrale Instanz stellt einen Engpass dar. Deren Ausfall führt zur Unbrauchbarkeit des Systems.
Eines der bekanntesten Beispiele für ein zentralisiertes P2P-System ist die File-Sharing-Plattform Napster. Gerichtliche Schritte der Musikindustrie gegen die Copyright-Verletzungen von Napster führten 2001 jedoch zu dessen Stilllegung [41].

2.4.1.2 Dezentrale Systeme

Dezentrale oder auch *reine P2P-Systeme* bieten keinerlei zentralen Index, forcieren keine spezifische Netzwerktopologie und machen keine Annahme über die Bereitstellung von Inhalten. Sie sind komplett dezentral und selbstorganisiert. Die beteiligten Knoten unterscheiden sich nicht in ihrer Funktionalität.
Neue Knoten treten dem Netzwerk bei, indem sie die Verbindungen anderer Knoten kopieren und mit der Zeit weitere eigene Verbindungen etablieren. Da kein zentraler Index existiert, benötigen dezentrale P2P-Systeme einen Mechanismus zur Lokalisierung der veröffentlichten Inhalte. Die dazu notwendigen Suchverfahren reichen von reinem Fluten (engl: *flooding*) [83] und Zufallsläufen (engl: *random walks*) [28] bis hin zur lokalen Indexierung beim Routing (engl: *routing indices*) [51]. Diese Verfahren sind zwar äußerst tolerant gegenüber häufigen Änderungen der Netzwerktopologie, skalieren jedoch nicht und verursachen unter Umständen eine hohe Last auf den einzelnen Knoten.
Um den Skalierungsproblemen zu begegnen, wurden zahlreiche Optimierungen vorgeschlagen. Dazu zählen das Zwischenspeichern von Suchanfragen (engl: *query caching*) [192], die expandierende Ringsuche (engl: *expanding ring search*) [139], k-Random Walks [139] oder auch die adaptive Flusskontrolle (engl: *adaptive flow control*) [140].

Gnutella in seiner Version 0.4 [250] ist das bekannteste Protokoll, das in dezentralen P2P-Netzwerken eingesetzt wurde. Aufgrund der Skalierungsprobleme setzen etablierte P2P-Plattformen nicht mehr auf rein dezentrale Protokolle.

2.4.1.3 Hybride Systeme

Um die Vorteile zentralisierter und dezentraler Systeme zu vereinen und gleichzeitig deren Nachteile zu minimieren, wurden hybride P2P-Systeme entwickelt. Diese ordnen Knoten entsprechend ihrer Rechen- und Speicherkapazität, ihrer Zuverlässigkeit und ihrer Netzwerkkonnektivität unterschiedlichen Aufgaben zu. Ziel ist die Ausbildung einer hierarchischen Struktur, in der eine kleine Menge leistungsfähiger Knoten stellvertretend für eine großen Menge leistungsschwacher Knoten die Auflösung aller Suchanfragen übernimmt.
Bei hybriden Systemen handelt es sich in der Regel um reine P2P-Systeme, wobei eine Teilmenge aller Knoten dynamisch zu privilegierten Knoten (engl: *superpeers*) erhoben wird. Welche Knoten zu Superpeers ernannt werden, hängt von den Anforderungen des jeweiligen P2P-Netzwerks ab. In der Regel handelt es sich um Eigenschaften wie die Rechenkapazität oder die Konnektivität eines Knotens. Superpeers übernehmen als Stellvertreter einer Untermenge an Knoten zusätzliche Aufgaben. Ein Superpeer stellt beispielsweise einen zentralen Index für alle ihm untergeordneten Knoten bereit. Suchanfragen können somit direkt durch einen Superpeer beantwortet werden.
Die Superpeers bilden untereinander ein eigenes Overlay-Netzwerk. Theoretisch können auch mehr als zwei Hierarchiestufen zum Einsatz kommen.
Der bekannteste Vertreter eines hybriden P2P-Protokolls mit zwei Hierarchiestufen ist Gnutella in seiner Version 0.6 [261].

2.4.2 Strukturierte Systeme

Unstrukturierte Systeme leiden unter Skalierungsproblemen und der nichtdeterministischen Auflösung von Suchanfragen. Durch Einführung eines global konsistenten Protokolls versuchen strukturierte P2P-Systeme, diesen Problemen zu begegnen. Knoten verbinden sich nach klar definierten Regeln. Auch die Indexierung publizierter Inhalte erfolgt nach einem einheitlichen Schema.
Zur Speicherung und Suche von Inhalten kommt meist eine *verteilte Hash-Tabelle* (*DHT*, engl: *distributed hash table*) [43] zum Einsatz. In Analogie zu einer gewöhnlichen Hash-Tabelle dient ein DHT der effizienten Datenspeicherung in verteilten Systemen. Mit Hilfe einer Hash-Funktion werden die Adressen der gespeicherten Inhalte auf die Schlüssel eines linearen Wertebereichs abgebildet. Jeder Knoten ist für die Verwaltung einer Teilmenge dieses Wertebereichs verantwortlich. Über ein dezentrales Protokoll handeln die Knoten ihre Zuständigkeiten für die entsprechenden Teilmengen aus. Die Lokalisierung der verantwortlichen Knoten erfolgt auf der Basis eines verteilten Routing-Verfahren.
Auf der Suche nach der Adresse eines bestimmten Inhalts generiert ein Knoten mit der Hash-Funktion den zugehörigen Schlüssel. Danach initiiert er im DHT eine

Suchanfrage nach der unter dem Schlüssel gespeicherten Adresse. Das eingesetzte Routing-Verfahren garantiert, dass die Suchanfrage denjenigen Empfänger erreicht, der für die entsprechende Teilmenge verantwortlich ist. Abhängig von der Existenz eines zugehörigen Eintrags liefert der Empfänger dem Initiator der Suchanfrage die Adresse des gespeicherten Inhalts bzw. eine Fehlermeldung als Antwort zurück.
Um einen Inhalt und seine Adresse als Schlüssel/Wert-Paar (engl: *key value pair*) im System zu publizieren, steht einem Knoten die *put(key,value)*-Funktion zur Verfügung. Über die *get(key)*-Funktion erhält ein Knoten Zugriff auf den indexierten Inhalt.
Ein Nachteil strukturierter P2P-Systeme liegt in ihrer mangelhaften Unterstützung unscharfer Suchanfragen [179]. Anders als in unstrukturierten Systemen werden Suchanfragen nicht lokal, sondern von dem für den jeweiligen Index verantwortlichen Knoten aufgelöst. Die Verwendung und Verteilung eines Schlüssel/Wert-Paares für alle möglichen Schlagworte würde nicht skalieren. Dennoch finden sich in der Literatur einige Ansätze, die diesem Problem begegnen [197, 179, 132].

2.4.2.1 Flache DHT-Systeme

Die erste Generation strukturierter P2P-Netzwerke basiert auf einem *flachen* DHT-Design. Die bekanntesten Vertreter sind *CAN* [178], *Chord* [195], *Pastry* [183], *Tapestry* [229] und *Kademlia* [144]. Alle Systeme sind dezentral aufgebaut und selbstorganisiert. Alle Knoten sind gleichberechtigt und verbinden sich mit derselben systemabhängigen Anzahl von Nachbarn. Die durchschnittliche Anzahl n an Hops, die eine Suchanfrage bis zur Lokalisierung eines Inhalts benötigt, wird mit einer Komplexität von $O(log(n))$ garantiert.

2.4.2.2 Hierarchische DHT-Systeme

Ein charakteristisches Merkmal aller P2P-Netzwerke ist ihre hohe Dynamik. Während sich immer wieder Knoten vom System trennen, treten auch kontinuierlich neue Knoten bei. Dieser Umstand ist weitläufig auch unter dem Begriff *Churn*-Effekt [196] bekannt.
Der Churn-Effekt wirkt sich in der Regel stärker in strukturierten als in unstrukturierten Systemen aus. Der Grund dafür liegt in dem zusätzlichen Aufwand zur Aufrechterhaltung eines konsistenten DHTs. Geht ein Knoten u offline, so müssen alle Knoten einen Ersatz für u finden, die u bisher als potentiellen Nachbarn für die Weiterleitung von Suchanfragen verwendet haben. Auch die Integration neuer Knoten in die verteilte Struktur setzt in der Regel die Erzeugung von $O(log(n))$ neuen Verbindungen voraus. Aufgrund der schlechteren Konnektivität verstärkt sich der Churn-Effekt bei der Teilnahme mobiler Endgeräte enorm.
Eine Gegenmaßnahme besteht in der Einführung einer mehrstufigen Hierarchie [77, 79]. *Hierarchische DHT-Systeme* haben eine gewisse Ähnlichkeit mit hybriden unstrukturierten Systemen. Die meisten Ansätze unterscheiden zwischen hoch- und niedriggradigen Knoten, wobei hochgradige Knoten als Superpeers die Kontrolle

über die P2P-Kommunikation in der höheren Ebene des Overlay-Netzwerks übernehmen. Niedriggradige Knoten organisieren sich in der unteren Ebene und beschränken ihren Einfluss auf einen lokalen Bereich.
Die meisten in der Literatur beschriebenen Beispiele hierarchischer DHT-Systeme basieren auf Chord und Pastry [191, 232, 151, 74, 10, 149, 79].

2.5 Graphen- und netzwerktheoretische Definitionen

Zur Charakterisierung, Analyse und Modellierung von RSNs und OSNs aber auch jeglicher anderer Ausprägung von Netzwerken hat sich die Graphentheorie bewährt. Im Folgenden werden alle auf der Graphentheorie basierenden Definitionen, Begrifflichkeiten, Metriken und Konzepte vorgestellt, die für soziale Netzwerke im Kontext dieser Arbeit von Bedeutung sind. Zudem werden einige netzwerktheoretische Konzepte präsentiert. Sie dienen als Grundlage zur Beurteilung und Charakterisierung sozialer Graphen auf der Basis ihrer Struktur.

2.5.1 Struktur sozialer Graphen

Im Folgenden werden die wichtigsten Begrifflichkeiten für die Beschreibung struktureller Eigenschaften sozialer Netzwerke vorgestellt.

2.5.1.1 Sozialer Graph

RSNs und OSNs können als Graph $G = (V, E)$ [121] dargestellt werden, wobei V die Menge aller *Nutzer* und $E = \{(u, v) : u, v \in V\}$ die Menge aller *Beziehungen* zwischen u und v darstellt. Beschreibt ein Graph die Struktur und die Eigenschaften eines sozialen Netzwerks, dann spricht man von einem *sozialen Graphen*. Abhängig von den Eigenschaften eines OSNs eignet sich zur strukturellen Beschreibung ein gerichteter bzw. ein ungerichteter Graph [121]. Solange es nicht explizit angegeben ist, werden im Rahmen dieser Arbeit soziale Graphen immer als ungerichtete Graphen interpretiert.
Die Begriffe *Nutzer*, *Individuum*, *Person*, *Teilnehmer*, *Mitglied* und *Knoten* werden synonym verwendet. Ebenfalls als Synonyme werden die Begriffe *Beziehung*, *Freundschaft*, *Verbindung* und *Kante* eingesetzt. Um eine Kante zwischen zwei Knoten u und v zu referenzieren, wird auch die Notation $e(u, v)$ angewandt.

2.5.1.2 Kontakt- und Interaktionsgraph

Der *Kontaktgraph* eines sozialen Netzwerks beschränkt sich ausschließlich auf das Wissen über die strukturelle Verbundenheit beteiligter Knoten.
Kontaktgraphen eignen sich nur bedingt für die Analyse von OSNs. Nach Ausbildung einer Freundschaft erfolgt zwischen dem überwiegenden Anteil der Nutzer keine weitere Interaktion.

Um der Häufigkeit sozialer Interaktionen mehr Gewicht zu verleihen, wurde das Konzept der *Interaktionsgraphen* entwickelt [217]. Ein Interaktionsgraph umfasst alle temporalen Ereignisse zwischen den Knoten eines sozialen Graphen, die jegliche Art der Interaktion zwischen den Mitglieder eines OSNs beschreiben. In einem Interaktionsgraphen existiert eine Verbindung zwischen zwei Knoten nur dann, wenn sie direkt interagiert oder über eine bestimmte Anwendung miteinander kommuniziert haben.
Wie stark sich der Interaktions- vom Kontaktgraphen unterscheidet, hängt maßgeblich davon ab, wie sich Interaktionen definieren. Wird schon das Schließen einer Freundschaft als Interaktion interpretiert, dann sind Kontakt- und Interaktionsgraph zueinander isomorph.

2.5.1.3 Nachbar und Nachbarschaft

Existiert zwischen zwei Nutzern u und v eine (ungerichtete) Kante $e(u, v)$, dann bezeichnet man v als den *Nachbarn* von u bzw. u als den Nachbarn von v. Die Menge aller Nachbarn $\Gamma(u) = \{v : (u, v) \in E\}$ bezeichnet man als *Nachbarschaft* von u. Im Rahmen dieser Arbeit werden die Begriff *Nachbar*, *Kontakt* und *Freund* synonym verwendet.

2.5.1.4 Gemeinsamer Nachbar

Unabhängig davon, ob zwei Knoten eine Freundschaft unterhalten oder nicht, besteht die Möglichkeit, dass sich ihre Nachbarschaft überlappt. Man bezeichnet einen Knoten w als einen *gemeinsamen Nachbarn* der Knoten u und v, wenn gilt $w \in \Gamma(u) \cap \Gamma(v)$. Die Anzahl gemeinsamer Nachbarn von u und v berechnet sich als $|\Gamma(u) \cap \Gamma(v)|$.

2.5.1.5 Egonetzwerk

Das *Egonetzwerk* eines Nutzers u besteht aus der Menge der Knoten $V_u = \{u \cup \Gamma(u)\}$ und der Menge aller Verbindungen $E_u = \{(u, v) : u, v \in V_u\}$. Das Egonetzwerk umfasst neben den Kanten zwischen einem Nutzer und seiner Nachbarschaft auch alle Kanten zwischen zwei disjunkten Knoten seiner Nachbarschaft.

2.5.1.6 Freundeskreis höherer Ordnung

Unter dem *Freundeskreis n-ter Ordnung* eines Nutzers u versteht man die Menge aller Knoten, die man von u aus durch das Traversieren von minimal n Verbindungen erreicht. Bei der Nachbarschaft handelt es sich folglich um den Freundeskreis erster Ordnung.

2.5.1.7 Hop, Lauf und Pfad

Unter einem *Hop* versteht man den Übergang bzw. den Weg von einem Knoten zu einem seiner Nachbarn. Ein *Lauf* (engl: *walk*) entspricht einer Abfolge von Kno-

ten, in der jeder Knoten ein Nachbar des vorhergehenden und des nachfolgenden Knotens darstellt. In einem Lauf können sowohl Knoten als auch Kanten mehrmals auftreten. Bei einem *Pfad* handelt es sich um einen Lauf, in dem jeder Knoten und jede Kante genau ein Mal auftritt.

2.5.1.8 Hub

Man bezeichnet einen Knoten als *Hub*, wenn dieser eine weitaus höhere Anzahl von Verbindungen zu anderen Knoten besitzt als der Durchschnitt. Gerade in P2P-Netzwerken und OSNs nehmen Hubs oft eine zentrale Rolle für die Verbreitung von Informationen ein.

2.5.2 Eigenschaften und Maßzahlen sozialer Graphen

Für die Analyse sozialer Graphen können zahlreiche Eigenschaften untersucht werden. Im Folgenden werden Eigenschaften und Maßzahlen definiert, die im Rahmen dieser Arbeit von Bedeutung sind.

2.5.2.1 Knotengrad

In einem ungerichteten Graphen gibt der *Knotengrad* darüber Auskunft, wie viele Verbindungen ein Knoten zu anderen Knoten unterhält. Der Grad $deg(u)$ eines Knotens u entspricht der Anzahl seiner Nachbarn und berechnet sich entsprechend der Gleichung $deg(u) = |\Gamma(u)|$ [76]. In ungerichteten Graphen kann man zwischen Eingangs- und Ausgangsknotengraden unterscheiden. Der Eingangsknotengrad eines Knotens u entspricht der Anzahl gerichteter Kanten, die u als Endknoten besitzen. Umgekehrt bezeichnet der Ausgangsknotengrad von u die Anzahl der gerichteten Kanten, deren Startknoten u darstellt. Solange es nicht explizit angegeben ist, wird im Rahmen dieser Arbeit immer auf den Knotengrad eines ungerichteten Graphen Bezug genommen.
Der *durchschnittliche Knotengrad* $deg(G)$ eines Graphen G entspricht dem Durchschnitt der Knotengrade aller seiner Knoten. Er berechnet sich entsprechend der Gleichung

$$deg(G) = \frac{\sum_{u \in V} deg(u)}{|V|}.$$

2.5.2.2 Kürzester und durchschnittlich kürzester Pfad

Abhängig von der Struktur des Graphen existieren häufig mehrere unterschiedliche Pfade zwischen zwei Knoten u und v. Der Pfad mit dem man von u in einer minimalen Anzahl von Schritten zum Knoten v gelangt, wird auch als *kürzester Pfad* oder als *Distanz* bezeichnet. Um den kürzesten Pfad zwischen zwei Knoten u und v zu referenzieren, wird im Folgenden die Notation $d(u, v)$ verwendet. Der *durchschnittlich kürzeste Pfad* $dkP(G)$ eines Graphen G entspricht dem Durchschnitt

der kürzesten Pfade zwischen allen möglichen disjunkten Kombinationen zweier Knoten u und v. Er berechnet sich entsprechend der Gleichung

$$dkP(G) = \frac{\sum_{u \neq v \in V} d(u,v)}{|V|}.$$

2.5.2.3 Radius und Durchmesser

Der *Radius* $Rad(G)$ eines Graphen G entspricht der Länge des kürzesten aller kürzesten Pfade zwischen jeglicher Kombination zweier disjunkter Knoten u und v. Der Radius berechnet sich entsprechend der Gleichung

$$Rad(G) = \min_{u \neq v \in V} d(u,v).$$

Der *Durchmesser* bzw. *Diameter* $Dia(G)$ eines Graphen G entspricht der Länge des längsten aller kürzesten Pfade zwischen jeglicher Kombination zweier disjunkter Knoten u und v. Der Durchmesser berechnet sich entsprechend der Gleichung

$$Dia(G) = \max_{u \neq v \in V} d(u,v).$$

2.5.2.4 Clustering-Koeffizient

Allgemein spricht man von *Clustering*, falls die Wahrscheinlichkeit einer Verbindung zwischen zwei Knoten mit der Anzahl ihrer gemeinsamen Nachbarn steigt [134]. Der *Clustering-Koeffizient* dient als Maß, den Grad der Vernetzung der Nachbarn eines Nutzers untereinander zu bestimmen. Der Clustering-Koeffizient $CK(u)$ eines Nutzers u berechnet sich entsprechend der Gleichung

$$CK(u) = \frac{|\{(v,w) : v \neq w \in \Gamma(u) \wedge (v,w) \in E\}|}{\frac{1}{2}|\Gamma(u)|(|\Gamma(u)| - 1)}.$$

Ein hoher Clustering-Koeffizient ist ein Indiz dafür, dass ein Knoten zu einer stark vernetzen Gruppe gehört. Bezieht sich der Clustering-Koeffizient auf einen einzelnen Nutzer, spricht man vom *lokalen* Clustering-Koeffizienten.

Als charakteristische Kennzahl eines sozialen Graphen wird häufig nur der *globale* Clustering-Koeffizient $CK(G)$ angegeben. Er berechnet sich als Durchschnitt aller lokalen Clustering-Koeffizienten entsprechend der Gleichung

$$CK(G) = \frac{\sum_{u \in V} CK(u)}{|V|}.$$

Ein hoher globaler Clustering-Koeffizient lässt auf eine hohe Anzahl stark vernetzter Untergruppen von Nutzern schließen.

2.5.2.5 Lokale Transitivität

Eine weitere Eigenschaft sozialer Netzwerke stellt die Transitivität von Beziehungen dar [216]. Zwei Nutzer v und w, die jeweils eine Beziehung zu einem Nutzer u unterhalten, besitzen mit höherer Wahrscheinlichkeit auch untereinander eine Beziehung, als wenn sie keine Beziehung zu u hätten.
In sozialen Netzwerken führt eine hohe Transitivität oft zur Ausbildung lokaler Cluster.

2.5.3 Netzwerktheoretische Definitionen

Mit Ausnahme des globalen Clustering-Koeffizienten fokussieren sich die bisher erläuterten Kenngrößen sozialer Graphen auf die Eigenschaften einzelner Knoten. Daneben spielen bei der Charakterisierung aber auch globale statistische Eigenschaften wie das Kleine-Welt-Phänomen oder die Skaleninvarianz eine wichtige Rolle [170].
Im Folgenden werden die im Kontext dieser Arbeit wichtigsten netzwerktheoretischen Definitionen vorgestellt.

2.5.3.1 Kleine-Welt-Phänomen

Der Begriff des *Kleine-Welt-Phänomens* (engl: *small world phenomenon*, auch bekannt als *six degrees of separation* [88]) bezieht sich auf die Hypothese, dass jeder Mensch über eine sehr kurze Kette von sozialen Beziehungen mit jedem anderen Menschen auf der Welt verbunden ist.
Den ersten Nachweis für die Gültigkeit dieser Vermutung lieferten Milgram [148], Travers [201] und White [215] in ihren “Kleine-Welt-Experimenten”. Milgram stellte fest, dass der durchschnittlich kürzeste Pfad für das soziale Netzwerk der USA zwischen fünf und sechs Personen umfasst [148]. Später konnte diese Hypothese auch in einem weiteren Kleine-Welt-Experiment auf der Basis des Microsoft Messenger Instant-Messaging-Systems [130] sowie des E-Mail-Systems [61, 84] bestätigt werden.
Das Kleine-Welt-Phänomen lässt sich auch bei einer Vielzahl realer Netzwerke und Graphen aus Natur und Technik beobachten. Dazu zählen neuronale Netzwerke, das Energieversorgungsnetzwerk der USA, wissenschaftliche Zitationsgraphen oder auch der Hyperlink-Graph des WWW [213]. Netzwerke, die dem Kleine-Welt-Phänomen folgen, werden auch als *Kleine-Welt-Netzwerke* bezeichnet.
Neuste Studien [205, 11] bestätigen die Existenz des Kleine-Welt-Phänomens auch für OSNs. Für den sozialen Graphen von Facebook ergibt sich ein durchschnittlich kürzester Pfad mit weniger als fünf Knoten.
Neben sehr geringen durchschnittlich kürzesten Pfadlängen besitzen Kleine-Welt-Netzwerke auch eine starke Tendenz zur Bildung von Clustern [114].

2.5.3.2 Skaleninvarianz und Potenzgesetz

Ein weiteres charakteristisches Merkmal sozialer Netzwerke stellt ihre *Skaleninvarianz* dar [19, 21]. Man bezeichnet ein Netzwerk als *skaleninvariant* oder *skalenfrei*, wenn der Anteil $P(k)$ an Knoten vom Grad k einem *Potenzgesetz* der Form

$$P(k) \sim k^{-\gamma}$$

folgt. γ bezeichnet man als den *Potenzgesetzkoeffizienten* (engl: *powerlaw coefficient*) oder einfach nur als *Koeffizienten*. Es handelt sich dabei um eine einheitslose Kenngröße.

Eine Verteilung der Knotengrade folgt einem Potenzgesetz, wenn sehr viele Knoten mit einem sehr kleinen, viele Knoten mit einem mittleren und einige wenige Knoten mit einem sehr hohen Grad existieren. Potenzgesetze sind skaleninvariant bezüglich Streckung und Stauchung. Die Skalierung mit einem konstanten Faktor c bewirkt ausschließlich eine proportionale Skalierung des Potenzgesetzes selbst. Es gilt

$$P(ck) \sim (ck)^{-\gamma} = c^{-\gamma}k^{-\gamma} \sim c^{-\gamma}P(k) \sim P(k).$$

Die Skalierung mit c bewirkt also lediglich eine Multiplikation mit der Konstanten $c^{-\gamma}$.

Eine charakteristische Eigenschaft skaleninvarianter Netzwerke ist das Auftreten von Hubs. Zudem zeigen skaleninvariante Netzwerke kleine maximale Distanzen zwischen beliebigen Knotenpaaren. Sie besitzen daher einen kleinen Durchmesser. Dieser wächst logarithmisch mit der Anzahl der Knoten. Da skaleninvariante Netzwerke ein erhöhtes Clustering aufweisen, handelt es sich auch um Kleine-Welt-Netzwerke [6].

In einer Vielzahl realer Netzwerke folgt die Verteilung der Knotengrade einem Potenzgesetz [69, 5, 35, 6]. Auch für P2P-Netzwerke [108, 139, 192] und OSNs [150, 217, 122] wurde gezeigt, dass die Verteilung der Knotengrade einem Potenzgesetz gehorcht.

Daneben existieren aber auch Beispiele [3, 205, 187], bei denen die Verteilung für wachsende Knotengrade ein exponentielles bzw. gaußsches Abfallen aufzeigt (engl: *truncated powerlaw*). Diese meist exponentielle Abweichung von der Verteilung eines Potenzgesetzes bezeichnet man als *Cut-Off*. Der Cut-Off lässt sich mit einem zusätzlichen Faktor modellieren. Folgt die Verteilung der Knotengrade einem Potenzgesetz mit exponentiellem Cut-Off, dann lässt sich der Anteil $P(k)$ an Knoten vom Grad k über die Beziehung

$$P(k) \sim k^{-\gamma} f(k)$$

mit $f(k) = e^{-\delta k}$ modellieren.

2.5.3.3 Gemeinschaftsstruktur und Modularität

Ein wichtiges Maß zur Klassifizierung sozialer Netzwerke stellt die *Gemeinschaftsstruktur* (engl: *community structure*) [161, 57] dar. Eine Gemeinschaft ist eine Gruppe stark vernetzter Knoten, wobei Knoten unterschiedlicher Gruppen untereinander wiederum nur schwach vernetzt sind [29]. Um eine Gemeinschaftsstruktur zu erkennen, benötigt man ein Verfahren zur *Partitionierung* eines Netzwerks. Eine Partitionierung entspricht der Aufteilung der Menge aller Knoten V in k disjunkte, nicht leere Teilmengen $U_1, \ldots, U_k$. Es muss also gelten $\bigcup_{i=1}^{k}(U_i) = V$, $U_i \neq \emptyset$ und $U_i \cap U_j = \emptyset$ für $1 \leq i, j \leq k$.

Das Partitionierungsverfahren darf keinerlei Annahme darüber treffen, wie viele Gemeinschaften in einem Graphen erkannt werden müssen bzw. wie viele Kanten höchstens zwischen unterschiedlichen Gemeinschaften verlaufen dürfen. Eine *valide* Partitionierung definiert sich lediglich dadurch, dass sich das Verhältnis der Anzahl der Kanten *innerhalb* von Gemeinschaften zu der Anzahl der Kanten *zwischen verschiedenen* Gemeinschaften deutlich von dem Verhältnis in einer Partitionierung unterscheidet, bei der die Identifikation von Gemeinschaften rein zufällig erfolgt [160].

Die statistisch auffällige Konstellation von Kanten lässt sich mit dem Maß der *Modularität* (engl: *modularity*) [168] quantifizieren. Die Modularität einer Partition beschreibt das Verhältnis der Dichte an Verbindungen innerhalb der existierenden Gemeinschaften zu der Dichte zwischen diesen Gemeinschaften. Sie berechnet sich durch die Anzahl der Kanten, die sich innerhalb aller Gemeinschaften befinden, abzüglich der erwarteten Anzahl der Kanten eines äquivalenten Netzwerks, bei dem die Kanten rein zufällig platziert werden [160].

Die Modularität $Mod(G)$ eines Graphen G entspricht einem skalaren Wert zwischen -1 und 1 [168] und berechnet sich entsprechend der Gleichung [160, 29]

$$Mod(G) = \frac{1}{2m} \sum_{u,v \in V} \left[A_{uv} - \frac{k_u k_v}{2m} \right] \delta(c_u c_v).$$

A_{uv} entspricht dem Gewicht der Kante $e(u, v)$. $k_u = \sum_{v \in V} A_{uv}$ repräsentiert die Summe der Kantengewichte, die zum Knoten u führen. c_u referenziert die Gemeinschaft von Knoten u. Die δ-Funktion $\delta(i, j) = 1$ falls $i = j$ und 0 sonst. $m = \frac{1}{2} \sum_{u,v \in V} A_{uv}$ ist die Summe aller Kantengewichte. Sind Kantengewichte nicht relevant, liefert die Matrixdarstellung von A_{uv} für alle möglichen Kombinationen von u und v die Adjazenzmatrix von G. k_u entspricht in diesem Fall dem Grad des Knotens u und m der Summe aller Knotengrade. Eine Gemeinschaftsstruktur ist nur für $Mod(G) > 0$ gegeben.

Es existiert eine Vielzahl an Verfahren zur Aufdeckung der Gemeinschaftsstruktur und zur Berechnung der Modularität [81, 168, 177, 49, 220, 160]. Ein sehr schnelles Verfahren ist das von Blondel et. al [29]. Es wird im Rahmen dieser Arbeit für die Berechnung der Modularität sozialer Netzwerke verwendet.

2.6 Zusammenfassung

Dieses Kapitel befasste sich mit den Grundlagen sozialer Netzwerke, lieferte einen Überblick zu P2P-Systemen und erläuterte die im Kontext dieser Arbeit relevanten graphen- und netzwerktheoretischen Definitionen.
Einer kurzen Darstellung der Begrifflichkeiten Sicherheit und Privatsphäre folgte eine Charakterisierung und Unterscheidung realer, virtueller und Online Sozialer Netzwerke. Danach wurden OSNs allgemein definiert und deren typische Eigenschaften aufgezählt.
Die unterschiedlichen Klassen und Ausprägungen von P2P-Netzwerken wurden vorgestellt. Insbesondere strukturierte P2P-Netzwerke dienen als Grundlage zahlreicher dezentraler, sicherer und die Privatsphäre schützender OSNs. Einige solcher P2P-basierten OSNs werden im nächsten Kapitel untersucht.
Es wurden die im Rahmen dieser Arbeit verwendeten graphentheoretischen Begrifflichkeiten erläutert. Für die Beschreibung struktureller Eigenschaften sozialer Netzwerke wurden die dafür notwendigen Maßzahlen präsentiert. Sie stellen die Grundlage für die spätere Analyse der Strategien zur Priorisierung von Suchanfragen innerhalb dezentraler OSNs dar. Für das Verständnis charakteristischer Netzwerkstrukturen innerhalb sozialer Netzwerke wurden die dafür relevanten netzwerktheoretischen Definitionen vorgestellt. Diese finden insbesondere bei der Analyse und Modellierung sozialer Graphen Anwendung.

3 Sicherheit und Schutz der Privatsphäre in OSNs

Mittlerweile existiert für jede erdenkliche Interessengemeinschaft ein spezialisiertes OSN (vgl. Kap. 2.3.1). So vielfältig wie ihre unterschiedlichen Ausprägungen sind auch die Plattformen und Architekturen, auf denen sie basieren. Überwiegend handelt es sich um zentralisierte Systeme unter einer einzigen administrativen Domäne. Neben dem uneingeschränkten Zugriff auf persönliche Inhalte haben deren Betreiber die Möglichkeit, alle strukturellen und semantischen Verknüpfungen zwischen ihren Nutzern nachzuvollziehen. Mit über einer Milliarde Nutzern [251] besitzt Facebook dieses Potential für ein siebtel der gesamten Weltbevölkerung.

In den letzten Jahren wurde in unzähligen Medienberichten über das Gefahrenpotential und den Missbrauch von OSNs aufgeklärt. Parallel wuchs der gesellschaftliche und politische Druck auf die Betreiber, für mehr Sicherheit und einen besseren Schutz persönlicher Daten zu sorgen [271, 257]. Als Antwort auf die Kritik rüsten Betreiber ihre Plattformen mit Konfigurationsmöglichkeiten für einen besseren Schutz der Privatsphäre auf.

In vielen Fällen steht dieses Vorgehen jedoch im Widerspruch zu den Interessen der Betreiber selbst. Im Hinblick auf das Vermarktungspotential persönlicher Daten [223, 85] sind Unternehmen wie Facebook und Twitter an einem möglichst starken Vernetzungsgrad ihrer Nutzer und einer hohen Anzahl öffentlich zugänglicher Profile interessiert.

Mit zunehmendem strukturellen und semantischen Informationsgehalt sozialer Graphen wächst auch die Bereitschaft Werbetreibender, für die individualisierte Platzierung von Werbebotschaften sowie den Zugang zu Daten für effektivere Kampagnen [279] zu bezahlen. Trotz entsprechender Regulierungsanstrengungen durch die zuständigen Regierungsbehörden kann nicht ausgeschlossen werden, dass finanzkräftige Werbetreibende direkten Zugriff auf die persönlichen Informationen der Nutzer erhalten. Jüngsten Erkenntnissen zu Folge besitzen Nachrichtendienste und Ermittlungsbehörden in den USA schon seit Jahren direkten Zugriff auf die Server von OSNs wie Google, Facebook und YouTube [252].

Die Sicherheits- und Privatsphäreeinstellungen von OSNs bieten offensichtlich keinen ausreichenden Schutz. Es sind das mangelnde Vertrauen in die Betreiber, in ihre Angestellten und in die Qualität der verwendeten Software, die den Schutz der Privatsphäre und die Sicherheit der Identität limitieren.

In den letzten Jahren wurden zahlreiche Konzepte und Implementierungen publiziert, die darauf abzielen, die Probleme und Risiken von OSNs zu beseitigen. In der Praxis haben sich zwei Richtungen etabliert:

- Die Integration zusätzlicher Sicherheitsmechanismen in ein bereits etabliertes zentralisiertes OSN [137, 89, 199].

- Die Entwicklung einer selbständigen unabhängigen dezentralen Lösung [189, 38, 55, 242, 86, 158].

Als weiterer Vertreter einer dezentralen Lösung entstand im Rahmen dieser Arbeit das sichere und die Privatsphäre schützende OSN Vegas [292].
Dieses Kapitel widmet sich der Konzeption von Vegas. Nach einer detaillierten Betrachtung unterschiedlicher Anforderungen werden zunächst einige existierende Architekturvorschläge diskutiert. Danach folgt eine detaillierte Beschreibung der Sicherheitsarchitektur und der Protokolle von Vegas.

3.1 Anforderungen an ein sicheres und die Privatsphäre schützendes OSN

Ein sicheres und die Privatsphäre schützendes OSN muss den grundlegenden Anforderungen der Informationssicherheit genügen [56]. Dazu gehören die Authentifizierung der Nutzer untereinander, die Integrität privater Daten und der autorisierte Zugriff auf persönliche Inhalte (vgl. Kap. 2.1).
Der Fokus liegt auf der Konzeption eines OSNs, das neben der Informationssicherheit den Schutz der Privatsphäre inklusive der Anonymität seiner Nutzer garantiert. Die Bereitstellung der notwendigen Maßnahmen ist mit einer steigenden Komplexität und einer eingeschränkten Benutzerfreundlichkeit der entsprechenden Anwendung verbunden.
Im Folgenden werden vier Anforderungen präsentiert, die ein sicheres und die Privatsphäre schützendes OSN erfüllen muss [296]. Anforderungen der Informationssicherheit wie Authentifizierung, Autorisierung und Datenintegrität werden als zwingend vorausgesetzt und an dieser Stelle nicht gesondert diskutiert.

3.1.1 Informationelle Selbstbestimmung

Die wesentliche Anforderung an ein sicheres und die Privatsphäre schützendes OSN stellt die Gewährleistung des Rechts auf informationelle Selbstbestimmung dar. Darunter versteht man das Recht jedes Einzelnen, persönlich über die Preisgabe und die Verwendung der eigenen personenbezogenen Daten zu bestimmen (vgl. Kap. 2.1).
Eine zentralisierte Administration von Nutzerprofilen und persönlichen Inhalten steht offensichtlich im Konflikt mit der Einhaltung des Rechts auf informationelle Selbstbestimmung [235]. Selbst wenn eine vertrauenswürdige dritte Partei existiert, die den sicheren und vertraulichen Zugriff auf die Daten innerhalb eines OSNs garantiert, bleibt das Problem der zentralisierten Administration weiterhin bestehen.

Ein sicheres und die Privatsphäre schützendes OSN darf auf keiner zentralisierten Komponente basieren. Die Sicherheit darf nicht vom Einsatz einer dritten vertrauenswürdigen Partei abhängen. Die Wahrung des Rechts auf informationelle Selbstbestimmung setzt voraus, dass das eigene Nutzerprofil und damit assoziierte persönliche Inhalte ausschließlich für den dafür vorgesehenen Personenkreis sichtbar sind.
Um seine Anonymität gegenüber allen anderen Teilnehmern zu wahren, muss es einem Nutzer möglich sein, die Kenntnis über die Existenz seiner Identität und seiner persönlichen Inhalte bewusst auf einen dedizierten Personenkreis einzuschränken. Auch der Zugriff auf persönliche Daten sollte für jedes Mitglied dieses Personenkreises selektiv konfigurierbar sein.
Das Recht auf informationelle Selbstbestimmung ist erst erfüllt, wenn ein Nutzer die Möglichkeit besitzt, flexibel und selektiv den Zugriff auf bereits veröffentlichte Informationen auch wieder zu unterbinden. Dies schließt die Möglichkeit ein, die eigene Mitgliedschaft komplett zu beenden und jeden weiteren Zugriff auf publizierte Inhalte auszuschließen.

3.1.2 Starkes Vertrauen in direkte Beziehungen

Die exakte Abbildung der Semantik einer realen Beziehung auf den virtuellen Kontakt innerhalb eines OSNs ist kaum möglich. Dies ist der Art und Weise geschuldet, wie Nutzer ihrem Profil neue Kontakte hinzufügen. Anders als in der Realität werden innerhalb eines OSNs aus flüchtigen Bekanntschaften sofort dauerhafte Kontakte. Oft rückt die Existenz solcher Kontakte jedoch schnell wieder aus dem Bewusstsein eines Nutzers. So erhalten auch Kontakte Zugriff auf persönliche Informationen, die man mit diesen in der Realität nie teilen würde.
Im Gegensatz zu virtuellen Kontakten hängt das Vertrauen in reale Personen von sozialen Aspekten wie der Dauer einer Bekanntschaft oder der Häufigkeit regelmäßiger Treffen ab [106]. Solche Aspekte stehen wiederum in direkter Beziehung zu den existierenden Verbindungen des realen sozialen Graphen. Stellt z.B. Nutzer u seinen Freund v einem anderen seiner Freunde w vor, dann bedeutet dies noch lange nicht, dass v das gleiche Vertrauen in w setzt wie u.
Das unüberlegte Akzeptieren eines entsprechenden virtuellen Freundschaftsangebots kann dazu führen, dass sehr persönliche Informationen ungewollt solchen Kontakten zugänglich gemacht werden. Zusätzlich führen derartige Freundschaften zu einem unerwünschten Anwachsen der eigenen Kontaktliste. Je mehr solcher Freundschaften man etabliert, desto höher liegt der Aufwand für die Pflege der eigenen Kontaktliste. Auch die Wahrscheinlichkeit für einen unerwünschten Zugriff auf persönliche Informationen durch Dritte nimmt zu.
Um dem Entstehen solcher Kontakte vorzubeugen, soll ein sicheres und die Privatsphäre schützendes OSN nur die Bildung von Verbindungen erlauben, für die in der Realität eine Beziehung existiert. Diese Anforderung trägt auch zur Wahrung der Anonymität der Nutzer bei. Die Suche nach unbekannten Personen und das Browsen ihrer Profile ist nicht ohne Unterstützung des Egonetzwerks möglich.

3.1.3 Permanenter Profilzugriff

Die Dezentralisierung stellt eine grundlegende Anforderung dar, dem Missbrauch bzw. der Rekonstruktion des sozialen Graphen durch Werbetreibende, Angreifer, Regierungsbehörden und andere dritte Parteien vorzubeugen. In einem dezentralen OSN ist jeder Nutzer selbst für die Verwaltung seines Profils und seiner persönlichen Inhalte verantwortlich. Ausgehend von einem reinen P2P-Ansatz ist das eigene Profil für Freunde nur noch dann zugreifbar, wenn der entsprechende Speicherort von überall und zu jeder Zeit erreichbar ist.

Als weitere Anforderung muss ein dezentrales OSN die Möglichkeit bieten, auf sichere und die Privatsphäre schützende Art und Weise, das eigene Profil zu publizieren und permanent erreichbar zu speichern. Der Zugriff muss auch dann möglich sein, wenn die entsprechende Person gerade nicht mit dem OSN in Verbindung steht.

3.1.4 Unterstützung der Mobilität

Die fortschreitende Entwicklung mobiler Endgeräte und Anwendungen (engl: *apps*) zeigt auch Auswirkungen auf das Interaktionsverhalten der Nutzer eines OSNs [127]. Betreiber wie Facebook und Google stellen ihren Nutzern Apps zur Verfügung, die den vom Web gewohnten Funktionsumfang ihrer OSNs implementieren.

Der Trend hin zur mobilen Nutzung von OSNs ist unbestritten. Heute interagieren bereits drei von vier Mitgliedern mit Facebook über ihr mobiles Endgerät [247]. Als letzte Anforderung soll auch ein dezentrales OSN mobile Nutzer explizit berücksichtigen.

Um grundlegende Anforderungen wie Authentifizierung, Autorisierung und Datenintegrität sicherzustellen, setzt die Mehrzahl dezentraler OSNs auf die Verwendung stark kryptographischer Verfahren. Angesichts hardwarespezifischer Einschränkungen wie einer reduzierten Prozessorleistung bzw. eines kleineren Hauptspeichers kann der Einsatz solcher Verfahren auf mobilen Endgeräten zu spürbaren Leistungseinbußen führen. Bei der Diskussion der Unterstützung mobiler Nutzer in dezentralen OSNs müssen auch die Auswirkungen hardwarespezifischer Einschränkungen untersucht werden.

Außerdem müssen eine zuverlässige Kommunikation und ein reibungsloser Zugriff auf persönliche Inhalte sichergestellt sein. Gerade in strukturierten P2P-Netzwerken wirkt sich ein hoher Anteil mobiler Nutzer sehr negativ auf die Effizienz und Zuverlässigkeit des Gesamtsystems aus. Deren Unterstützung misst sich insbesondere daran, wie robust ein OSN auf Anomalien wie den Churn-Effekt (vgl. Kap. 2.4.2.2) reagiert.

3.2 Klassifizierung existierender Ansätze

In der Literatur finden sich einige Konzepte und praktische Implementierungen dezentraler OSNs [189, 38, 55, 242, 86, 158]. Alle Ansätze verfolgen das Ziel, dem Nutzer mehr Sicherheit und Schutz der Privatsphäre zu garantieren. Gerade die Anonymität der Nutzer wird bei vielen Konzepten überhaupt nicht oder nur am Rande berücksichtigt.
Im Folgenden werden einige Konzepte dezentraler OSNs vorgestellt und im Hinblick auf die Erfüllung der genannten Anforderungen hin untersucht.

3.2.1 Persona

Bei Persona [14] handelt es sich um ein dezentrales OSN, dessen Sicherheitskonzept auf der Kombination traditioneller asymmetrischer und attributbasierter Verschlüsselung (*ABE*, engl: *attribute based encryption*) [25] aufbaut. Der Grund für die Kombination liegt in einer erhöhten Flexibilität zur Verwaltung von Nutzergruppen. Nach Auffassung der Autoren sind andere bekannte Verfahren der Gruppenverschlüsselung [219, 155] dadurch limitiert, dass sich der autorisierte Zugriff auf persönliche Inhalte für eine Schnittmenge der Mitglieder mehrerer Gruppen nur sehr umständlich realisieren lässt.

Konzept Persona unterscheidet zwei Kategorien von Objekten. Es existieren *Nutzer* (engl: *users*), welche die Inhalte generieren und *Anwendungen* (engl: *applications*), die den Nutzern Dienste zum Verändern der Inhalte bereitstellen. Jeder Nutzer interagiert mit Persona über eine Browser-Erweiterung, die sämtliche kryptographische Operationen übernimmt.
Als Anwendungen existieren ein *Speicherdienst* (engl: *storage service application*) und ein *Dokumentendienst* (engl: *doc application*). Der Speicherdienst wird dazu verwendet, Freunden persönliche Inhalte verschlüsselt zur Verfügung zu stellen. Es werden lediglich Operationen zum Abrufen und zum Speichern bereitgestellt.
Der Dokumentendienst dient der Realisierung kollaborativer Angebote wie z.B. einer persönlichen Pinnwand. Nutzer können anderen Teilnehmern oder Gruppen Schreibrechte auf den eigenen Dokumentendienst zuweisen. Die einzige Anforderung an einen Speicher- bzw. Dokumentendienst stellt die Implementierung einer vorgegebenen Programmierschnittstelle dar.
Jeder Nutzer kann individuell entscheiden, welche seiner persönlichen Informationen er mit welchen seiner Freunde teilt. Dazu verwaltet jeder Nutzer seine Freunde als Gruppen. Die Zugehörigkeit zu einer Gruppe basiert auf den gemeinsamen Attributen, die ein Nutzer mit den Mitgliedern dieser Gruppe teilt.
Jeder Nutzer generiert einen öffentlichen Schlüssel (*APK*, engl: *ABE public key*) und einen Generalschlüssel (*AMSK*, engl: *ABE master secret key*). Neben der Verschlüsselung dient der APK als Identifikator eines Mitglieds. Der APK wird *außer Band* (*OOB*, engl: *out of band*) den Freunden mitgeteilt. Hinzu kommt pro Freund ein

geheimer Schlüssel (*ASK*, engl: *ABE secret key*). Dieser wird mit Hilfe des AMSK erzeugt.

Der ASK basiert auf der Menge aller Attribute, welche diejenige Gruppe definiert, der ein entsprechender Freund angehören soll. Für die Verschlüsselung eines Objekts muss eine Zugriffsstruktur in Form eines logischen Ausdrucks der Attribute spezifiziert werden. Wurde eine Information z.B. mit der Zugriffsstruktur `("Bergsteiger" ODER "Nachbar")` verschlüsselt, so kann ein anderer Nutzer diese genau dann entschlüsseln, wenn er zu einer Gruppe gehört, der mindestens eines der beiden Attribute zugeordnet wurde.

Vordefinierte Gruppen existieren letztendlich nicht. Sie werden erst bei der Verschlüsselung der Daten auf der Basis des entsprechenden ASKs implizit erzeugt. Die Kenntnis eines APKs und seiner assoziierten Attribute reicht aus, um eine Information mit der gewünschten Zugriffsstruktur zu verschlüsseln.

Der einzige Weg, Mitglieder aus einer Gruppe auszuschließen, besteht in der Erzeugung neuer ASKs für jedes verbleibende Gruppenmitglied. Um zukünftigen Freunden die bisherigen Inhalte einer Gruppe vorzuenthalten, können Zugriffsstrukturen mit Ungleichheiten wie `(Datum < 2013)` versehen werden.

Diskussion Mit der dezentralen Datenhaltung über Speicher- und Dokumentendienste erfüllt Persona die Anforderung an einen permanenten Profilzugriff. Nutzer können auf persönliche Inhalte eines Gruppenmitglieds zugreifen, auch wenn dieses gerade offline ist.

Die von Persona angedachten Anwendungen sind äußerst komplex. Jede neue Anwendung erfordert die Implementierung einer Autorisierungslogik für die Einhaltung entsprechender Zugriffskontrolllisten. Dafür reicht in ABE schon die Kenntnis des APKs und des ASKs aus, den Zugriff auf Informationen für die an den ASK gebundenen Gruppenmitglieder zu beschränken. Bezogen auf die informationelle Selbstbestimmung besteht hier jedoch ein gravierendes Problem: Gruppenmitglieder werden implizit durch die Attributliste eines ASKs determiniert. Die Zuweisung der Attribute an die Gruppenmitglieder kennt jedoch lediglich der Erzeuger der ASKs. Unter Umständen wissen nicht alle Mitglieder über ihre gemeinsame Gruppenzugehörigkeit Bescheid. Es besteht also die Gefahr, dass verschlüsselte Inhalte ungewollt von unbekannten Gruppenmitgliedern mitgelesen werden. Zudem basiert die Identifikation eines Nutzers auf einem einzigen APK. Eine Deanonymisierung der Nutzer ist in Persona möglich.

APKs werden in Persona direkt zwischen den Nutzern außer Band ausgetauscht. Dies impliziert ein starkes Vertrauen in eine Freundschaft.

Persona wurde auch für den Einsatz auf mobilen Endgeräte hin untersucht. Als klarer Nachteil erweist sich der enorme Rechenaufwand für die attributbasierte Verschlüsselung. Im Vergleich zu RSA [180] ist eine ABE-Operation zwischen hundert und tausend mal langsamer. Lediglich die Mehrfachverwendung symmetrischer Schlüssel kann dem Problem entgegenwirken.

3.2.2 Safebook

Safebook [55, 56], ein dezentrales OSN auf der Basis eines strukturierten P2P-Overlays, forciert den Schutz der Privatsphäre seiner Mitglieder und ihren Schutz vor böswilligen Angreifern. Safebook setzt dazu auf die Ausbildung starker Vertrauensbeziehungen zwischen seinen Nutzern. Der Ansatz verfolgt die Sicherstellung der Vertraulichkeit in der Kommunikation, die Zugriffskontrolle auf persönliche Daten sowie deren permanente Verfügbarkeit und Integrität.

Konzept Die Architektur von Safebook unterteilt sich in drei Schichten. Die *soziale Netzwerkschicht* (*SN*, engl: *social network layer*) beinhaltet die digitalen Repräsentationen der Nutzer und ihrer Beziehungen. Die *Dienstanwendungsschicht* (*AS*, engl: *application service layer*) beschreibt die Anwendungsinfrastruktur des Betreibers. Die *Kommunikations- und Transportschicht* (*CT*, engl: *communication and transport layer*) umfasst die Kommunikations- und Transportdienste des Netzwerks.
In der Implementierung von Safebook dient das Internet als CT-Schicht. Angreifer treten als böswillige Nutzer auf der SN-Schicht, als böswillige Betreiber auf der AS-Schicht oder als böswillige dritte Partei auf der CT-Schicht in Erscheinung.
Bei den drei Kernkomponenten von Safebook handelt es sich um *Matrjoschkas*, das P2P-Overlay sowie einen *vertrauenswürdigen Identifikationsdienst* (*TIS*, engl: *trusted identification service*).
Die verteilte Struktur einer Matrjoschka dient der vertraulichen Kommunikation und der Speicherung persönlicher Inhalte. Sie besteht aus mehreren logischen konzentrischen Kreisen um einen Nutzer. Der innerste Kreis umfasst das Egonetzwerk eines Nutzers. Es handelt sich um die Kontakte, denen ein Nutzer vertraut. Bei den Knoten auf dem Ring der Ebene $n + 1$ handelt es sich jeweils um die engsten Kontakte der Knoten auf Ebene n.
Jeder Nutzer repliziert seine persönlichen Inhalte auf den Knoten seines innersten Rings. Soll eine Nachricht an einen Nutzer u weitergeleitet werden, so gelangt diese zunächst an einen Knoten des äußersten Kreises der Matrjoschka von u. Über die inneren Ringe wird die Nachricht sukzessive zu u transferiert. Die Vertraulichkeit wird dadurch gewährleistet, dass Knoten ausschließlich mit ihrem Egonetzwerk kommunizieren. Kein Knoten auf dem Pfad zu u hat direkte Kenntnis vom eigentlichen Empfänger der Nachricht.
Alle Teilnehmer sind in einem P2P-Overlay organisiert. Ein DHT hilft dabei, die Profilinformationen anderer Nutzer aufzufinden. Dazu werden für jeden Nutzer Verweise auf die Knoten seines äußersten Rings im DHT registriert. Der Pfad einer Suchanfrage von u zu Nutzer v wird erst über den DHT und dann über die Matrjoschka von v bestimmt. Trifft die Suchanfrage auf einen der äußersten Knoten der Matrjoschka von v, wird sie über je ein Mitglied der konzentrischen Ringe an v delegiert. Die Antwort wird auf dem gleichen Pfad an u zurück geleitet.
Der TIS stellt sicher, dass sich nur real existierende Personen bei Safebook registrieren können. Außer Band wird jedem Nutzer ein eindeutiges Schlüssel/Wert-Paar zugeordnet. Dieses besteht aus dem Knotenidentifikator und einem Pseudonym. Der

Knotenidentifikator wird auf der Basis ausgewählter Nutzerattribute wie Name, Geburtstag und Geburtsort generiert. Ein Nutzer registriert sich mit seinem Pseudonym im P2P-Overlay, bevor er sein Egonetzwerk und seine Matrjoschka etabliert. Ab diesem Zeitpunkt kann ein Nutzer u über die äußersten Knoten der Matrjoschka von Nutzer v das Profil und andere persönliche Inhalte von v abrufen.

Diskussion Bei Safebook handelt es sich um ein strukturiertes P2P-System. Die Speicherung persönlicher Inhalte erfolgt direkt auf dem Endgerät eines Nutzers. Trotz der Replikation im Egonetzwerk ist die Anforderung an eine permanente Verfügbarkeit nur bedingt erfüllt. Sind alle Knoten des Egonetzwerks offline, kann nicht auf die persönlichen Inhalte eines Nutzers zugegriffen werden.
Die Replikation persönlicher Inhalte erfolgt in verschlüsselter Form. Ein Nutzer hat aber keinen Einfluss darauf, ob und wann ein Nachbar seine Inhalte löscht.
Der Einsatz eines TIS als zentrale Instanz verstößt gegen das Recht auf informationelle Selbstbestimmung. Nutzer müssen sich dort mit sensiblen Informationen wie Name, Geburtstag oder Adresse registrieren. Der TIS nimmt somit eine ähnliche Stellung wie der Betreiber eines zentralisierten OSNs ein.
Der TIS soll das gegenseitige Vertrauen der Nutzer sicherstellen. Es handelt sich hierbei jedoch um eine dritte Partei. Kryptographische Schlüssel werden mit dem TIS zwar außer Band ausgetauscht. Die Anforderung an eine starke Vertrauensbeziehung *direkt* zwischen den Nutzern ist damit jedoch nicht erfüllt.
Für Safebook finden sich keine Informationen zu den eingesetzten kryptographischen Verfahren. Zudem existieren keine Annahmen zur technischen Beschaffenheit beteiligter Endgeräte. Eine Beurteilung der Performanz auf mobilen Endgeräten ist daher nicht möglich. Die Verwendung eines DHT-basierten P2P-Overlays lässt jedoch darauf schließen, dass ein hoher Anteil mobiler Nutzer sich sehr negativ auf die Effizienz und Zuverlässigkeit des Gesamtsystems auswirkt. Die Unterstützung mobiler Nutzer ist durch den Churn-Effekt stark limitiert.

3.2.3 PeerSoN

Wie Safebook beruht das dezentrale OSN PeerSoN [38, 39] auf einem P2P-Overlay. Konzeptionell basieren Sicherheit und der Schutz der Privatsphäre auf dem Einsatz asymmetrischer Verschlüsselung.

Konzept PeerSoN verwendet als Overlay einen DHT. Jeder Knoten repräsentiert einen Nutzer und beteiligt sich an der verteilten Speicherung der persönlichen Daten aller Mitglieder. Zur Lokalisierung der Nutzer kommen global eindeutige Identifikatoren (*GUIDs*, engl: *globally unique IDs*) zum Einsatz. PeerSoN verwendet als GUID den Hash-Wert der E-Mail Adresse eines Nutzers.
Das System basiert auf dem *Login-*, dem *Getting-a-File-* und dem *Asynchronous-Messages-Protokoll*.
Mit dem Login-Protokoll registriert sich ein Nutzer mit seinen entsprechenden Metadaten im Netzwerk. Unter der GUID wird im DHT ein Datensatz als

Schlüssel/Wert-Paar eingefügt. Darin sind die verschiedenen Standorte bzw. Endgeräte zusammen mit dem aktuellen Verbindungsstatus eines Nutzers hinterlegt. Mit jedem Login bringt ein Nutzer diese Informationen auf den aktuellen Stand. Will sich Nutzer u mit Nutzer v verbinden, ruft u über die GUID von v im DHT den entsprechenden Datensatz ab. Daraus extrahiert u den aktuellen Standort von v.

Über das Getting-a-File-Protokoll können die Profildaten eines Nutzers abgerufen werden. Für jedes Datum wird im DHT ein Schlüssel/Wert-Paar hinterlegt. Das Schlüssel/Wert-Paar gibt Auskunft darüber, welche Knoten in welcher Version die zugehörige Datei gespeichert haben. Ein Nutzer erlangt Zugriff auf eine Datei, indem er über das Schlüssel/Wert-Paar die Adresse des Knotens mit der aktuellen Version identifiziert und sich dann direkt mit diesem Knoten verbindet. Alle Dateien sind durch eine entsprechende Zugriffskontrolle geschützt.

Will u mit v kommunizieren, erfolgt die Nachrichtenübermittlung direkt. Ist v gerade offline, kommt das Asynchronous-Messages-Protokoll zum Einsatz. u hinterlegt zunächst die Nachricht unter einem speziellen Schlüssel im DHT. Sobald sich v zu einem späteren Zeitpunkt wieder einloggt, kann v über den Schlüssel die Nachrichten abrufen.

Vor jedem Datenaustausch wird zusätzlich ein Handshake-Protokoll ausgeführt. Zum einen beugt dies etwaigen Spam-Problemen vor, zum anderen unterstützt es die Angabe der Größe der zu transferierenden Daten. Ist ein Nutzer beispielsweise gerade über sein Smartphone eingeloggt, dann kann er dynamisch entscheiden, ob der Datentransfer sofort oder erst bei Bestehen einer breitbandigen Verbindung durchgeführt werden soll.

Diskussion Die Speicherung persönlicher Inhalte erfolgt in PeerSoN komplett verschlüsselt und dezentral. Im Gegensatz zu Safebook kann ein Nutzer jedoch nicht festlegen, welchen Knoten er traut und auf welchen seine Daten hinterlegt werden sollen. Das Löschen persönlicher Inhalte kann durch den Nutzer ebenfalls nicht erzwungen werden. Zudem sind Deanonymisierungsattacken wie bei Safebook möglich. Das Recht auf informationelle Selbstbestimmung ist nur teilweise erfüllt.

Da sich PeerSoN bei der Speicherung persönlicher Inhalte ausschließlich auf das P2P-Overlay verlässt, ist die permanente Verfügbarkeit der Nutzerprofile und persönlicher Inhalte nicht garantiert.

Die Art der Verschlüsselung und der Zugriffskontrollverfahren wird für PeerSoN nur sehr vage formuliert. Würden öffentliche Schlüssel außer Band übertragen, wäre zumindest die Anforderung an starke Vertrauensbeziehungen sichergestellt. Die Anonymität der Nutzer wäre damit aber noch nicht garantiert.

Mit der Option, sich mit mehreren Endgeräten gleichzeitig im Netzwerk anzumelden und mit dem Einsatz des Asynchronous-Messages- und des Handshake-Protokolls liefert PeerSoN die Grundlage zur Unterstützung mobiler Teilnehmer. Zudem existiert in PeerSoN eine Erweiterung, die es den Nutzern erlaubt, Daten auch ohne Internetverbindung über mobile Ad-Hoc-Netzwerke wie Bluetooth oder WLAN direkt auszutauschen. Wie Safebook unterliegt PeerSoN dem Churn-Effekt.

Ein hoher Anteil mobiler Nutzer wirkt sich negativ auf die Effizienz und Zuverlässigkeit des Gesamtsystems aus.

3.2.4 Vis-à-Vis

Vis-à-Vis [189] ist ein dezentrales OSN auf der Basis eines strukturierten P2P-Overlays. Die Speicherung persönlicher Inhalte erfolgt exklusiv auf einem eigenen Server. Dieser läuft in einer virtuellen Maschine (*VM*) innerhalb einer Cloud-Computing-Infrastruktur (*CCI*) wie z.B. Amazon EC2 [245]. Optional haben Nutzer weiterhin die Möglichkeit, persönliche Inhalte auf ihrem Endgerät zu speichern.

Konzept Vis-à-Vis führt das Konzept *virtueller individueller Server* (*VISs*, engl: *virtual individual servers*) ein. Zur Speicherung persönlicher Inhalte verwendet jeder Nutzer seine eigene VIS-Instanz. Basierend auf dem Hash-Wert der IP-Adresse erhält jede VIS-Instanz einen eindeutigen Identifikator. Dieser dient der Organisation aller VIS-Instanzen innerhalb des mehrschichtig angelegten DHTs.
Bei der obersten Schicht handelt es sich um die *Metagruppe* (engl: *meta group*). Sie dient der Verwaltung aller VIS-Instanzen und der Suche nach anderen Nutzern.
Vis-à-Vis erlaubt die Speicherung persönlicher Inhalte auf dem eigenen Endgerät bzw. auf einer Kombination aus Endgerät und VIS-Instanz. Die ausschließliche Verwendung einer VIS-Instanz bietet jedoch zwei wichtige Vorteile. Zum einen garantiert eine CCI die permanente Verfügbarkeit der VMs, zum anderen übernimmt der Betreiber die gesamte Administration wie z.B. das Einspielen neuester Sicherheits-Updates.
Vis-à-Vis definiert zwei Kategorien persönlicher Informationen: *Eingeschränkte Informationen* (engl: *restricted information*) können nur von vertrauenswürdigen Kontakten eingesehen werden. *Suchbare Informationen* (engl: *searchable information*) stehen einer größeren Nutzergemeinde zur Verfügung und erlauben die Suche nach unbekannten Personen mit ähnlichen Interessen. Eingeschränkte und suchbare Informationen müssen auf unterschiedliche Art und Weise im DHT verwaltet werden.
Beim Zugriff auf eingeschränkte Informationen fungiert die VIS-Instanz als Referenzmonitor. Nur Personen mit den entsprechenden Zugriffsrechten können diese Daten einsehen. Die Konfiguration der Zugriffsregeln erfolgt durch den Nutzer selbst. Zudem speichert die VIS-Instanz eines Nutzers für jeden seiner Freunde einen gemeinsamen geheimen Schlüssel zusammen mit einem Verweis auf die VIS-Instanz des entsprechenden Freundes. Dieser Schlüssel wird beim Schließen einer neuen Freundschaft, z.B. auf der Basis des Diffie-Hellman-Protokolls [60], erzeugt. Er dient dem Aufbau eines sicheren Kommunikationskanals zwischen sich gegenseitig vertrauenden VIS-Instanzen.
Suchbare Informationen werden in Form von *typisierten Gruppen* (engl: *typed groups*) organisiert. Als Identifikatoren dienen die Metagruppen. Die Auflösung auf ein einzelnes Element übernimmt die zugehörige VIS-Instanz lokal. Jede typisierte Gruppe besteht aus Nutzern mit gemeinsamen Attributen oder Interessen. Als

Referenz für eine Gruppe wird eine Kombination aus dem Typ und einem entsprechenden Schlüssel verwendet.
Suchanfragen lassen sich an alle Mitglieder einer typisierten Gruppe stellen. Bei der Erzeugung bzw. dem Beitritt zu einer typisierten Gruppe wird als Identifikator der Gruppe der Hash-Wert der Referenz verwendet. Der Identifikator dient dazu, diejenige VIS-Instanz zu bestimmten, die schließlich die entsprechende Suchanfrage bearbeitet.
Der Beitritt zu bzw. die Anfrage an eine typisierte Gruppe wird automatisch durch die Zustimmung aller gegenwärtigen Gruppenmitglieder oder über einen nicht näher spezifizierten Authentifizierungsmechanismus geregelt. Es können auch versteckte typisierte Gruppen angelegt werden. Nur Mitglieder dieser Gruppen werden über deren Existenz in Kenntnis gesetzt. Das Hinzufügen von Informationen zu einer typisierten Gruppe wird durch das Einfügen eines entsprechenden Schlüssel/Wert-Paares in den DHT realisiert.

Diskussion Die Verwendung einer CCI garantiert jedem Nutzer die permanente Verfügbarkeit seiner persönlichen Inhalte. Damit verbunden sind jedoch zusätzliche Kosten für das Anmieten und den Betrieb einer VM.
Die Organisation der Teilnehmer erfolgt in Vis-à-Vis dezentral. Zudem verwaltet und speichert jeder Nutzer seine persönlichen Informationen ausschließlich auf seiner eigenen VIS-Instanz. Das Recht auf informationelle Selbstbestimmung ist dennoch nicht sichergestellt. Da auf den VIS-Instanzen keine Verschlüsselung erfolgt, haben Betreiber die Möglichkeit, persönliche Inhalte auszulesen und zu kopieren.
Vis-à-Vis liefert keine ausreichende Anonymisierung persönlicher Informationen. Deanonymisierungsattacken sind aufgrund suchbarer Informationen weiterhin möglich. Auch Sybil-Attacken [64] werden nicht explizit ausgeschlossen. Der bösartige Betrieb einer unbestimmten Anzahl von VIS-Instanzen ermöglicht ein Vortäuschen beliebiger Identitäten.
Der Austausch kryptographischer Schlüssel erfolgt in Vis-à-Vis außer Band. Die Voraussetzungen für eine starke Vertrauensbeziehung sind damit erfüllt.
Vis-à-Vis forciert die Unterstützung mobiler Teilnehmer. Die hohe Verfügbarkeit der CCI ermöglicht auch mobilen Nutzern einen schnellen und zuverlässigen Zugriff auf das OSN. Informationen über die Effizienz der kryptographischen Operationen und den mobilen Betrieb fehlen.

3.2.5 Beurteilung der vorgestellten Ansätze

Offensichtlich wird keiner der betrachteten Ansätze allen der in Kapitel 3.1 genannten Anforderungen gerecht.
Keines der vier dezentralen OSNs gewährleistet in vollem Umfang das Recht auf informationelle Selbstbestimmung. Während ein unbewusster Zugriff auf persönliche Inhalte in Persona nicht ausgeschlossen werden kann, werden sie in Vis-à-Vis unverschlüsselt hinterlegt. Betreiber einer CCI besitzen das Potential, persönliche

Inhalte auszulesen und zu kopieren. Für Safebook und PeerSoN können Deanonymisierungsattacken nicht ausgeschlossen werden.

In Persona und Vis-à-Vis erfolgt der Austausch kryptographischer Schlüssel direkt zwischen den Nutzern außer Band. Damit ist die Anforderung an eine starke Vertrauensbeziehung erfüllt. Für PeerSoN ist der Schlüsselaustausch nicht näher spezifiziert. Auch in Safebook werden kryptographische Schlüssel außer Band ausgetauscht. Die starke Vertrauensbeziehung existiert aber nur zwischen dem Nutzer und dem TIS und nicht direkt zwischen den einzelnen Teilnehmern.

Die Verwendung gesonderter Speicherdienste bzw. virtueller Server stellt den permanenten Profilzugriff in Persona bzw. Vis-à-Vis sicher. In Safebook und PeerSoN erfolgt die Speicherung ausschließlich im P2P-Overlay. Beide Systeme leiden unter Anomalien wie dem Churn-Effekt. Die Anforderung eines permanenten Profilzugriffs ist hier nicht erfüllt.

Alle vier Systeme besitzen das Potential zur Unterstützung mobiler Teilnehmer. Aufgrund der P2P-Charakteristik von Safebook wirkt sich eine verstärkte Teilnahme mobiler Nutzer jedoch sehr negativ auf die Effizienz und Zuverlässigkeit des Gesamtsystems aus. Trotz der Replikation persönlicher Inhalte ist auch PeerSoN anfällig für den Churn-Effekt. Bezogen auf die Effizienz und Zuverlässigkeit des Gesamtsystems schneidet Vis-à-Vis mit seinem VIS-Konzept bei der Unterstützung mobiler Nutzer am Besten ab.

Zur Effizienz eingesetzter kryptographischer Verfahren auf mobilen Endgeräten existieren kaum Informationen. Lediglich für Persona kann man festhalten, dass der Einsatz von ABE deutlich mehr Rechenaufwand verursacht als beispielsweise RSA.

Tabelle 3.1 gibt abschließend einen Überblick über die Erfüllung der Anforderungen in den betrachteten dezentralen OSNs.

OSN	Informationelle Selbstbestimmung	Starke Vertrauensbeziehung	Permanenter Profilzugriff	Unterstützung der Mobilität
$Persona$	$\times$	$\surd$	$\surd$	$\ominus$
$Safebook$	$\ominus$	$\times$	$\times$	$\times$
$PeerSoN$	$\ominus$	$?$	$\times$	$\ominus$
Vis-à-Vis	$\times$	$\surd$	$\surd$	$\ominus$

Tabelle 3.1: Überblick über die Erfüllung der Anforderungen an die untersuchten dezentralen OSNs ($\surd$ = erfüllt, $\times$ = nicht erfüllt, $\ominus$ = teilweise erfüllt, $?$ = keine Informationen vorhanden).

3.3 Konzeption einer sicheren und die Privatsphäre schützenden Architektur

Mit einer dezentralen Organisation begegnet jedes der im vorhergehenden Kapitel diskutierten Systeme dem Problem einer singulären administrativen Domäne. Bezogen auf die Anforderungen aus Kapitel 3.1 bietet jedoch keines einen ausreichenden Schutz der Privatsphäre seiner Nutzer.
Mit *Vegas* [292] liefert das folgende Kapitel das Konzept eines dezentralen OSNs, das mit seiner restriktiven Sicherheitsarchitektur diese Lücke füllt.
Zunächst folgt ein Überblick über den Schutz der Privatsphäre eines Nutzers und die Sichtbarkeit seiner persönlichen Informationen innerhalb von Vegas. Ausgehend von diesem Blickwinkel wird die Kernarchitektur von Vegas vorgestellt, bevor sich die anschließenden Kapitel mit dem Kommunikationsprotokoll, der Datenhaltung, und weiteren wichtigen Funktionen von Vegas beschäftigen.

3.3.1 Überblick

Um die starke Vertrauensbeziehung in virtuelle Kontakte sicherzustellen, beschränkt Vegas Freundschaften auf solche Beziehungen, die in der Realität auch tatsächlich existieren. Dazu wird in Vegas der Blickwinkel eines Nutzers auf sein eigenes Egonetzwerk limitiert. Entgegen der Definition eines Egonetzwerks (vgl. Kap. 2.5.1.5) hat ein Nutzer in Vegas nicht einmal Kenntnis darüber, ob zwei seiner Nachbarn ebenfalls eine Freundschaft pflegen oder nicht.
Abbildung 3.1 veranschaulicht die Situation eines Nutzers u im Kontext des sozialen Graphen von Vegas. u kennt ausschließlich diejenigen Mitglieder, die sich in dessen persönlichen Freundeskreis $\Gamma(u) = \{v_1, v_2, v_3\}$ befinden. Das Browsen eines unbekannten Mitglieds $w_i \in V \setminus (\Gamma(u) \cup u)$ ist nicht möglich. Auch direkte Freundschaftsanfragen zwischen u und w_i sind ausgeschlossen.
Betrachtet man die Freundschaft zwischen u und einem Freund $v_i \in \Gamma(u)$ isoliert, dann kennt u lediglich die persönlichen Informationen, für die v_i den Zugriff explizit gewährt hat (in Abb. 3.1 symbolisiert durch ein Schloss). u hat keinerlei Kenntnis der Beziehungen zwischen zwei seiner Freunde v_i und v_j ($v_i, v_j \in \Gamma(u)$) oder einem seiner Freunde v_i und einem unbekannten Nutzer $w_i \notin \Gamma(u)$.

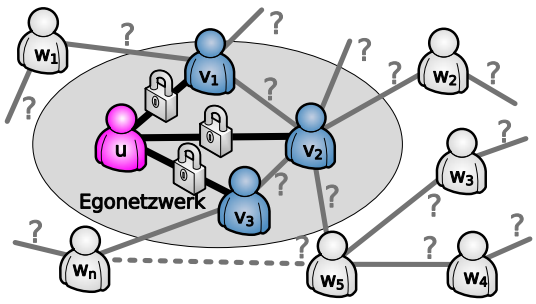


Abbildung 3.1: Blickwinkel eines Nutzers auf den sozialen Graphen von Vegas.

Um diese Situation entsprechend den in Kapitel 3.1 beschriebenen Anforderungen zu modellieren, unterscheidet Vegas zwischen den folgenden Domänen:

Client-Domäne Die Client-Domäne umfasst alle Nutzer des Systems. Technisch gesehen wird ein *Client* durch ein mit der entsprechenden Vegas-Software konfiguriertes Endgerät des Nutzers repräsentiert.

Exchanger-Domäne Zur Kommunikation zweier Clients dient in Vegas die Exchanger-Domäne. Bei einem *Exchanger* handelt es sich um das abstrakte Konzept eines Kommunikationskanals. Ein Exchanger ermöglicht den sicheren und anonymen Austausch von Nachrichten und anderen persönlichen Informationen zwischen zwei Clients. Um die permanente Verfügbarkeit und die Unterstützung mobiler Teilnehmer sicherzustellen, muss jeder Exchanger die Möglichkeit zur Zwischenspeicherung und zur asynchronen Auslieferung einer Nachricht bieten.

Datastore-Domäne Bei einem *Datastore* handelt es sich um das abstrakte Konzept eines Datenspeichers. Ein Datastore dient einem Nutzer zum permanenten Speichern seiner persönlichen Informationen und Inhalte. Die Gewährleistung des Rechts auf informationelle Selbstbestimmung erfordert die volle Zugriffskontrolle eines Clients auf einen Datastore. Damit die Freunde eines Nutzers diese Informationen jederzeit abrufen können, muss ein Datastore die permanente Verfügbarkeit der dort abgelegten Daten garantieren.

Abbildung 3.2 veranschaulicht die Abhängigkeiten und das Zusammenwirken von je zwei Ausprägungen der drei Domänen. Client u legt seine persönlichen Inhalte auf Datastore DS_u ab. Client v kann diese Daten jederzeit über DS_u abrufen. Zur Kommunikation stellen sich u und v gegenseitig einen Exchanger zur Verfügung. u sendet seine Nachrichten an v über dessen Exchanger EX_v und empfängt Nachrichten von v über seinen eigenen Exchanger EX_u.
Die Anzahl und der Einsatz von Clients, Exchangern und Datastores pro Nutzer unterliegt keinen besonderen Einschränkungen. Zum Empfang von Nachrichten über seinen Client kann ein Nutzer seinen Freunden lediglich einen, aber auch beliebig

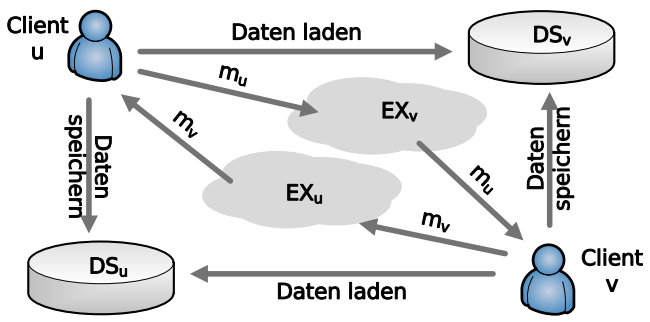


Abbildung 3.2: Abhängigkeiten und Zusammenwirken von je zwei Ausprägungen der Client-, der Exchanger- und der Datastore-Domäne.

viele Exchanger zur Verfügung stellen. Im Hinblick auf die Gewährleistung der eigenen Anonymität ist der Einsatz mehrerer Exchanger sehr zu empfehlen. Im Optimalfall stellt ein Nutzer jedem seiner Freunde einen dedizierten Exchanger zur Verfügung. Um die Wahrscheinlichkeit einer erfolgreichen Deanonymisierungsattacke zu minimieren, können auch mehrere Exchanger pro Freund zum Einsatz kommen. Aus denselben Gründen empfiehlt sich auch der Einsatz mehrerer Datastores.

3.3.2 Sicherheitskonzept

Um die Anforderungen an die Sicherheit und den Schutz der Privatsphäre zu gewährleisten, werden in Vegas alle Nachrichten und persönlichen Informationen in verschlüsselter Form versendet bzw. gespeichert. In Anlehnung an das Konzept *Pretty Good Privacy* (*PGP*) [80], das unter anderem eine sichere E-Mail-Kommunikation ermöglicht, kommt in Vegas eine Kombination aus asymmetrischer und symmetrischer Verschlüsselung zum Einsatz.
Die auszutauschenden Informationen werden zunächst auf der Basis eines symmetrischen Verfahrens verschlüsselt. Der symmetrische Schlüssel wird daraufhin mit dem öffentlichen Schlüssel des Empfängers verschlüsselt. Der verschlüsselte Inhalt setzt sich aus der verschlüsselten Information, dem asymmetrisch verschlüsselten symmetrischen Schlüssel, sowie einer Signatur auf der Basis des privaten Schlüssels des Absenders zusammen.
Ursprünglich basiert PGP auf dem Konzept des *Netz des Vertrauens* (*WOT*, engl: *web of trust*). Die Identität eines Nutzers wird mit einem eindeutigen öffentlichen Schlüssel assoziiert. Um für denselben Empfänger eine Nachricht zu verschlüsseln, verwenden beim WOT alle Sender den gleichen öffentlichen Schlüssel. Zudem verwendet ein Sender immer denselben privaten Schlüssel, um seine Nachrichten zu signieren.
Offensichtlich besteht bei Verwendung eines einzigen Schlüsselpaars das Risiko einer Deanonymisierung von Sender und Empfänger. Um in Vegas einen maximalen Schutz der Privatsphäre sicherzustellen, generiert jeder Nutzer u für jeden seiner Freunde $v \in \Gamma(u)$ ein *verbindunsspezifisches Schlüsselpaar* (*LSKP*, engl: *link specific key pair*) ($K^-_{u\to v}$/$K^+_{u\to v}$), wobei $K^-_{u\to v}$ den privaten und $K^+_{u\to v}$ den öffentlichen *verbindunsspezifischen Schlüssel* (*LSK*, engl: *link specific key*) referenziert.
Umgekehrt generiert auch jeder Freund v ein LSKP ($K^-_{v\to u}$/$K^+_{v\to u}$) für u. Die LSKPs ($K^-_{u\to v}$/$K^+_{u\to v}$) und ($K^-_{v\to u}$/$K^+_{v\to u}$) kommen exklusiv zur verschlüsselten Kommunikation und Bereitstellung persönlicher Inhalte zwischen u und v zum Einsatz. Aus Sicht von u definiert sich eine Freundschaft mit v über die drei LSKs $K^+_{u\to v}$, $K^-_{u\to v}$ und $K^+_{v\to u}$. Im folgenden Kapitel wird die Zuordnung der LSKs und deren Einsatz im Detail erklärt.
Die Abbildung einer Freundschaft auf zwei LSKPs geht mit einer erhöhten Komplexität bei der Verwaltung der Schlüssel einher. Da sich jede Freundschaft durch drei Schlüssel definiert, muss ein Nutzer u für seine $n = |\Gamma(u)|$ Freunde $3 * n$ anstelle der üblichen $n + 2$ Schlüssel verwalten.
Dieser Umstand wird bewusst in Kauf genommen. Zum einen beugt die Verwen-

dung der LSKPs Deanonymisierungsattacken vor. Jedes Paar von LSKPs kommt zur Kommunikation innerhalb genau einer Freundschaft zum Einsatz. Damit ist die Deanonymisierung des sozialen Graphen deutlich erschwert. Zum anderen reduziert sich das Zurückziehen von Schlüsseln auf das Löschen des kompromittierten LSKPs. Da jeder öffentliche LSK von genau einem Freund verwendet wird, müssen kompromittierte LSKPs nicht aufwendig in Sperrlisten verwaltet werden.

3.3.3 Kommunikation

Unabhängig von der Implementierung eines Exchangers ermöglichen LSKPs die anonyme und sichere Zustellung von Nachrichten. Im Folgenden wird der Einsatz der LSKPs bei der Kommunikation zweier Nutzer erläutert.

3.3.3.1 Nachrichtenformat

Zur Kommunikation zwischen zwei Nutzern kommen in Vegas *Locagramme* [214] zum Einsatz. Ein Locagram besteht aus einem *Identifikator*, einem *Inhalt* und einer *Signatur*. In Anlehnung an die Struktur eines Locagrams zeigt Abbildung 3.3 den Aufbau einer Nachricht in Vegas. Will ein Nutzer v seinem Freund u eine Nachricht m zukommen lassen, dann setzen sich die drei Bestandteile wie folgt zusammen.

Identifikator Beim Identifikator handelt es sich um den Hash-Wert des LSKs $K^+_{u\to v}$. Der Hash-Wert wird mit Hilfe einer geeigneten Hash-Funktion H erzeugt und dient der Zuordnung der Nachricht zum Empfänger u. Da u einen LSK $K^+_{u\to v}$ für jeden seiner Freunde generiert, ist die Anonymität des Empfängers u auch dann gewährleistet, wenn die Nachricht über einen ungesicherten Kommunikationskanal übertragen wird.

Inhalt Beim Inhalt handelt es sich um die auf der Basis des LSKs $K^+_{u\to v}$ verschlüsselte Nachricht m. Die Verschlüsselung gewährleistet den exklusiven und autorisierten Zugriff durch u. Umfangreiche Datenmengen werden zunächst mit einem symmetrischen Verfahren verschlüsselt. Der verwendete Schlüssel wird mit $K^+_{u\to v}$ verschlüsselt und der verschlüsselten Nachricht angehängt. Das Verfahren entspricht der Vorgehensweise von PGP. Für kleine Datenmengen kann m auch direkt mit $K^+_{u\to v}$ verschlüsselt werden. Unabhängig von der eingesetzten Variante wird im Folgenden von der Verschlüsselung „auf der Basis eines LSKs“ gesprochen.

Signatur Die Signatur dient der Sicherstellung der Nachrichtenintegrität und der Authentizität von v. Es handelt sich um den auf der Basis des LSKs $K^-_{v\to u}$ verschlüsselten Hash-Wert von m. Wieder kommt die Hash-Funktion H zum

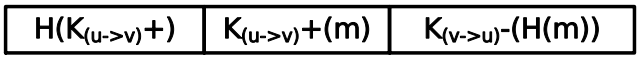


Abbildung 3.3: Aufbau einer Nachricht in Vegas.

Einsatz. Da lediglich u den LSK $K^{+}_{v \to u}$ kennt, ist die Anonymität des Senders v auch dann gewährleistet, wenn die Nachricht über einen ungesicherten Kommunikationskanal übertragen wird.

3.3.3.2 Nachrichtenaustausch

Abbildung 3.4 veranschaulicht die Kommunikationsbeziehung zwischen zwei Freunden u und v. Es kommen die zwei LSKPs ($K^{-}_{u \to v}/K^{+}_{u \to v}$) und ($K^{-}_{v \to u}/K^{+}_{v \to u}$) zum Einsatz. Aus Sicht von u definiert sich die Freundschaft mit v über die drei LSKs $K^{+}_{u \to v}$, $K^{-}_{u \to v}$ und $K^{+}_{v \to u}$.
Der Nachrichtenaustausch selbst erfolgt über die Exchanger EX_v bzw. EX_u. Die Funktionalität eines Exchangers gleicht der einer Mail-Box. Eine Nachricht wird so lange durch einen Exchanger vorgehalten, bis sie an den Empfänger ausgeliefert werden konnte. Jeder Exchanger wird durch eine eindeutige Adresse identifiziert.
Schließen zwei Nutzer in Vegas eine Freundschaft, so teilen sie sich gegenseitig mindestens eine Exchanger-Adresse mit. Um den Grad der Anonymität zu erhöhen, kann ein Nutzer jedem seiner Freunde eine andere Exchanger-Adresse zur Verfügung stellen (vgl. Kap. 3.3.1). Einem Freund können auch mehrere unterschiedliche Adressen bereitgestellt werden, die er beim Senden einer Nachricht z.B. zufällig alterniert.
Da jede Nachricht den Empfänger kodiert als Identifikator enthält, können Nachrichten über einen beliebigen Kommunikationskanal übertragen werden. Synchrone und asynchrone sowie push- und pull-basierte Implementierungen sind möglich.
Neben Protokollen wie XMPP [184], IRC [171] und SMTP [175] in Kombination mit POP3 [153] oder IMAP [53] können als Exchanger auch SMS oder Microblogging-Dienste wie Twitter zum Einsatz kommen. Auch die Übertragung über WLAN oder Bluetooth auf der Basis eines proprietären Ad-Hoc-Protokolls oder das Abfotografieren eines QR-Codes sind denkbar.

3.3.4 Datenspeicherung

Um dem eigenen Freundeskreis persönliche Inhalte zur Verfügung zu stellen, werden in Vegas Datastores eingesetzt. Der autorisierte Zugriff eines Nutzers u auf die persönlichen Inhalte eines Freundes v wird durch die Verschlüsselung der Inhalte auf der Basis des öffentlichen LSKs $K^{+}_{v \to u}$ sichergestellt. Die Umsetzung eines Datastores muss lediglich der Anforderung genügen, dass persönliche Inhalte über

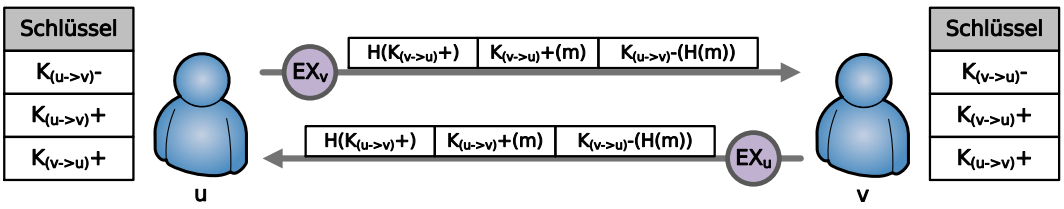


Abbildung 3.4: Kommunikationsbeziehung zweier Nutzer in Vegas.

einen global gültigen *einheitlichen Quellenanzeiger* (*URL*, engl: *uniform resource locator*) referenziert und abgerufen werden können. Zur Gewährleistung des permanenten Profilzugriffs muss die Implementierung eines Datastores eine hohe Verfügbarkeit garantieren. Denkbar sind FTP-Server [176], WebDAV-Server [67] oder CCIs wie Amazon S3 [270] oder Dropbox [244].

3.3.4.1 Adressierungsschema

Damit Freunde auf das eigene Profil und andere persönliche Inhalte zugreifen können, verwaltet jeder Nutzer u für jeden Freund v eine *Indexdatei* $I_{u \to v}$. Diese wird auf der Basis des LSKs $K^{+}_{v \to u}$ verschlüsselt und auf einem Datastore DS_u abgelegt. Schließen zwei Nutzer in Vegas eine Freundschaft, teilen sie sich gegenseitig die URLs zu ihren Indexdateien mit.
Eine Indexdatei setzt sich aus *Einträgen* und *Ressourcen* zusammen.

Einträge Bei einem Eintrag handelt es sich um einfache Profilinformationen eines Nutzers. Dazu zählen sein Name, sein Alter oder sein Geschlecht. Einträge werden in Form von Schlüssel/Wert-Paaren wie z.B. `Alter = 37` in der Indexdatei hinterlegt. Eine Indexdatei kann beliebig viele solcher Einträge enthalten. Die Namen der Schlüssel sind frei wählbar.

Ressourcen Für komplexere Inhalte steht eine Liste von Ressourcen zur Verfügung. Bei Ressourcen handelt es sich z.B. um Bilder, Videos oder auch um eine persönliche Pinnwand. Ressourcen können rekursive aufgebaut sein. Ein Beispiel dafür wäre eine als Liste von Ressourcen implementierte Pinnwand. Jeder einzelne Pinnwandeintrag stellt wieder eine Ressource dar.

Aufgrund der unterschiedlichen Eigenschaften wie z.B. dem Datenvolumen multimedialer Inhalte existieren in Vegas verschiedene Möglichkeiten zur Einbindung. Ressourcen können

a) direkt in der Indexdatei z.B. kodiert als Base64-Zeichenkette [31],

b) als Verknüpfung mit einer unverschlüsselten Datei,

c) als Verknüpfung mit einer auf der Basis des öffentlichen LSKs eines Freundes verschlüsselten Datei oder

d) als Verknüpfung mit einer symmetrisch verschlüsselten Datei

gespeichert werden. Um die Indexdatei schlank zu halten, sollte Variante a) nur für kleine Datenobjekte verwendet werden. Variante b) bietet sich an, falls eine externe öffentlich zugängliche Ressource wie z.B. das Fotoalbum eines bekannten Künstlers eingebunden werden soll. Diese Variante macht insbesondere dann Sinn, wenn ein Nutzer in der Öffentlichkeit nicht mit dem referenzierten Inhalt in Verbindung gebracht werden möchte.

Soll eine Ressource nur für einen einzigen Freund zur Verfügung stehen, eignet sich Variante c). Variante d) bietet sich an, wenn eine Ressource für einen eingegrenzten Freundeskreis zugänglich gemacht werden soll. Ein Beispiel

dafür wäre die persönliche Pinnwand. Der verwendete symmetrische Schlüssel kann selbst wieder als Ressource über Variante c) eingebunden werden.

Für den Zugriff von Nutzer u auf das Profil eines Freundes v wird zunächst die Indexdatei $I_{v \to u}$ geladen. Will u über Verknüpfungen eingebundene Ressourcen abrufen, öffnet u die entsprechende URL und entschlüsselt den referenzierten Inhalt. Handelt es sich beim Inhalt wieder um eine Verknüpfung, wird dieser Schritt wiederholt.
Mit diesem Mechanismus kann ein Nutzer flexibel die persönlichen Inhalte seiner Freunde browsen.

3.3.4.2 Verteilte Speicherung

Ebenso wie mehrere verschiedene Exchanger für einen oder mehrere Freunde eingesetzt werden können, hat ein Nutzer die Möglichkeit, seine persönlichen Informationen durch die Verwendung eines oder mehrerer Datastores verteilt und selektiv zur Verfügung zu stellen. Zum einen behält ein Nutzer dadurch sein Recht auf informationelle Selbstbestimmung, zum anderen erlaubt dies eine strikte Trennung sensibler und anderer persönlicher Inhalte, falls z.B. rechtliche Bestimmungen oder gesetzlichen Regelungen einen Nutzer dazu zwingen.
Das Konzept der Datastores erlaubt auch die Integration anderer sozialer Plattformen. Beispielsweise lassen sich OSNs wie Flickr oder Last.fm zur Bereitstellung öffentlich zugänglicher Fotos oder Musikdateien als Datastores in Vegas integrieren. Für die Publikation persönlicher Inhalte kann ein anderer Dienst wie z.B. ein privat administrierter FTP-Server eingesetzt werden.
Um den Zugriff auf einen ausgewählten Freundeskreis zu beschränken, werden die Inhalte auf der Basis der zugehörigen LSKs verschlüsselt abgelegt. Über die Exchanger-Domäne kann ein Nutzer seine Freunde jederzeit über Veränderungen in der Datastore-Domäne unterrichten. Ändert Nutzer u z.B. Einträge der Indexdatei $I_{u \to v}$ oder beschließt u, $I_{u \to v}$ auf einen anderen Datastore zu verschieben, kann u den Freund v auf dem Exchanger EX_v darüber informieren.

3.3.5 Ausbildung von Freundschaften

Aufgrund der Anforderung an ein starkes Vertrauensverhältnis zwischen Freunden (vgl. Kap. 3.1.2) existiert in Vegas keinerlei Unterstützung für die Suche nach anderen Nutzern. Das Browsen des sozialen Graphen wird nicht unterstützt. Die Möglichkeiten, anderen Nutzern eine Freundschaftsanfrage zukommen zu lassen, sind somit stark limitiert.
Im Folgenden werden die verschiedenen Varianten zur Ausbildung einer Freundschaft vorgestellt. Zuvor werden einige Begrifflichkeiten erläutert, die für den Austausch von Nachrichten relevant sind.

3.3.5.1 Austausch von Nachrichten

In Vegas definiert sich ein *sicherer Kommunikationskanal* zwischen zwei Freunden über den Besitz und die Verwendung ihrer LSKPs (vgl. Kap. 3.3.3.2). Neben der Nachrichtenintegrität und der Authentizität des Kommunikationspartners stellt die Verwendung der LSKPs den autorisierten Zugriff auf die ausgetauschten Inhalte sicher. In Vegas existiert ein sicherer Kommunikationskanal erst dann, wenn zwei Nutzer befreundet sind.
Wollen zwei Nutzer eine neue Freundschaft eingehen, müssen bestimmte Informationen über einen *unsicheren Kanal* ausgetauscht werden. Im Kontext von Vegas handelt es sich um einen Kanal, der nicht durch die Verwendung der LSKPs der beiden Nutzer abgesichert ist.
Neben der Differenzierung zwischen sicheren und unsicheren Kommunikationskanälen wird zwischen *In Band* (*IB*) und *Außer Band* (*OOB*, engl: *out of band*) Nachrichten unterschieden.

OOB-Nachrichten OOB-Nachrichten werden nicht über einen Exchanger übertragen. Sie werden lediglich beim Aufbau einer Freundschaft versendet. Die gesamte Nachricht wird nicht auf der Basis öffentlicher LSKs verschlüsselt. Einzelne Teile des Inhalts können aber sehr wohl auf der Basis eines öffentlichen LSKs verschlüsselt sein.

IB-Nachrichten IB-Nachrichten werden über einen Exchanger übertragen. Sie werden auf der Basis eines öffentlichen LSKs komplett verschlüsselt. IB-Nachrichten werden durch eine *Bestätigung* (*ACK*, engl: *acknowledgement*) quittiert.

Im Folgenden werden auch die Begriffe *IB-* bzw. *OOB-Kanal* verwendet, wenn über einen Kanal IB- bzw. OOB-Nachrichten ausgetauscht werden.

3.3.5.2 Das Friendship-Protokoll

Grundlage für das Schließen einer Freundschaft ist das *Befreundungsprotokoll* (engl: *friendship protocol*). Die initialen Schritte des Protokolls erfolgen über einen OOB-Kanal.
Im Folgenden wird beim Austausch von Informationen immer vom *Nachrichtenaustausch* gesprochen. Auf welchem Weg Informationen zwischen zwei Nutzern ausgetauscht werden, ist durch das Protokoll jedoch nicht weiter spezifiziert. Handelt es sich bei den ausgetauschten Informationen um einfache Textdateien, können diese auf beliebigem Wege z.B. per E-Mail oder durch das Abfotografieren eines QR-Codes übertragen werden. Grundvoraussetzung für das Befreunden zweier Nutzer über einen OOB-Kanal ist die Kenntnis eines gemeinsamen Geheimnisses sec.
Das Schließen einer Freundschaft ist in Abbildung 3.5 dargestellt und beinhaltet folgende Schritte:

a) u generiert zunächst ein LSKP ($K^{+}_{u^{(1)}\to v}$/$K^{-}_{u^{(1)}\to v}$). Aus dem öffentlichen LSK erzeugt u eine *Freundschaftsanfrage* $FReq$ (*FReq*, engl: *friendship request*). Mit

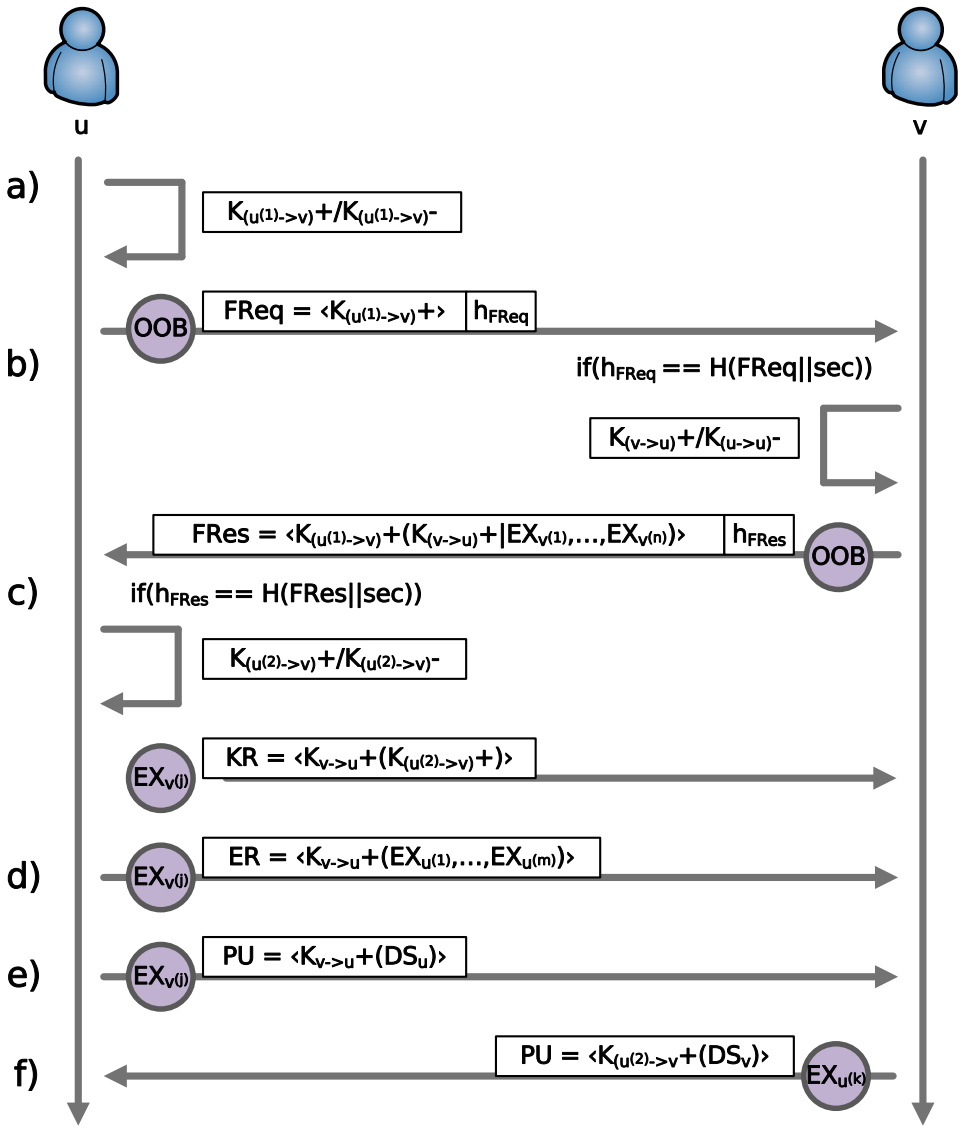


Abbildung 3.5: Ausbildung einer Freundschaft über das Friendship-Protokoll.

Hilfe einer kryptographischen Hash-Funktion [193] H und dem gemeinsamen Geheimnis sec erzeugt u den *Nachrichtenauthentifizierungscode* (***HMAC***, engl: *keyed hash message authentication code* [119]) $h_{FReq} = H(Freq||sec)$. Diesen überträgt u zusammen mit der Freundschaftsanfrage über einen OOB-Kanal an v.

b) Sind die Integrität und die Authentizität der Freundschaftsanfrage verifiziert, erzeugt v das LSKP ($K^+_{v\to u}$/$K^-_{v\to u}$). Mittels $K^+_{u^{(1)}\to v}$ verschlüsselt v den öffentlichen LSK $K^+_{v\to u}$ zusammen mit einer beliebigen Anzahl von Exchanger-Adressen $EX_{v^{(1)}}, \ldots, EX_{v^{(n)}}$ und erzeugt daraus die *Freundschaftsantwort* $FRes$ (***FRes***, engl: *friendship response*). Zusammen mit dem HMAC $h_{FRes} = H(Fres||sec)$ überträgt v die Freundschaftsantwort über einen OOB-Kanal an u.

c) Mittels $K^-_{u^{(1)}\to v}$ entschlüsselt u den Inhalt von $FRes$. Über den beigefügten HMAC h_{FRes} kann u die Integrität der Freundschaftsanfrage $FRes$ und die Authentizität von v sicherstellen.

   Da der Schlüssel $K^+_{u^{(1)}\to v}$ in Schritt a) unverschlüsselt übertragen wird, muss u diesen vor Bekanntgabe seiner Exchanger-Adressen mit einer *Schlüsselerneuerung* KR (***KR***, engl: *key refresh*) ersetzen. Ein Angreifer mit Zugriff auf den OOB-Kanal hätte sonst die Möglichkeit, nachfolgende Nachrichten von v an u über den verwendeten Identifikator zu deanonymisieren.

   u erzeugt ein neues LSKP ($K^+_{u^{(2)}\to v}$/$K^-_{u^{(2)}\to v}$). Mittels $K^+_{v\to u}$ verschlüsselt u den öffentlichen LSK, erzeugt daraus KR und überträgt KR über einen IB-Kanal an v. Dabei kommt ein Exchanger $EX_{v^{(j)}} \in \{EX_{v^{(1)}}, \ldots, EX_{v^{(n)}}\}$ zum Einsatz.

d) Bisher kennt v noch keinen Exchanger von u. u verschlüsselt daher eine beliebige Anzahl von Exchanger-Adressen $EX_{u^{(1)}}, \ldots, EX_{u^{(m)}}$ auf der Basis des LSK $K^+_{v\to u}$, erzeugt daraus eine *Exchanger-Erneuerung* ER (***ER***, engl: *exchanger refresh*) und überträgt ER über $EX_{v^{(j)}}$ an v.

e) Abschließend müssen sich u und v jeweils die URL zur Indexdatei $I_{u\to v}$ bzw. $I_{v\to u}$ in ihrer Datastore-Domäne mitteilen. Dazu verschlüsselt u die entsprechende Adresse DS_u auf der Basis des LSK $K^+_{v\to u}$, erzeugt daraus eine *Profilerneuerung* PU (***PU***, engl: *profile update*) und überträgt PU über $EX_{v^{(j)}}$ an v.

f) Nachdem v die Schlüsselerneuerung von u erhalten hat (vgl. Schritt c)), führt auch v die Profilerneuerung für u durch. Dieser Schritt erfolgt unabhängig von den Schritten d) und e).

Um Replay-Attacken entgegenzuwirken, verwendet das Friendship-Protokoll für jede Nachricht ein zufällig generiertes Nonce. Zudem können Freundschaftsanfragen mit einem Zeitstempel versehen werden. Dieser definiert die maximale Gültigkeitsdauer einer Freundschaftsanfrage. Aus Gründen der Übersichtlichkeit sind diese Elemente in Abbildung 3.5 nicht visualisiert. Quittierungen der IB-Nachrichten, Identifikatoren der Absender und die notwendigen HMACs sind ebenfalls nicht dargestellt.

3.3.5.3 Das Coupling-Protokoll

Das Konzept von Vegas schließt das Browsen anderer Nutzer explizit aus. Um die Möglichkeiten zum Schließen einer Freundschaft zu erhöhen, stellt Vegas das *Verkuppelungsprotokoll* (engl: *coupling protocol*) zur Verfügung.
In RSNs ergeben sich neue Freundschaften oft in Folge eines sich gegenseitigen Kennenlernens über einen gemeinsamen Bekannten. Das Coupling-Protokoll ist der Versuch, diese Situation für das OSN Vegas zu modellieren.
Der Einsatz des Coupling-Protokolls bietet sich an, wenn ein Nutzer zwei Freunde besitzt, die untereinander noch keine gemeinsame Beziehung unterhalten. Dem Ansatz liegt die Annahme zugrunde, dass zwei Freunde v und w des Nutzers u mit hoher Wahrscheinlichkeit einer gegenseitigen Freundschaft zustimmen, wenn u gegenüber w und v eine wechselseitige Empfehlung ausspricht. Die Wahrscheinlichkeit für die Annahme einer Empfehlung hängt vom Grad des Vertrauens zwischen u und v bzw. u und w ab.
Abbildung 3.6 veranschaulicht das Coupling-Protokoll am Beispiel der drei Nutzer u, v und w mit $\{v, w\} \in \Gamma(u)$ und $(v, w) \notin E$. Beim Coupling-Protokoll werden alle Nachrichten über sichere IB-Kanäle ausgetauscht. Die folgenden Schritte werden dazu ausgeführt.

a) u sendet eine *Verkuppelungsanfrage* (*CR*, engl: *coupling request*) an v und w. Der Inhalt PI_w bzw. PI_v ist nicht näher spezifiziert. Es handelt sich dabei z.B. um eine Liste der von v bzw. w als öffentlich einsehbar deklarierten Profilattribute in Verbindung mit einer persönlichen Nachricht. Die CR-Nachrichten werden auf der Basis der öffentlichen LSKs von v bzw. w verschlüsselt.

b) Ist v an einer Freundschaft mit w interessiert, beantwortet v die Verkuppelungsanfrage mit einer *Verkuppelungsannahme* (*CA*, engl: *coupling accept*). Dazu generiert v ein neues LSKP $K^+_{v^{(1)}\to w}/K^-_{v^{(1)}\to w}$ und sendet den öffentlichen LSK $K^+_{v^{(1)}\to w}$ verschlüsselt auf der Basis des öffentlichen LSKs $K^+_{u\to v}$ zurück an u. Ist w ebenfalls an einer Freundschaft interessiert, vollzieht w analog die entsprechend notwendigen Schritte.

c) u entschlüsselt die CA von v und leitet sie verschlüsselt auf der Basis des öffentlichen LSKs $K^+_{w\to u}$ an w weiter. Analog verfährt u mit der CA von w.

d) Um im späteren Verlauf der Kommunikation zwischen v und w einer Deanonymisierung durch u vorzubeugen, führt w nach Erhalt der CA-Nachricht eine Schlüsselerneuerung durch. Dazu generiert w ein neues LSKP $K^+_{w^{(2)}\to v}/K^-_{w^{(2)}\to v}$ und übermittelt den öffentlichen LSK als KR-Nachricht verschlüsselt auf der Basis des LSKs $K^+_{v^{(1)}\to w}$ an v. v verfährt entsprechend analog.

e) Um Deanonymisierungsattacken vorzubeugen, können v und w optional auch die Exchanger-Adressen erneuern. Dazu sendet w eine neue Exchanger-Adresse $EX_{w^{(2)}}$ als ER-Nachricht verschlüsselt auf der Basis des LSKs $K^+_{v^{(2)}\to w}$ an v. v verfährt analog.

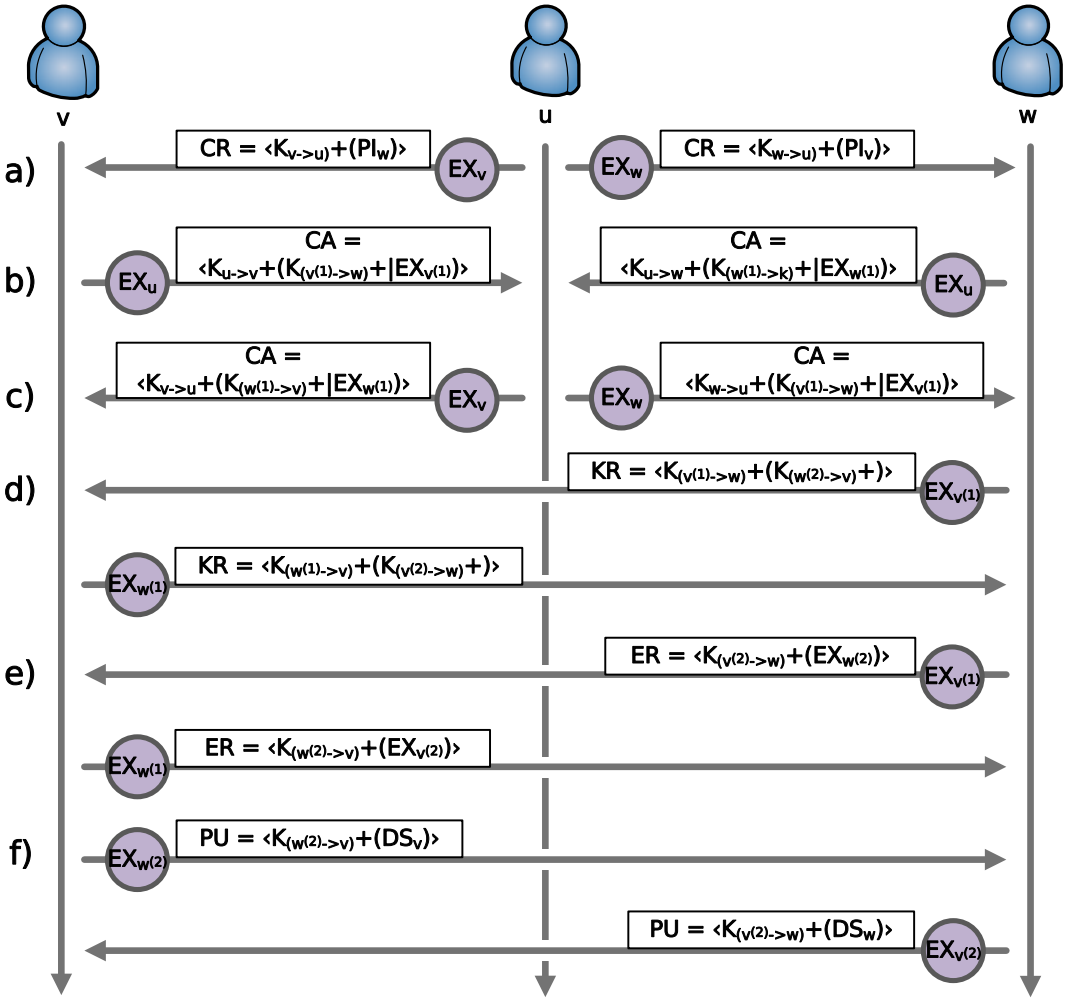


Abbildung 3.6: Kommunikationsverlauf beim Aufbau einer Freundschaft über das Coupling-Protokoll.

f) Zuletzt tauschen v und w über PU-Nachrichten jeweils den Pfad zu ihren Indexdateien in Form der Adressen DS_v bzw. DS_w aus. Erzeugung und Versand der Nachrichten erfolgen in Analogie zum Friendship-Protokoll.

Es liegt an der Konstruktion dieses Verfahrens, dass u von Anfang an einen *Man-In-The-Middle* (*MITM*) darstellt. Aufgrund der starken Vertrauensbeziehung zwischen u und w bzw. u und v wird eine potentielle MITM-Attacke jedoch nicht als Sicherheitslücke angesehen.

Das Problem lässt sich beseitigen, indem man von v und w in Schritt d) die Sicherstellung der Nachrichtenintegrität und der Authentizität der Kommunikationspartner verlangt. In Analogie zum Friendship-Protokoll kann dazu ein HMAC (vgl. Kap. 3.3.5.2) eingesetzt werden. Das notwendige gemeinsame Geheimnis von v und w muss über einen OOB-Kanal ausgetauscht werden.

3.3.6 Datensynchronisation

Damit ein Nutzer mit mehreren verschiedenen Endgeräten auf Vegas zugreifen kann, müssen sich die einzelnen Clients untereinander synchronisieren. Der Parallelbetrieb setzt eine dezentrale Koordination der Clients in Bezug auf den Austausch von Nachrichten und den Zugriff auf persönliche Inhalte voraus.
Von der konkreten Implementierung eines Datastores wird lediglich verlangt, dass jedes Datum über eine global gültige URL referenziert und abgerufen werden kann (vgl. Kap. 3.3.4). Abgesehen von der Sicherstellung atomarer Schreibzugriffe dürfen keine weiteren Annahmen über eine serverseitige Unterstützung der Datensynchronisation durch einen Datastore getroffen werden.
Für Vegas wird vereinfachend angenommen, dass zu einem Zeitpunkt immer nur ein Client eines Nutzers mit dem OSN interagiert. Generell sollte das zu synchronisierende Datenvolumen möglichst gering ausfallen. Daher werden nur Datenstrukturen zur Administration des eigenen Nutzerprofils in der Datastore-Domäne hinterlegt. Bei den Datenstrukturen handelt es sich z.B. um die Liste der Freunde und deren Zugriffsrechte auf die persönlichen Inhalte eines Nutzers.
Alle zu synchronisierenden Datenstrukturen eines Nutzers u werden als *Administrationsdateien* in der Datestore-Domäne von u hinterlegt. Um den autorisierten Zugriff durch u sicherzustellen, werden alle Administrationsdateien mit einem passwortgeschützten symmetrischen Schlüssel K^u_{AD} verschlüsselt. Zusätzlich wird eine *Client-Datei* mit einer Liste aller registrierten Clients von Nutzer u sowohl lokal auf jedem Client als auch in der Datastore-Domäne gespeichert.
Will ein Nutzer mit einem weiteren Endgerät über Vegas kommunizieren, muss er dieses als Client mit einem neuen IB-Kanal registrieren. Dazu fügt der Nutzer der Client-Datei einen öffentlichen LSK und eine Exchanger-Adresse hinzu. Der IB-Kanal dient der exklusiven Kommunikation zwischen bereits registrierten Clients und dem neuen Client des Nutzers.
Für den Zugriff auf die eigene Datastore-Domäne müssen einem neuen Client die URLs zur Client- und zu den Administrationsdateien mitgeteilt werden. Dies erfolgt zusammen mit dem symmetrischen Schlüssel K^u_{AD} über einen OOB-Kanal. Der neue Client kann fortan die Client- und Administrationsdateien vom Datastore abrufen, sich in die Client-Datei eintragen und mit der Version der Datastore-Domäne synchronisieren.
Bei jedem Start synchronisiert ein Client seine lokalen und die im Datastore gespeicherten Client- und Administrationsdateien. Lokal auf einem Client modifizierte Administrationsdateien werden als neue Version in der Datastore-Domäne hinterlegt. Veränderungen werden über die jeweils exklusiven IB-Kanäle allen anderen Clients eines Nutzers unmittelbar mitgeteilt. Der Empfang einer entsprechenden Mitteilung initiiert bei einem Client die Synchronisation mit der Datastore-Domäne.

3.3.7 Gemeinsame Nachbarn

Etablierte OSNs wie Facebook und Google+ bieten ihren Nutzern neben den üblichen Basisfunktionen (vgl. Kap. 2.3.1) ein weitreichendes Angebot an Erweiterun-

gen ihrer Dienste. Dazu zählen die automatisierte Empfehlung neuer Freundschaften oder die Suche nach Informationen auf der Basis des sozialen Graphen. Vegas hingegen ordnet solche Erweiterungen der Sicherheit und dem Schutz der Privatsphäre seiner Nutzer unter. Viele gewohnte Funktionen sind in Vegas daher nicht verfügbar.

Eine dieser Funktionen stellt die Identifikation *gemeinsamer Nachbarn* $GN_{uv} = \{w : w \in \Gamma(u) \cap \Gamma(v)\}$ (vgl. Kap. 2.5.1.4) zweier Nutzer u und v dar. Mit einem entsprechenden Mechanismus könnte ein Nutzer u z.B. automatisch von seinem Freund v informiert werden, falls v eine neue Beziehung mit einem Nutzer $w \in \Gamma(u)$ etabliert.

Denkbar ist auch eine Art *vertrauenswürdiger Verzeichnisdienstes* (engl: *directory buddy*), dessen Angebot auf den Beziehungen zwischen seinen Nutzern basiert. Ein Directory Buddy u könnte als vertrauenswürdiger Dienstvermittler zwischen seinen Nutzern v und w ($v, w \in \Gamma(u)$) fungieren, wobei v und w nicht zwangsläufig eine gemeinsame Freundschaft pflegen müssen.

Aufgrund der Vielzahl nützlicher Einsatzgebiete soll auch Vegas einen Mechanismus bereitstellen, der die Identifikation gemeinsamer Nachbarn unterstützt. Selbst wenn zwei Nutzer u und v bereits befreundet sind, bedingt ein solcher Mechanismus jedoch die Offenlegung persönlicher Informationen. Haben u und v einen gemeinsamen Nachbarn w, dann müssen u und v die Information über ihre wechselseitige Freundschaft mit w explizit freigeben. Da das Identifizieren gemeinsamer Nachbarn das Recht auf informationelle Selbstbestimmung verletzt, wird es dem Nutzer freigestellt, den folgenden Mechanismus zu verwenden.

Um den Austausch persönlicher Informationen zu minimieren, kommen in Vegas *Relationship Hashes* zum Einsatz. Ein Relationship Hash definiert sich als Hash-Wert über der XOR-Kombination der öffentlichen LSKs zweier Freunde. Für die Freunde u und v berechnet sich der Relationship Hash R_{uv} nach der Funktion $R_{uv} = H(K^+_{u\to v} \oplus K^+_{v\to u})$, wobei H eine geeignete Hash-Funktion und $\oplus$ die binäre XOR-Operation repräsentieren. Es gilt insbesondere $R_{uv} = R_{vu}$.

Das folgende Beispiel skizziert den Ablauf der Identifikation gemeinsamer Nachbarn: Seien u, v und w drei untereinander befreundete Nutzer mit $\{(u, v), (u, w), (v, w)\} \in E$. Beispielsweise über seine Datastore-Domäne stellt u dem Freund w eine Liste seiner Relationship Hashes $LR_u = \{R_{ux_i} : x_i \in \Gamma(u)\}$ zur Verfügung. Zu diesem Zeitpunkt kann w über $|LR_u|$ lediglich auf die Gesamtanzahl der Freunde von u schließen. Will u diese Information anonymisieren, kann u optional eine beliebige Anzahl zufällig erzeugter Relationship Hashes der Liste LR_u hinzufügen. Will v ebenfalls seine gemeinsamen Freunde w mitteilen, so stellt auch v seine Liste von Relationship Hashes $LR_v = \{R_{vy_i} : y_i \in \Gamma(v)\}$ zur Verfügung. w kann über die Schnittmenge der beiden Listen auf die gemeinsame Freundschaft von v und u schließen ($LR_u \cap LR_v \neq \emptyset$).

In dieser sehr vereinfachten Form können Relationship Hashes dazu missbraucht werden, Freundschaften mit Freunden zweiter Ordnung vorzutäuschen. Beispielsweise kann w die manipulierte Liste $LR'_w = LR_u \cup \{R_{wl}\}$ an einen Freund $l \in \Gamma(w)$ ($l \notin \Gamma(u)$) weiterleiten. Pflegt l eine Freundschaft mit einem Knoten $k \in \Gamma(u)$

($k \neq w$), den w nicht kennt ($k \notin \Gamma(w)$) und hat l Zugriff auf die Relationship Hashes LR_k von k, dann erkennt l eine Freundschaft zwischen w und k, die nicht existiert ($LR'_w \cap LR_k \neq \emptyset$). Das Täuschungsmanöver gelingt natürlich nur unter der Voraussetzung, dass sowohl k als auch u den entsprechenden Relationship Hash R_{ku} den Knoten l bzw. w zuvor mitgeteilt haben.
Für Vegas macht dieses Szenario jedoch wenig Sinn, da Freundschaften per Definition auf der Basis einer starken Vertrauensbeziehung entstehen.
Als Beispiel für den Einsatz eines Directory Buddies in Kombination mit Relationship Hashes wurde im Rahmen dieser Arbeit ein kontextabhängiger Visitenkarten-Browser implementiert [294]. Der Prototyp realisiert einen sicheren und anonymen Austausch digitaler Visitenkarten über ein drahtloses Ad-Hoc-Netzwerke zwischen den Teilnehmern einer Konferenz.

3.4 Prototypische Umsetzung

Das Konzept von Vegas wurde für das Betriebssystem Android [234] als mobile Anwendung implementiert. Der Prototyp *Vegas Mobile* [277] steht im *Google Play Store* [259] zum Download zur Verfügung.
Um technisch weniger versierten Nutzern den Einstieg zu ermöglichen, wird Vegas Mobile mit einer fertigen Startkonfiguration ausgeliefert. Als Exchanger kommt der *ejabberd* XMPP-Server [269], als Datastore der *vsftpd* FTP-Server [278] zum Einsatz. Beide Instanzen laufen auf Rechnern des Instituts für Informatik der Ludwig-Maximilians-Universität München.
Der Prototyp dient dazu, das Interaktionsverhalten der Nutzer und den sozialen Graphen von Vegas zu studieren. Im Widerspruch zu den Anforderungen an ein sicheres und die Privatsphäre schützendes OSN wurde Vegas Mobile mit einem Logging-Framework versehen. Das Logging ist optional und für eine finale Softwarelösung nicht vorgesehen.
Bisher ist der Nutzerkreis von Vegas Mobile zu klein, um belastbare Aussagen über das OSN treffen zu können. Im Rahmen dieser Arbeit wird auf eine weitere Diskussion der gesammelten Daten verzichtet.

3.5 Zusammenfassung

Der Missbrauch persönlicher Inhalte durch Angreifer und Betreiber verdeutlicht, dass der Einsatz ausgereifter Privatsphäreeinstellungen weder die notwendige Sicherheit noch einen ausreichenden Schutz der Privatsphäre der Nutzer garantieren kann. Ein hinreichender Schutz erfordert die Dezentralisierung eines OSNs, wodurch sich jedoch andere Einschränkungen bzw. Probleme ergeben.
In Kombination mit der Informationssicherheit muss ein dezentrales OSN zusätzliche Anforderungen erfüllen. Neben der Gewährleistung des Rechts auf informationelle Selbstbestimmung und der Voraussetzung starker Vertrauensbeziehungen für das Ausbilden einer Freundschaft, wurden als Anforderungen der permanente

Zugriff auf Profilinformationen sowie die hinreichende Unterstützung der Mobilität identifiziert.
Die detaillierte Betrachtung von Persona, Safebook, PeerSoN und Vis-à-Vis hat gezeigt, dass keines dieser dezentralen OSNs allen Anforderungen genügt. Infolge dessen wurde das OSN Vegas mit seiner Unterteilung in Client-, Exchanger- und Datastore-Domäne entwickelt.
Während die Client-Domäne die Integration mehrere Endgeräte erlaubt, sorgt die Exchanger-Domäne für den sicheren, anonymen und protokollunabhängigen Austausch von Nachrichten. Die Datastore-Domäne gewährleistet den permanenten Zugriff auf persönliche Inhalte und gleichzeitig die notwendige Unterstützung mobiler Teilnehmer.
Das Konzept verbindungsspezifischer Schlüsselpaare stellt die Grundlage für den Schutz des Rechts auf informationelle Selbstbestimmung der Teilnehmer dar. Durch den flexiblen Einsatz einer unbeschränkten Anzahl von Exchangern und Datastores in Kombination mit der Verwendung verbindungsspezifischer Schlüsselpaare steht es jedem Nutzer frei, den Grad seiner Anonymität im OSN zu erhöhen.
Vegas setzt für die Ausbildung einer Freundschaft die Existenz einer realen Beziehung voraus. Technisch wird das Schließen einer Freundschaft über das Friendship-Protokoll realisiert. Da sich dessen Einsatz auf das reale Egonetzwerk einer Person beschränkt, erfüllt Vegas auch die Anforderung an starke Vertrauensbeziehungen virtueller Kontakte.
Um die Möglichkeiten zum Schließen einer Freundschaft weiter zu erhöhen, steht das Coupling-Protokoll zur Verfügung. Es bildet den realen Prozess des Kennenlernens neuer Freunde über einen gemeinsamen Bekannten in den virtuellen Raum von Vegas ab.
Optional können Nutzer Relationship Hashes ihrer Beziehungen publizieren. Mit minimaler Preisgabe persönlicher Informationen ist es damit möglich, gemeinsame Kontakte mit der eigenen Nachbarschaft zu identifizieren.
Das folgende Kapitel beschäftigt sich eingehend mit der Priorisierung von Freunden bei der Suche und Verbreitung von Informationen in dezentralen OSNs. Dort zeigt sich auch der konkrete Nutzen und die Relevanz von Relationship Hashes im Falle von Vegas.

4 Informationsverbreitung in dezentralen OSNs

Neben der Kommunikation und der Bereitstellung persönlicher Informationen ist es in zentralisierten OSNs möglich, die Liste seiner eigenen Kontakte aber auch diejenigen anderer Nutzer zu traversieren (vgl. Kap. 2.3.1). In Anbetracht des enormen Informationsgehalts des assoziierten sozialen Graphen erscheint es vielversprechend, ein OSN nicht nur zufällig zu browsen, sondern darin auch gezielt nach Informationen zu suchen. Auch bei der Suche innerhalb anderer Quellen kann die Berücksichtigung sozialer Informationen zu besseren Ergebnissen führen.

Problematisch erweist sich das Suchen innerhalb dezentraler OSNs, welche die Sicherheit und den Schutz der Privatsphäre ihrer Nutzer forcieren. In Vegas führt dies zur Einschränkung jeglicher Kommunikation auf die Mitglieder des eigenen Egonetzwerks. Das Browsen des sozialen Graphen und somit auch die Suche nach Informationen ist ohne entsprechende Anpassungen nicht möglich.

Bei den meisten dezentralen OSNs handelt es sich um P2P-Systeme. Viele Ansätze basieren auf einem strukturierten P2P-Overlay [56, 39, 159, 189]. Solche OSNs profitieren doppelt von dieser Struktur, da Suchverfahren direkt auf Routing-Algorithmen zurückgreifen können, die ohnehin zur Organisation der Nutzer im DHT bereitgestellt werden.

Mit dem Ziel, digitale Verbindungen auf starke Vertrauensbeziehung zu beschränken, unternimmt Vegas den Versuch, reale Kontakte auf die virtuellen Identitäten eines OSN abzubilden. Im Vergleich zu strukturierten P2P-Ansätzen ist es in Vegas nicht möglich, aktiv Einfluss auf die Netzwerktopologie bzw. den sozialen Graphen zu nehmen, um die Weiterleitung von Suchanfragen zu optimieren. Zudem besitzen Nutzer in Vegas keine globale Sicht auf den zugrunde liegenden sozialen Graphen. Neben Suchverfahren strukturierter P2P-Netzwerke scheiden daher auch alle Routing-Strategien aus, die auf der Ausbildung (semantischer) Overlays in unstrukturierten P2P-Netzwerken entwickelt wurden [52, 192, 209, 48, 71, 9].

Dieses Kapitel beschäftigt sich mit der Verbreitung bzw. der Suche nach Informationen in dezentralen OSNs. Nach einem Überblick über den Begriff und die Ausprägungen der sozialen Suche werden Besonderheiten für eine Umsetzung innerhalb dezentraler OSNs mit einem Fokus auf Vegas diskutiert. Anschließend werden zahlreiche Strategien zur Weiterleitung von Suchanfragen in P2P-Netzwerken vorgestellt. Basierend auf einigen ausgewählten Strategien wird ein generischer Algorithmus für das Routing sozialer Suchanfragen in Vegas konzipiert.

Im Mittelpunkt des Kapitels steht die Untersuchung zahlreicher Priorisierungsstra-

tegien für den entwickelten Algorithmus. Die Priorisierungsstrategien werden simulativ auf der Basis künstlich erzeugter und realer OSN-Datensätze evaluiert.
In den folgenden Abschnitten werden die Begriffe Routing und Weiterleitung synonym verwendet.

4.1 Soziale Suche

Die Suchmaschinen von Google [256] und Yahoo [283] zählen zu den am meisten verwendeten Diensten im Internet [233]. Anfangs beruhte die Sortierung der Suchergebnisse auf sehr schlanken Verfahren wie z.B. dem *PageRank*-Algorithmus [124]. Dieser gewichtet eine Menge an Dokumenten lediglich auf der Grundlage der Struktur ihrer wechselseitigen Verlinkung.
Man hat früh erkannt, dass auch die Berücksichtigung sozialer Faktoren bei der Gewichtung einen sehr positiven Effekt auf die individuell empfundene Relevanz der Suchergebnisse haben kann. Die Gewichtung entsprechend der Übereinstimmung sozialer Metadaten mit den Schlagwörtern einer Suchanfrage kann zu einer gesteigerten Qualität der Suchergebnisse führen [18, 95, 186]. Zu solchen Metadaten zählen z.B. Tags, die man mit Diensten wie Delicious [241] generieren kann.
Der positive Effekt zeigt sich auch bei der Berücksichtigung der Inhalte von OSNs wie Facebook oder Twitter. Durch *soziale Erweiterungen* (engl: *social plugins*) [275] ermöglicht Facebook z.B. die Anreicherung von Webangeboten um Informationen des sozialen Graphen. Bookmarks, Bewertungen, Kommentare und Tags für bestimmte Inhalte werden direkt im sozialen Graphen hinterlegt. Diese können an anderer Stelle über das *Open Graph Protocol* [94, 248] abgerufen werden.
Berücksichtigt man beim Durchsuchen einer Menge an Dokumenten neben rein strukturellen Eigenschaften auch den mit der Dokumentenstruktur assoziierten sozialen Graphen, so spricht man weitläufig auch von *sozialer Suche* (engl: *social search*) [7, 42].
Das Konzept der sozialen Suche findet auch im P2P-Umfeld Anwendung. Beispielsweise können Suchanfragen gezielt nur an solche Personen weitergereicht werden, die ein Nutzerprofil besitzen, das eine hohe *Ähnlichkeit* mit der Suchanfrage bzw. dem Nutzerprofil des Anfragestellers aufweist [61, 22, 15, 225, 135]. Die Ähnlichkeit definiert sich z.B. über den Grad der Überschneidung persönlicher Profilattribute zweier Nutzer. Die Kenntnis des sozialen Graphen dient zudem als Entscheidungsgrundlage zur Weiterleitung von Anfragen innerhalb mobiler (Ad-Hoc-)-Netzwerke [40, 198, 133, 78].

4.2 Besonderheiten in dezentralen OSNs

Im Gegensatz zu zentralisierten Systemen müssen in dezentralen OSNs soziale Suchanfragen durch die Nutzer weitergeleitet werden. Das Ziel einer entsprechenden Routing-Strategie besteht darin, Anfragen an Freunde zu delegieren, die mit

hoher Wahrscheinlichkeit eine Antwort auf die Anfrage liefern können. Zudem sollte eine Strategie das notwendige Nachrichtenaufkommen, den Protokoll-Overhead, die Antwortzeit sowie die Erfolgsrate optimieren.
Im Hinblick auf die Umsetzung der sozialen Suche in dezentralen OSNs müssen einige Aspekte gesondert berücksichtigt werden. Vegas unterliegt hier sehr speziellen Anforderungen, da sich jegliche Kommunikation auf das Egonetzwerk beschränkt.

4.2.1 Statische Netzwerktopologie

In vielen P2P-Netzwerken ist es möglich, je nach Bedarf neue Kanten zwischen zwei Knoten einzufügen. Das Vorgehen bietet sich an, wenn sich dadurch die Effizienz einer Routing-Strategie deutlich erhöht. Der Einsatz von Superpeers oder eines DHTs erlaubt es zudem, Verbindungen automatisiert zwischen zwei Knoten zu generieren.
Die Mehrzahl dezentraler OSNs verwendet ein strukturiertes P2P-Overlay auf der Basis eines DHTs [56, 39, 159, 189]. Das Ziel strukturierter P2P-Netzwerke besteht in der effizienten Speicherung und Suche von Inhalten (vgl. Kap. 3.2). Wird ein Datum in einem DHT abgelegt und indexiert, dann ist der effiziente Zugriff mit einer Komplexität von $O(log(n))$ garantiert [136].
Im sozialen Graphen von Vegas entstehen neue Kanten ausschließlich durch die Ausbildung einer neuen Freundschaft. Eine dynamische Erzeugung *abkürzender Kommunikationsverbindungen* (engl: *short cut links*) ist nicht möglich. Suchverfahren strukturierter P2P-Netzwerke können an dieser Stelle nicht eingesetzt werden.
Eine Routing-Strategie für Vegas darf sich lediglich auf die statische Topologie des sozialen Graphen beziehen. Dies muss bei der Entwicklung einer entsprechenden Routing-Strategie beachtet werden.

4.2.2 Integration von Kontextinformationen

Die Einbeziehung von Kontextinformationen der Nutzer kann das Routing von Suchanfragen enorm verbessern (vgl. Kap. 4.1). Suchanfragen an Freunde können zu schnelleren bzw. besseren Resultaten führen, wenn sie ähnliche Eigenschaften oder Interessen mit dem Anfragesteller teilen [154]. Auch die Berücksichtigung des Anfrageinhalts bei der Selektion eines Freundes kann eine deutliche Verbesserung bewirken [71]. Umgekehrt kann es sich auch als sinnvoll erweisen, für das Routing einer Suchanfrage einen Freund zu selektieren, der nur eine sehr geringe Übereinstimmung mit den eigenen Profilattributen bzw. Interessen hat [101, 111, 228]. Unter Umständen gelangt man über diesen Freund an Informationen, die in der eigenen Nachbarschaft nicht existieren.
In Vegas kann ein Nutzer für jeden seiner Freunde individuell die Sichtbarkeit seiner Profilinformationen einstellen. Somit können sich auch die einsehbaren Kontextinformationen der Nutzer des eigenen Egonetzwerks stark voneinander unterscheiden. Der Mangel an Kontextinformationen kann starken Einfluss auf das Verhalten der einen oder anderen Routing-Strategie haben. Eine Routing-Strategie sollte stets ein

robustes Verhalten aufweisen, auch wenn nur eine reduzierte Menge an Kontextinformationen zur Verfügung steht.

4.2.3 Sichtbarkeit des sozialen Graphen

Bedingt durch die Anforderungen an die Sicherheit und den Schutz der Privatsphäre unterliegen dezentrale OSNs einer eingeschränkten Sichtbarkeit des sozialen Graphen. Beispielsweise erlaubt Vegas lediglich den Zugriff auf Profilinformationen von Nutzern des eigenen Egonetzwerks. Unter Umständen kann eine Routing-Strategie jedoch deutlich bessere Ergebnissen erzielen, wenn man Abstriche in Bezug auf die eigene Anonymität in Kauf nimmt. Es wäre z.B. denkbar, dass eine Priorisierung von Freunden mit einem großen Freundeskreis die Erfolgsrate einer Routing-Strategie signifikant verbessert. Da sich im Falle von Vegas diese Informationen nicht direkt aus dem eigenen Egonetzwerk erschließen, müssen sie durch Freunde explizit bereitgestellt werden. Die Veröffentlichung kann jedoch den Grad der Anonymität anderer Nutzer mindern. Weiß ein Nutzer u, dass zwei seiner Freunde v und w jeweils genau zwei Kontakte besitzen, dann kann u mit dem Einsatz von Relationship-Hashes (vgl. Kap. 3.5) direkt auf eine Freundschaft zwischen v und w schließen.
Es muss von Fall zu Fall analysiert werden, inwieweit die Erfolgsrate einer Routing-Strategie die Veröffentlichung solcher Informationen rechtfertigt. Eine Routing-Strategie darf daher zunächst keine Annahmen über die Herkunft verwendeter Kontextinformationen treffen.

4.2.4 Asynchrone Kommunikation

Dezentrale OSNs verwenden in der Regel ein asynchrones Kommunikationsprotokoll (vgl. Kap. 3.2). Auch das Exchanger-Konzept von Vegas sieht einen asynchronen Nachrichtenaustausch vor. Es existieren keinerlei Vorgaben, wann bzw. wie schnell eine Nachricht übermittelt werden muss. Um die Toleranz gegenüber zeitlichen Verzögerungen zu wahren, sollten generell Routing-Mechanismen vermieden werden, die ein hohes Kommunikationsaufkommen verursachen. Dazu zählen das wiederholte Senden einer Suchanfrage nach Ablauf eines Timers, das Senden von Kontrollnachrichten zur Bestimmung der Netzlast bzw. der Verfügbarkeit bestimmter Knoten oder auch der Einsatz von Synchronisationsnachrichten zur Identifikation erfolgreicher bzw. fehlgeschlagener Suchanfragen.

4.3 Existierende Ansätze

Für unstrukturierte P2P-Netzwerke wurde in der Vergangenheit eine Vielzahl von Weiterleitungsstrategien entwickelt. Allgemein können Routing-Algorithmen für unstrukturierte P2P-Netzwerke in *blinde* (engl: *blind search*) und *informierte Suchverfahren* (engl: *informed search*) unterteilt werden [202]. Daneben existieren zahl-

reiche Verfahren, die eines der beiden Konzepte um die Integration von Kontextinformationen der beteiligten Knoten erweitern.
Im Folgenden werden Suchverfahren der drei Kategorien vorgestellt, die auch für dezentrale OSNs wie Vegas in Frage kommen. Davon ausgenommen sind Verfahren, die auf einer hierarchischen Netzwerkstruktur aufbauen. Ansätze dieser Art sind aufgrund der stringenten Anforderungen an den Schutz der Privatsphäre nicht geeignet.

4.3.1 Blinde Suchverfahren

Bei blinden Suchverfahren existiert keinerlei Vorwissen über den Speicherort einer angefragten Ressource. Um die Wahrscheinlichkeit einer positiven Antwort zu erhöhen, versuchen diese Verfahren eine Anfrage an möglichst viele Nutzer zu adressieren. In der Regel ist das Ziel ein vernünftiger Kompromiss zwischen der Anzahl erreichter Nutzer, dem Anteil versendeter Nachrichten und der erzielten Antwortzeiten.
Im Folgenden werden die wichtigsten blinden Suchverfahren vorgestellt.

4.3.1.1 Flooding

In unstrukturierten P2P-Netzwerken wie Gnutella [261] und FastTrack [268] kommt als Routing-Strategie das *Fluten* (engl: *flooding*) zum Einsatz. In der Graphentheorie ist das Flooding-Konzept auch unter dem Begriff der *Breitensuche* (*BFS*, engl: *breadth first search*) bekannt [121].
Der Anfragesteller sendet seine Suchanfrage an alle seine Nachbarn. Kann ein Knoten eine eingehende Suchanfrage nicht beantworten, leitet er die Suchanfrage ebenfalls an alle seine Nachbarn weiter. Dieser Prozess wird so lange wiederholt, bis die Suche eine maximale Anzahl von Hops traversiert hat. In der Regel kommt ein Zähler für die *Lebenszeit* (*TTL*, engl: *time to live*) einer Suchanfrage zum Einsatz. Jeder Knoten, den die Suchanfrage traversiert, erniedrigt den TTL-Wert um eins. Die Suche stoppt, sobald der TTL-Zähler den Wert 0 erreicht.
Problematisch erweist sich die Bestimmung einer angemessenen Anzahl von Hops. Ein zu hoher TTL-Wert führt zu einer unnötigen Belastung des Netzwerks. Ohne entsprechende Einschränkungen wächst die Nachrichtenlast mit jedem Weiterleitungsschritt exponentiell an [222]. Ein zu niedriger Wert führt dazu, dass eine Suchanfrage zum Erliegen kommt, bevor sie einen Knoten erreicht, der eine Antwort liefern kann.
Ein weiteres Problem stellen Nachrichtenduplikate dar. Besonders negativ wirken sie sich in stark vernetzen Graphen aus [139]. Eine Suchanfrage wird als Duplikat angesehen, wenn ein Knoten die Kopie derselben Suchanfrage von mehreren Nachbarn erhält. Duplikate erzeugen eine unnötige Belastung der Knoten, ohne dabei die Chance zu erhöhen, eine Antwort auf die Suchanfrage zu erhalten. Ein Mechanismus zur Erkennung von Duplikaten wird beim Flooding und seinen Varianten immer als zwingend erforderlich vorausgesetzt.

Flooding kann unabhängig von der zugrunde liegenden Topologie eines Netzwerks eingesetzt werden. Zudem basiert Flooding auf keinerlei Kontextinformationen der Suchanfrage oder Nutzer. Sieht man von der Erkennung von Duplikaten ab, werden Suchanfragen beim Flooding an jeden möglichen Nachbarn weitergeleitet. Asynchrone Kommunikationsstrukturen stellen daher kein Problem dar. Flooding kann ohne Modifikationen direkt für die Umsetzung der sozialen Suche in dezentrale OSNs wie Vegas eingesetzt werden.

4.3.1.2 Modifizierte Breitensuche

Die *modifizierte Breitensuche* (engl: *modified BFS*) [109] stellt eine einfache Abwandlung des Floodings dar. Die Weiterleitung einer Suchanfrage wird auf eine Teilmenge der Nachbarschaft eingeschränkt. Die Anzahl der Empfänger wird über ein vorgegebenes Verhältnis der Größe der Teilmenge zur Anzahl aller Nachbarn festgelegt. Die Auswahl der Empfänger erfolgt zufällig.
Die modifizierte Breitensuche reduziert die Nachrichtenlast gegenüber Flooding signifikant. Gleichzeitig erreicht das Verfahren nahezu die gleiche Erfolgsrate bei der Beantwortung von Suchanfragen.
Wie im Falle von Flooding werden keinerlei Annahmen über die Topologie des Netzwerks getroffen. Asynchrone Kommunikationsstrukturen stellen kein Problem dar und es gibt keine Abhängigkeit von bestimmten Kontextinformationen. Modified BFS kann wie Flooding direkt für die Umsetzung der sozialen Suche in dezentralen OSNs wie Vegas eingesetzt werden.

4.3.1.3 Expandierende Ringsuche

Die *expandierende Ringsuche* (engl: *expanding ring search*) [139] versucht die Netzlast beim Flooding durch eine iterative Anpassung des TTL-Wertes zu reduzieren.
Neben dem statischen TTL-Wert wird jeder Suchanfrage ein zusätzlicher Tag angefügt, der die Anzahl von Hops vorgibt, an die eine Suchanfrage maximal weitergeleitet werden soll. Die Anzahl der Hops erhöht sich mit jeder weiteren Iteration der Suche, bis der Anfragesteller eine Antwort erhält oder der TTL-Wert überschritten ist. Um redundante Nachrichten zu vermeiden, werden eingehende Suchanfragen für kurze Zeit zwischengespeichert. Die Anfrage wird „eingefroren“. Knoten, die eine Anfrage einfrieren, werden auch als *Anfragefront* bezeichnet. In einer nachfolgenden Iteration wird nicht die Suchanfrage, sondern lediglich eine Aufforderung zum wiederholten Senden der Anfrage verschickt. Die Aufforderung beinhaltet die Anzahl der Hops, an die eine Suchanfrage in der nächsten Iteration weitergeleitet werden soll. Erreicht sie die Anfragefront, wird die eingefrorene Suchanfrage entsprechend der geforderten Anzahl von Hops propagiert.
Die expandierende Ringsuche skaliert besonders gut, wenn die Mehrzahl der Suchanfragen in weniger als TTL Schritten beantwortet werden kann. Die Anzahl reduziert sich proportional zur Replikationsdichte der gesuchten Informationen. Da

beim normalen Flooding der Anteil gesendeter Nachrichten mit jedem Schritt exponentiell anwächst, bedeutet es insgesamt weniger Aufwand, eine Suchanfrage mehrmals in geringer als nur einmalig in großer Tiefe zu verarbeiten.
Ein Problem des Verfahrens stellt die gesteigerte Antwortzeit dar. Abhängig von der Replikationsdichte kann es sehr lange dauern, bis die Anfragefront Knoten erreicht, die eine Anfrage beantworten können.
Im Gegensatz zum Flooding eignet sich die expandieren Ringsuche weniger gut für Netzwerke mit hoher Kommunikationsverzögerung. In Bezug auf den Einsatz innerhalb dezentraler OSNs wie Vegas bleiben alle anderen Eigenschaften jedoch unberührt.

4.3.1.4 Random Walks

Im Gegensatz zum Flooding veranschlagen *Zufallsbewegungen* (engl: *random walks*) eine konstante Anzahl von Suchanfragen pro Weiterleitungsschritt. In der einfachsten Variante kommt genau ein Walker zum Einsatz, der sich in jedem Schritt von einem Knoten zum nächsten bewegt. Ein Knoten selektiert zufällig einen seiner Nachbarn und leitet seine Suchanfrage an diesen weiter. Der Nachbar beantwortet entweder die Suchanfrage, oder leitet diese wiederum an einen zufällig ausgewählten Nachbarn weiter.
Offensichtlich reduziert ein Random Walk das Nachrichtenaufkommen auf Kosten der Antwortzeit. Ein Walker irrt unter Umständen lange durch den Graphen, bevor er auf einen Knoten trifft, der die Suchanfrage beantworten kann. Einen Kompromiss stellt der *k-Random Walk* dar. Um die Antwortzeit zu senken, werden mehrere Walker parallel gestartet.
Die Lebensdauer eines Walkers lässt sich durch eine TTL begrenzen. Alternativ kann man einen Walker dazu zwingen, regelmäßig zu überprüfen, ob der Anfragesteller bereits eine Antwort auf seine Suchanfrage erhalten hat. Abhängig vom Replikationsgrad sowie der Anzahl paralleler Walker kann dieser Ansatz das Nachrichtenaufkommen um bis zu zwei Größenordnungen reduzieren [139]. Die Antwortzeiten nehmen hingegen nur geringfügig zu.
Wie das Flooding machen Random Walks keinerlei Annahmen über die Topologie des zugrunde liegenden Graphen. Zudem werden keine gesonderten Kontextinformationen bei der Auswahl des nächsten Nachbarn verwendet. Ein Problem bei der zweiten Möglichkeit einen Walker zu terminieren, stellt die Notwendigkeit eines Rückkanals dar. Für dezentrale OSNs mit asynchronem Kommunikationsprotokoll ist dieser Mechanismus ungeeignet.

4.3.1.5 Two Level Random Walks

Der *Two Level Random Walk* [104] versucht bei maximal gleichbleibendem Nachrichtenaufkommen, die Anzahl gemeinsamer Knoten paralleler Random Walks zu reduzieren. Dazu werden pro Suchanfrage zwei Zähler TTL_1 und TTL_2 eingesetzt. Ein Knoten startet eine Suchanfrage, indem er TTL_1 mit einem vordefinierten Wert

initialisiert und k_1 zufällig ausgewählten Nachbarn zukommen lässt. Die Nachbarn leiten die Suchanfrage wie beim gewöhnlichen Random Walk Verfahren weiter.
Erreicht TTL_1 den Wert 0, wird die Suchanfrage „multipliziert“. Dazu wird TTL_2 mit einem vordefinierten Wert belegt und die Suchanfrage an k_2 zufällig ausgewählte Nachbarn weitergeleitet. Von hier aus erfolgt das Routing in Analogie zum gewöhnlichen Random Walk.
Die reduzierte Anzahl paralleler Walker in der Startphase des Two Level Random Walks führt zur Vermeidung sich überlappender Pfade. Verglichen mit einem gewöhnlichen k-Random Walk liefert das Verfahren mehr Ergebnisse bei gleichzeitiger Minimierung redundanter Treffer. Die Verbesserung geht unter Umständen jedoch zu Lasten einer erhöhten Antwortzeit. Bezogen auf die Besonderheiten dezentraler OSNs ergeben sich die gleichen Aussagen wie beim gewöhnlichen Random Walk.

4.3.2 Informierte Suchverfahren

Informierte Suchverfahren setzen ein Vorwissen über den Speicherort einer gesuchten Ressource voraus. Der Fokus liegt also nicht auf der Weiterleitung einer Suchanfrage an möglichst viele Nutzer, sondern an solche, die mit hoher Wahrscheinlichkeit eine gute Antwort auf eine Suchanfrage liefern können. Im Folgenden werden die wichtigsten informierten Suchverfahren vorgestellt.

4.3.2.1 Gerichtete und intelligente Breitensuche

Die *gerichtete Breitensuche* [222] (engl: *directed BFS*) beruht auf der Auswertung der Ergebnisse vorhergehender Suchanfragen. Neue Suchanfragen werden an solche Knoten weitergeleitet, die in der Vergangenheit schon einmal dieselbe bzw. eine ähnliche Anfrage positiv beantworten konnten.
Ähnlich zur modifizierten Breitensuche (vgl. Kap. 4.3.1.2) leitet ein Knoten die Suchanfrage nur an eine Teilmenge seiner Nachbarn weiter. Die Auswahl erfolgt jedoch nicht zufällig, sondern auf der Basis zuvor gesammelter Statistiken darüber, welcher Knoten eine Suchanfrage mit welcher Güte beantworten kann.
Untersuchung haben gezeigt, dass Heuristiken wie z.B „wähle Knoten, welche die meisten Ergebnisse liefern“ oder „wähle Knoten, welche die jüngsten Anfragen am schnellsten beantworten konnten“ die Anzahl und die Qualität der Ergebnisse im Vergleich zu einer rein zufälligen Auswahl stark verbessern konnten.
Im Unterschied zur gerichteten berücksichtigt die *intelligente Breitensuche* (engl: *intelligent BFS*) [109] auch den Inhalt der Suchanfrage.
Basierend auf einem bestimmten Ähnlichkeitsmaß wird eine eingehende Suchanfrage nur an Knoten weitergeleitet, die in der Vergangenheit bereits ein gutes Ergebnis für eine ähnliche Anfrage liefern konnten. Untersuchungen haben gezeigt, dass die intelligente Breitensuche eine vergleichbare Anzahl positiver Ergebnisse liefert und gleichzeitig die Netzlast reduziert.
Die gerichtete bzw. intelligente Breitensuche basiert auf den Kontextinformationen benachbarter Knoten. Unter der Voraussetzung, dass ein Knoten auch den Inhalt

einer Antwortnachricht einsehen kann, kommt dieser Ansatz auch für dezentrale OSNs wie Vegas in Betracht. Liegen keine Kontextinformationen vor, verhält sich das Verfahren wie die modifizierte Breitensuche.

4.3.2.2 Adaptive probabilistische Suche

Die *adaptive probabilistische Suche* (*APS*, engl: *adaptive probabilistic search*) [203] stützt sich auf die Güte der Ergebnisse vorhergehender Suchanfragen. Ein Knoten startet eine Suchanfrage, indem er einen parallelen Random Walk (vgl. Kap. 4.3.1.4) für k zufällig ausgewählte Nachbarn initiiert. Für den weiteren Pfad erfolgt die Auswahl des nächsten Hops auf der Basis eines probabilistischen Verfahrens.
Jeder Knoten verwaltet eine Tabelle, deren Inhalt sich aus Trippeln der Form `(Nachbar, Anfrage, Wahrscheinlichkeit)` zusammensetzt. Ein Trippel gibt darüber Auskunft, mit welcher Wahrscheinlichkeit eine bestimmte Art von Anfragen an den assoziierten Nachbarn weitergeleitet werden soll. Der Algorithmus unterscheidet eine *optimistische* und eine *pessimistische* Variante.
Im optimistischen Fall wird angenommen, dass der Walker ein positives Ergebnis liefert. Die Wahrscheinlichkeit für die Auswahl des selektierten Nachbarn wird erhöht. Liefert der Walker tatsächlich ein positives Ergebnis, bleibt der Eintrag unverändert. Schlägt der Walker fehl, werden alle Knoten auf dem bisherigen Pfad durch eine Synchronisationsnachricht über den Misserfolg informiert. Alle intermediären Knoten verringern ihre Wahrscheinlichkeit für die Auswahl des entsprechenden Nachbarn.
Im pessimistischen Fall wird davon ausgegangen, dass der Walker fehlschlägt. Die Wahrscheinlichkeit für die Auswahl des selektierten Nachbarn wird erniedrigt. Liefert der Walker ein positives Ergebnis, wird die Wahrscheinlichkeit entsprechend erhöht.
Der Einsatz eines der beiden Verfahren hängt von der generell erwarteten Erfolgsrate einer Suchanfrage ab. Liegt der Anteil erfolgreicher Random Walks über 50%, wird die optimistische, andernfalls die pessimistische Variante bevorzugt.
Trotz der zusätzlichen Synchronisationsnachrichten haben Untersuchungen gezeigt, dass die APS ein deutlich geringeres Nachrichtenaufkommen nach sich zieht als ein gewöhnlicher Random Walk. Kommt es zu starken Verzögerungen oder dem Verlust von Synchronisationsnachrichten, verhält sich der Ansatz im schlechtesten Fall wie ein gewöhnlicher Random Walk. Die APS ist unabhängig von der gegebenen Netzwerktopologie, basiert aber auf den Kontextinformationen benachbarter Knoten.

4.3.2.3 Lokale Indizes

Die bisher betrachteten informierten Suchverfahren beziehen ihre Informationen aus der Historie und der Erfolgsrate bereits ausgeführter Suchanfragen. *Lokale Indizes* (engl: *local indices)* [51] hingegen basieren auf der Verwaltung eines Index über die Inhalte der Knoten in lokaler Umgebung.

Ein globaler Parameter r bestimmt den Radius, über den sich die lokale Umgebung erstreckt. Jeder Knoten kann Suchanfragen stellvertretend für alle Knoten mit der Distanz r verarbeiten. Die Rolle ähnelt der eines Superpeers in hierarchischen DHT-Systemen (vgl. Kap. 2.4.2.2). Ob ein Knoten als Superpeer fungiert, hängt von der Distanz zum Anfragesteller ab.
Ein Knoten startet seine Suchanfrage mit einem lokalen Flooding. Dabei spezifiziert er die Distanz zu den Knoten, die die Suchanfrage verarbeiten sollen. Trifft die Suchanfrage auf einen Knoten mit der entsprechenden Distanz, führt er eine Suche auf seinem lokalen Index durch. Alle anderen Knoten leiten die Suchanfrage lediglich weiter.
Lokale Indizes reduzieren das Nachrichtenaufkommen enorm. Die letzten r Weiterleitungsschritte werden stets vermieden, da alle Informationen als lokaler Index vorliegen. Einen Nachteil stellt der Aufwand zum Aufbau und zur Verwaltung des Index dar. Der Churn-Effekt kann zudem dazu führen, dass der lokale Index Inhalte liefert, die nicht mehr existieren bzw. dass Inhalte nicht gefunden werden, obwohl sie existieren.
Das Verfahren basiert auf den Kontextinformationen der lokalen Umgebung eines Nutzers. Aufgrund der relativ statischen Topologie sozialer Graphen [82] stellen lokale Indizes eine interessante Möglichkeit zur Verbesserung der sozialen Suche dar. Basiert ein lokaler Index z.B. auf den Profilinformationen eines Nutzers, sind hochfrequente Veränderungen eher selten. Für Vegas könnte ein lokaler Index mit $r = 1$ z.B. auf der Basis von Profilinformationen der Nutzer des Egonetzwerks generiert werden.

4.3.3 Suchverfahren auf der Basis von Grapheigenschaften

Neben blinden und informierten Suchverfahren existieren zahlreiche Ansätze, die auf den formalen Eigenschaften eines P2P-Netzwerks beruhen. Dazu zählen Verfahren auf der Basis von Knotengraden, der Güte von Verbindungen zwischen Knoten oder auch lokaler Clustering-Eigenschaften des zugrunde liegenden Graphen. Genau genommen handelt es sich nicht um eigenständige Routing-Verfahren. Vielmehr sind diese Ansätze als Priorisierungsstrategien zu verstehen. Konzeptionell erfolgt die Ausbreitung einer Suchanfrage wie beim Random Walk. Der wesentliche Unterschied besteht darin, dass der nächste Hop in Abhängigkeit einer Gewichtungsfunktion ausgewählt wird.

4.3.3.1 Bevorzugung hochgradiger Knoten

In sozialen Netzwerken nehmen hochgradige Knoten oftmals eine besondere Stellung ein. Simulative Untersuchungen aber auch praktische Tests im P2P-Netzwerk Gnutella haben gezeigt, dass die Bevorzugung hochgradiger Knoten zu größeren Erfolgsraten führt [1]. Auch eine Reduktion der durchschnittlichen Länge von Suchpfaden konnte beobachtet werden [113].

In skaleninvarianten Netzwerken kann diese Art der Priorisierung zur Bildung einzelner sehr *stark ausgelasteter Knoten* (engl: *hot spot nodes*) führen. Im Hinblick auf den Churn-Effekt können sich gerade in P2P-Netzwerken Hot Spot Nodes sehr negativ auswirken.
Mit der Kombination eines gewöhnlichen mit einem gewichteten Random Walk, der hochgradige Knoten bevorzugt, kann man dem Problem entgegenwirken [226]. Zu Beginn sendet ein Knoten seine Suchanfrage an eine zufällig ausgewählte Untermenge seiner Nachbarn. Jeder Nachbar leitet die Suchanfrage wiederum an denjenigen seiner Nachbarn weiter, der den höchsten Knotengrad besitzt. In jedem weiteren ungeraden Schritt wird ein Nachbar zufällig, in jedem geraden Schritt entsprechend des höchsten Knotengrades selektiert.
Der Knotengrad stellt eine Kontextinformation aus der Nachbarschaft eines Nutzers dar. Da OSNs eine relativ statische Topologie aufweisen [82], unterliegt dieser Kontext jedoch nur selten einer Veränderung.
Wie P2P-Netzwerke stellen auch die meisten OSNs skaleninvariante Netzwerke dar (vgl. Kap. 2.5.3.2). Für dezentrale OSNs wie Vegas, die eine starke Vertrauensbeziehung als Anforderung für die Ausbildung einer Freundschaft voraussetzen, kann diese Aussage jedoch nicht verallgemeinert werden. Unter Umständen hat die Bevorzugung hochgradiger Knoten hier keinen positiven Effekt. Dennoch sollte die Strategie auch zur Weiterleitung von Suchanfragen in dezentralen OSNs in Betracht gezogen werden.

4.3.3.2 Bevorzugung schwacher Verbindungen

Schon früh hat man in RSNs beobachtet, dass nicht nur starke, sondern auch schwache Beziehungen zwischen Personen für die Verbreitung von Informationen von großer Bedeutung sind. Bekannt ist dieser Sachverhalt unter der Theorie über *die Stärke schwacher Verbindungen* (engl: *the strength of weak ties*) [87]. Sie besagt, dass schwach gebundene Freunde oft eine Vielzahl an Bekanntschaften pflegen, die man selbst nicht kennt. Stark gebundene Freunde hingegen besitzen einen großen Anteil gemeinsamer Freunde.
Als Beispiel für die Bedeutung schwacher Verbindungen sei hier das eines Arbeitnehmers u angeführt, der über seinen Bekanntenkreis einen Job in einem anderen Unternehmen sucht [73]. Nach der Theorie der Stärke schwacher Verbindungen besitzt ein flüchtiger Bekannter im Gegensatz zu einem sehr guten Freund von u mehr Kontakte zu Personen, die u nicht schon selbst kennt. Damit erhöht sich auch die Wahrscheinlichkeit für u, über den entfernten Bekannten an einen neuen Job zu gelangen.
Der positive Effekt konnte bereits für den Informationsaustausch in mobilen Adhoc-Netzwerken beobachtet werden [101, 111]. Neuste Studien belegen den starken Einfluss schwacher Verbindungen auch am Beispiel von Facebook [16]. Neue Informationen verbreiten sich um so schneller, je größer der Anteil eigener Kontakte ausfällt, mit denen man nur eine schwache Verbindung unterhält. Auch wenn Nutzer eher die Informationen ihrer starken Verbindungen teilen, überwiegt die An-

zahl schwacher Verbindungen derart, dass sich der größte Anteil an Informationen dennoch über schwachen Verbindungen ausbreitet.
Ähnliche Ergebnisse konnten auch schon früher für die Ausbreitung von Expertenwissen in OSNs beobachtet werden [228]. Im Gegensatz zur Priorisierung zufälliger bzw. stärkster Verbindungen konnte bei der Bevorzugung schwächster Verbindungen eine beachtliche Steigerung des Informationsaustauschs gemessen werden.
Die Stärke einer Verbindung lässt sich über die Anzahl gemeinsamer Nachbarn definieren. Dabei handelt es sich um relativ statische Kontextinformation aus der Nachbarschaft eines Nutzers. Für die Bevorzugung schwacher Verbindungen als Weiterleitungsstrategie müssen keinerlei Annahmen über die Topologie des Netzwerks getroffen werden. Asynchrone Kommunikationsstrukturen stellen ebenfalls kein Problem dar. Die Weiterleitungsstrategie eignet sich offensichtlich auch für dezentrale OSNs. Für Vegas kann die Anzahl gemeinsamer Nachbarn auf die Berechnung der Relationship-Hashes zurückgeführt werden (vgl. Kap. 3.3.7).

4.3.4 Beurteilung der vorgestellten Ansätze

Grundsätzlich hängt es von den konzeptionellen Einschränkungen eines dezentralen OSNs ab, welche Weiterleitungsstrategien überhaupt eingesetzt werden können. Aufgrund der hohen Anforderungen an die Sicherheit und den Schutz der Privatsphäre erscheint die Berücksichtigung spezifischer Besonderheiten von Vegas bei einer Beurteilung zweckmäßig.
Blinde Suchverfahren verwenden keine speziellen Kontextinformationen und machen keinerlei Annahmen über die zugrunde liegende Netzwerktopologie. Unabhängig von den speziellen Anforderungen eines dezentralen OSNs ist der Einsatz blinder Suchverfahren als Weiterleitungsstrategie prinzipiell immer möglich. Kritisch zu bewerten sind die expandierende Ringsuche und die Variante der Random Walks, die zur Überprüfung eingegangener Antworten zum Anfragesteller zurückkehren. Kommt ein asynchrones Kommunikationsprotokoll zum Einsatz, können sich bei beiden Strategien die Antwortzeiten unverhältnismäßig stark erhöhen. Generell lässt sich festhalten, dass Flooding-basierte Ansätze von ihrer hohen Erfolgsrate profitieren, während Random Walks das Nachrichtenaufkommen stark reduzieren.
Informierte Suchverfahren verwenden Kontextinformationen, um das Nachrichtenaufkommen zu verringern. Die Entscheidung für die Weiterleitung einer Suchanfrage basiert bei der gerichteten BFS, der intelligenten BFS und der APS auf der Güte zuvor beantworteter Suchanfragen. Alle drei Ansätze treffen keine Annahmen über die zugrunde liegende Netzwerktopologie und sind robust gegenüber der Abwesenheit entsprechender Kontextinformationen. Im schlechtesten Fall verhält sich die gerichtete bzw. intelligente BFS wie die modifizierte BFS und die APS wie ein gewöhnlicher Random Walk.
Lokale Indizes basieren auf der Verwaltung eines Index über die Inhalte der Knoten in einer lokalen Umgebung. Die Größe der Umgebung bestimmt, wie stark sich das Nachrichtenaufkommen reduziert bzw. wie sehr der Verwaltungsaufwand für

die Indexstruktur steigt. Das Verfahren eignet sich auch für Vegas, solange sich der Index auf Profilinformationen des Egonetzwerks beschränkt.
Bei Suchverfahren auf der Basis von Grapheigenschaften wird die Entscheidung zur Weiterleitung einer Suchanfrage in Abhängigkeit der Charakteristik des zugrunde liegenden Graphen gefällt. Es werden zwar keine besonderen Anforderungen an die Netzwerktopologie gestellt, die Erfolgsrate hängt aber weiterhin maßgeblich von der Vernetzung der Knoten ab. Sowohl für die Bevorzugung hochgradiger Knoten als auch für die Priorisierung schwacher Verbindungen kann die Abwesenheit von Kontextinformationen starken Einfluss auf das Verhalten der jeweiligen Weiterleitungsstrategie haben. Beide Verfahren sind aber auch dann anwendbar, wenn nur eine reduzierte Menge an Kontextinformationen zur Verfügung steht.
Da es sich bei Suchverfahren auf der Basis von Grapheigenschaften um reine Priorisierungsstrategien handelt, können diese problemlos mit Weiterleitungsstrategien wie Random Walks oder Flooding-Verfahren kombiniert werden.

4.4 Routing für Vegas

Im Rahmen dieser Arbeit wurden die Möglichkeiten zur Umsetzung der sozialen Suche am Beispiel von Vegas untersucht. OSNs unterliegen spezifischen Eigenschaften wie z.B. der Skaleninvarianz, dem Kleine-Welt-Phänomen und einer hohen Modularität (vgl. Kap. 2.5.3). Um deren Einfluss auf die Weiterleitung sozialer Suchanfragen zu verstehen, lag der Schwerpunkt der Untersuchungen auf graphbasierten Suchverfahren (vgl. Kap. 4.3.3).
Für Vegas wurde ein generischer Routing-Algorithmus entwickelt, der lediglich die Konfiguration einer Strategie zur Priorisierung benachbarter Knoten erlaubt. Der Algorithmus kombiniert Random Walks mit einem lokalen Index. Dieser setzt sich z.B. aus den Profilinformationen der Nachbarschaft zusammen.

4.4.1 IPW-Algorithmus

Für das Routing von Suchanfragen wurde der *informierte priorisierte Lauf* (*IPW*, engl: *informed prioritized walk*) entwickelt [293]. Er besteht aus der *Lokalen Index Suche* und dem *Priorisierten Random Walk*. Unabhängig von der verwendeten Priorisierungsstrategie läuft jeder der beiden Teile nach dem gleichen Schema ab. Zusammengefasst entspricht IPW einer Kombination aus parallelen Random Walks, der Priorisierung von Knoten und der Verwaltung eines lokalen Index.

4.4.1.1 Lokale Index Suche

Um das Nachrichtenaufkommen zu reduzieren, verwendet der IPW-Algorithmus einen lokalen Index (vgl. Kap. 4.3.2.3). Er basiert auf den Profilinformationen und anderen persönlichen Inhalten, die ein Nutzer seinen Freunden über die Datastore-Domäne zur Verfügung stellt. Neben Profilinformationen wie dem Alter, dem Ge-

schlecht oder dem Wohnort kann es sich bei persönlichen Inhalten auch um Kontextinformationen wie z.B. den Funktionsumfang des jeweiligen Clients handeln. Steht beispielsweise im Index von Nutzer u, dass sein Freund v in New York wohnt und als Client ein Smartphone mit Temperatursensor verwendet, dann ist die Wahrscheinlichkeit relativ gering, dass v eine Suchanfrage nach der aktuellen Temperatur in Berlin beantworten kann. Mit dem lokalen Index kann die Auswahl solcher Nachbarn frühzeitig ausgeschlossen werden.
Offensichtlich hängt die Effektivität eines lokalen Index stark von der Semantik und der Struktur einer Suchanfrage ab. Durch den Einsatz eines wohldefinierten Protokolls können zahlreiche Profilinformationen automatisch mit einer Suchanfrage verglichen werden. Bietet sich für bestimmte Informationen keine Möglichkeit für einen automatischen Abgleich an, muss der Nutzer selbst interagieren. Unter Umständen ist dies auch explizit gewünscht, um dem Nutzer die volle Kontrolle über die Verbreitung persönlicher Informationen zu garantieren. Ist ein automatischer Abgleich nicht möglich oder verweigert ein Nutzer eine manuelle Interaktion, wird die Suchanfrage lokal in der Nachbarschaft geflutet. Wie oft lokal geflutet wird, hängt vom verwendeten Protokoll ab.
Die Konzeption eines wohldefinierten Protokolls wurde im Rahmen dieser Arbeit nicht durchgeführt [297].

4.4.1.2 Priorisierter Random Walk

Das Routing von Suchanfragen basiert auf mehreren parallelen Random Walks. Neben einer vordefinierten TTL wird die Lebensdauer eines Walkers durch die Gültigkeit einer Suchanfrage beschränkt. Dazu werden Suchanfragen mit einem Zeitstempel markiert.
Im Gegensatz zu einem gewöhnlichen Random Walk werden Knoten nicht rein zufällig, sondern nach Priorität selektiert. Die Priorisierung basiert z.B. auf Grapheigenschaften wie der Zentralität hochgradiger Knoten, einem Ähnlichkeitsmaß wie der maximalen Überlappung gemeinsamer Profilattribute, der Anzahl bereits beantworteter Suchanfragen oder der Ähnlichkeit von Suchanfrage und Nutzerprofil.
Auf der Suche nach einer Antwort zur Frage Q überprüft ein Nutzer u zunächst seinen lokalen Index. Jedem Nachbarn, der sich entsprechend seinem Eintrag im Index potentiell für die Beantwortung von Q eignet, sendet u eine Suchanfrage mit TTL = 1. Kann und will ein Nachbar Q beantworten, sendet er seine Antwort an u. Die Nachbarn selbst leiten die Suchanfrage nicht weiter.
Die Kommunikation in Vegas verläuft asynchron und ist tolerant gegenüber Verzögerungen. Auch wenn im Index von u Knoten existieren, die eine Frage mit hoher Wahrscheinlichkeit beantworten können, so ist nicht garantiert, ob und wann u eine entsprechende Antwort erhält.
Daher startet u parallel zur Indexsuche einen k-Random Walk. Jeder Walker wird mit demselben vordefinierten TTL-Wert initialisiert. Dieser wird von jedem weiteren Hop dekrementiert. Abhängig von der verwendeten Priorisierungsstrategie verteilen sich die Walker auf die k am stärksten priorisierten Knoten. Erreicht ein Walker mit TTL > 1 einen Knoten v, dann führt v in Analogie zu u eine Suche auf

seinem lokalen Index aus. Zudem leitet v die Suchanfrage an seinen am höchsten priorisierten Nachbarn weiter.
Gewöhnliche Random Walks treffen keine Annahme darüber, wie oft ein Walker die Kante zwischen zwei Knoten traversiert. Um Duplikate von Suchanfragen zu minimieren, werden beim IPW-Algorithmus Suchanfragen nur ein Mal an denselben Nachbarn weitergeleitet. Dazu verwaltet jeder Knoten eine Tabelle, die darüber Auskunft gibt, an welchen seiner Nachbarn eine Suchanfrage bereits weitergeleitet wurde. Zur Identifikation werden Suchanfragen mit einem *global eindeutigen Identifikator* (*GUID*, engl: *globally unique ID*) [126] versehen.
Ein Knoten leitet seine Suchanfrage stets an den am stärksten priorisierten Nachbarn weiter, der die Suchanfrage bisher noch nicht erhalten hat. Ausgeschlossen davon ist der vorhergehende Absender der Suchanfrage. Ist der TTL-Zähler noch nicht abgelaufen und wurde eine Suchanfrage bereits an alle Nachbarn weitergeleitet, wird der entsprechende Eintrag für diese Suchanfrage in der Tabelle zurückgesetzt. Erst jetzt erfolgt die Auswahl entsprechend der verwendeten Priorisierungsstrategie. Ein Walker wird nur dann terminiert, wenn sein TTL-Zähler oder sein Zeitstempel abgelaufen ist. Nur Knoten vom Grad eins dürfen Suchanfragen an den Absender zurücksenden.

4.4.2 Integration für Vegas

Unter Berücksichtigung der starken Vertrauensbeziehung zwischen Freunden, lässt sich die soziale Suche in Vegas durch eine geringfügige Erweiterung des Kommunikationsprotokolls integrieren. Die Sicherheit und der Schutz der Privatsphäre eines Nutzers stehen bei der Umsetzung weiterhin im Vordergrund. Für das Routing von Suchanfragen stehen ein symmetrisches und eine asymmetrisches Verfahren [297] zur Verfügung.

4.4.2.1 Symmetrische Kommunikation

Abbildung 4.1 veranschaulicht den Kommunikationsverlauf mit einem symmetrischen Austausch von Anfrage- und Antwortnachrichten. Ein Nutzer u erzeugt aus seiner Frage Q zunächst die symmetrische Anfragenachricht $SQ(u) = \langle K^{+}_{v \to u}(Q|id_Q) \rangle$ (*SQ*, engl: *symmetric query*) und sendet diese an einen Freund v. Der Aufbau der Nachricht entspricht dem Format aus Kapitel 3.3.3.1 mit dem Inhalt Q und id_Q. id_Q stellt den GUID der Anfrage Q dar (vgl. Kap. 4.4.1.2). Identifikator und Signatur der Nachricht selbst sind hier aus Gründen der Übersichtlichkeit nicht dargestellt. Der Nachrichtenaustausch erfolgt über entsprechende Exchanger der beteiligten Nutzer.
Kann v keine Antwort zu Q liefern, erzeugt v die Nachricht $SQ(v) = \langle K^{+}_{w_1 \to v}(Q|id_Q) \rangle$ und sendet diese an einen Freund w_1. Zudem speichert v den Identifikator der Anfrage zusammen mit dem Absender als Tuppel (id_Q, u). Dieser Schritt kann sich für w_1 und nachfolgende Knoten wiederholen. Besitzt ein Empfänger w_n die Antwort A zur Anfrage Q, so erzeugt w_n die symmetrische Antwortnachricht $SA(w_n) = \langle K^{+}_{w_{n-1} \to w_n}(A|id_Q) \rangle$ (*SA*, engl: *symmetric answer*) und sendet

diese an den Absender w_{n-1} zurück. Existiert ein Tuppel (id_Q, w_{n-2}), dann erzeugt w_{n-1} eine Nachricht $SA(w_{n-1}) = \langle K^+_{w_{n-2} \to w_{n-1}}(A|id_Q) \rangle$ und sendet diese an w_{n-2}. Der Prozess endet spätestens, sobald u die Nachricht $SA(v) = \langle K^+_{u \to v}(A|id_Q) \rangle$ von v erhält.
Der IPW-Algorithmus lässt sich auf der Basis der symmetrischen Kommunikation in Vegas umsetzen. Dazu muss eine Anfragenachricht lediglich um einen TTL-Zähler und einen Zeitstempel erweitert werden. Der GUID kann direkt zur Identifikation bereits verarbeiteter Nachrichten verwendet werden.

4.4.2.2 Asymmetrische Kommunikation

Die notwendige Anzahl der Weiterleitungsschritte für Frage und Antwort sind bei der symmetrischen Kommunikation immer gleich. Um die Anzahl der Antwortnachrichten und damit auch die Antwortzeit zu reduzieren, bietet es sich daher an, die Antwort direkt an den Anfragesteller zurückzuschicken. Wenn Antworten einen anderen Weg zum Anfragesteller nehmen, kann dieser Ansatz auch Nutzer mit einem hohen Anfrageaufkommen entlasten.
Abhängig von der konkreten Implementierung eines Exchangers kann das direkte Antworten jedoch die Anonymität des antwortenden Nutzers gefährden. Kommt z.B. das XMPP-Protokoll zum Einsatz, kann unter Umständen direkt über die Absenderadresse auf die Identität des Nutzers geschlossen werden. In diesem Fall darf eine Antwort nicht unmittelbar an den Anfragesteller zurückgeschickt werden.
Unter Berücksichtigung dieser Einschränkung veranschaulicht Abbildung 4.2 den Kommunikationsverlauf für einen asymmetrischen Austausch von Anfrage- und Antwortnachrichten. Der Aufbau der Nachrichten und die Abstraktion auf die wesentlichen Bestandteile erfolgt in Analogie zum symmetrischen Austausch (vgl. Kap. 4.4.2.1).
Ein Nutzer u generiert zunächst ein temporäres Schlüsselpaar $(K^+_{u \to *}/K^-_{u \to *})$. Aus seiner Frage Q erzeugt u eine asymmetrische Anfragenachricht $AQ(u) = \langle K^+_{v \to u}(Q|id_Q|mh_A|EX_u|K^+_{u \to *}) \rangle$ (*AQ*, engl: *asymmetric query*) und sendet diese an einen Freund v. Die Exchanger-Adresse EX_u wird der Nachricht hinzugefügt,

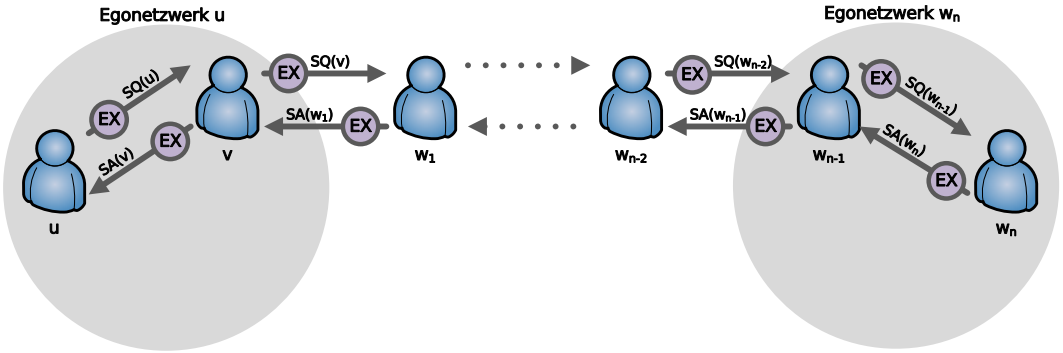


Abbildung 4.1: Kommunikationsverlauf bei der Suche mit einem symmetrischen Austausch von Anfrage- und Antwortnachrichten.

da zum Zeitpunkt der Anfrage der Rückweg für die Antwort noch nicht bekannt ist. Der Schlüssel $K^+_{u\to *}$ wird später für das Verschlüsseln einer Antwort benötigt. Der Wert mh_A (*mh*, engl: *minimum hops*) bestimmt die minimale Anzahl intermediärer Nutzer, die eine Antwortnachricht traversieren muss, bevor sie direkt an u gesendet werden darf. Der weitere Verlauf der Anfrage erfolgt wie im symmetrischen Fall.

Besitzt ein Empfänger w_n die Antwort A zur Anfrage Q, so bestimmt w_n einen seiner Freunde w_{n+1}, erzeugt die asymmetrische Antwortnachricht $AA(w_n) = \langle K^+_{w_{n+1}\to w_n}(K^+_{u\to *}(A|id_Q)|ttl|EX_u)\rangle$ (*AA*, engl: *asymmetric answer*) und sendet $AA(w_n)$ an w_{n+1}. Der TTL-Zähler ttl wird auf den Wert von mh_A gesetzt. Gilt $ttl > 1$, dann bestimmt w_{n+1} einen Freund w_{n+2}, erzeugt eine Nachricht $AA(w_{n+1}) = \langle K^+_{w_{n+2}\to w_{n+1}}(K^+_{u\to *}(A|id_Q)|ttl-1|EX_u)\rangle$ und sendet $AA(w_{n+1})$ an w_{n+2}. Abhängig von der minimalen Anzahl geforderter intermediärer Nutzer $m = mh_A$ wiederholt sich dieser Prozess m mal. Erreicht die Antwort mit $ttl = 1$ einen Nutzer w_{n+m}, so generiert dieser die Nachricht $AA(w_{n+m}) = \langle K^+_{u\to *}(A|id_Q)\rangle$ und sendet diese über EX_u an u.

Die Weiterleitung einer Antwort über mh_A Nutzer erschwert Deanonymisierungsattacken auf w_n. Problematisch bleibt jedoch die Kommunikation zwischen w_{n+m} und u. Abhängig von der Implementierung eines Exchangers könnte die Information der verwendeten Exchanger-Adressen zur Deanonymisierung von w_{n+m} bzw. u herangezogen werden.

Da der IPW-Algorithmus keine Annahmen darüber macht, auf welchem Weg eine Antwort an den Anfragesteller zurückgesendet werden muss, kann auch die asymmetrische Kommunikation für das Routing in Vegas verwendet werden. Das Verfahren macht jedoch nur Sinn, wenn sich dadurch das Nachrichtenaufkommen signifikant reduziert und eine Degradierung der Anonymität akzeptabel erscheint.

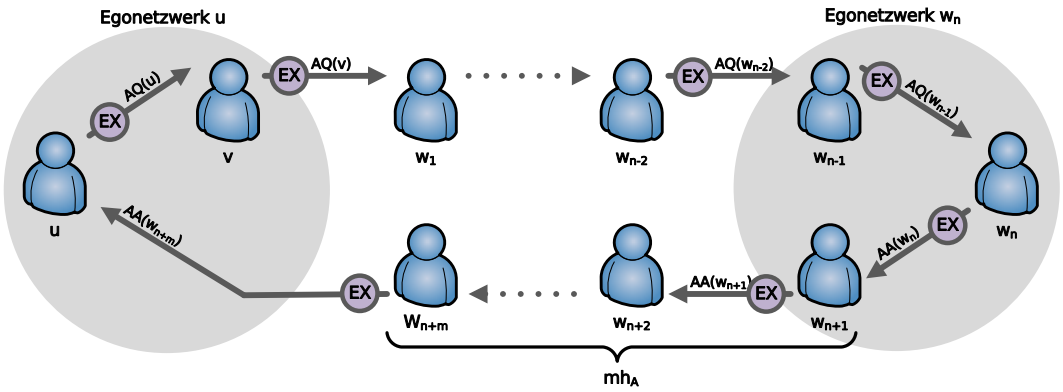


Abbildung 4.2: Kommunikationsverlauf bei der Suche mit asymmetrischem Austausch von Anfrage- und Antwortnachrichten.

4.5 Priorisierungsstrategien

Der IPW-Algorithmus (vgl. Kap. 4.4.1) repräsentiert einen generischen Ansatz für das Routing von Suchanfragen. Wie hoch die Erfolgsrate bei deren Beantwortung ausfällt, hängt letztendlich von der eingesetzten Priorisierungsstrategie ab.
Im Folgenden werden einige Priorisierungsstrategien vorgestellt. Darunter befinden sich auch solche, die nicht nur auf den Informationen des Egonetzwerks sondern auch auf dem globalen Wissen des sozialen Graphen basieren. Die Ergebnisse dienen als Anhaltspunkt dafür, wie stark eine lokal eingeschränkte Sicht auf den sozialen Graphen die Erfolgsrate einer Strategie negativ beeinflusst.
Zudem werden Verfahren untersucht, die auf dem Wissen der Vernetzung im Freundeskreis zweiter und dritter Ordnung basieren. Ziel ist es, herauszufinden ob und wie stark ein Aufweichen der Anforderungen an den Schutz der Privatsphäre zur Verbesserung der sozialen Suche führen kann.

4.5.1 Zufällige Priorisierung

Weniger als Strategie denn als Grundwahrheit dient eine *informierte zufällige Priorisierung* auf der Basis eines Random Walks (*IPRW*, engl: *informed prioritized random walk*). Kann eine Anfrage durch die lokale Indexsuche nicht beantwortet werden, wird sie an zufällig ausgewählte Nachbarn weitergeleitet. Die Wahrscheinlichkeit, dass ein Knoten v einen Nachbarn $u \in \Gamma(v)$ für die Weiterleitung auswählt, berechnet sich entsprechend der Gleichung

$$P(u) = \frac{1}{deg(v)}.$$

Effiziente Strategien sollten deutlich bessere Ergebnisse liefern als die zufällige Priorisierung.

4.5.2 Priorisierung nach Knotengrad

Die *Degree-Zentralität* [75, 172] dient als Maß zur Gewichtung der Nutzer eines sozialen Netzwerks auf der Basis ihrer Knotengrade. Dem Maß liegt die Annahme zugrunde, dass hochgradige Knoten für das Netzwerk von größerer Bedeutung sind als Knoten niedrigen Grades (vgl. Kap. 4.3.3.1). Die Nachbarn eines Knotens entsprechend ihrer Grade zu priorisieren, erfolgt daher intuitiv. Mit einer ansteigenden Anzahl von Nachbarn wächst auch die Wahrscheinlichkeit, dass einer der Nachbarn die Anfrage beantworten kann.
Die Degree-Zentralität $C_D(u)$ eines Knotens u berechnet sich entsprechend der Gleichung

$$C_D(u) = deg(u).$$

Für eine *Priorisierung nach Knotengrad* (*HIGHDEG*) benötigt ein Nutzer Informationen über den Freundeskreis zweiter Ordnung. Die Klassifizierung der Knotengrade als Information des Freundeskreises zweiter Ordnung begründet sich da-

durch, dass man über deren Kenntnis unter Umständen indirekt auf Freundschaften zwischen Personen aus dem Freundeskreis erster Ordnung schließen kann. Ein Beispiel für diese Art der Deanonymisierung wurde schon in Kapitel 4.2.3 diskutiert. In Vegas kann ein Nutzer seinen Knotengrad z.B. als Profilinformation in der Datastore-Domäne publizieren.

4.5.3 Priorisierung nach Closeness

Das Maß der *Closeness-Zentralität* [75, 172] dient dazu, die Nähe eines Knotens zu allen anderen Knoten im Graphen zu bestimmten. In sozialen Netzwerken wird ein Nutzer als umso wichtiger angesehen, je kürzer seine Pfade zu allen anderen Nutzern sind. Dies entspricht der Annahme, dass man über einen Pfad generell mehr Knoten erreichen kann, wenn der Pfad Knoten enthält, die sich in sehr kurzer Distanz zu einer Vielzahl anderer Knoten befinden.
Der Wert der Closeness-Zentralität $C_{SC}(u)$ eines Knotens u berechnet sich entsprechend der Gleichung

$$C_{SC}(u) = \left(\sum_{v \in V} d(u, v) \right)^{-1}.$$

Bei einer *Priorisierung nach Closeness* (*CLOSE*) werden Knoten entsprechend ansteigender Closeness-Zentralität bevorzugt.
Die Berechnung der Closeness-Zentralität beruht auf dem globalen Wissen des sozialen Graphen. Als Priorisierungsstrategie für Vegas kommt dieses Maß nicht in Frage. Es dient primär als Grundwahrheit für die nachfolgende Priorisierungsstrategie.

4.5.4 Priorisierung nach egozentrischer Closeness

Generell kann man zwischen *sozio-* und *egozentrischen Zentralitätsmaßen* unterscheiden [143]. Während Erstere das Wissen über den strukturellen Aufbau des gesamten sozialen Graphen benötigen, beruhen Letztere auf den Informationen des Egonetzwerks eines Nutzers.
Bedingt durch die Limitierung auf das Egonetzwerk reduziert sich die egozentrische Closeness-Zentralität auf eine Degree-Zentralität. Um dennoch die Auswirkungen einer Priorisierung auf der Basis lokal beschränkter Informationen zu verstehen, wird die egozentrische Closeness-Zentralität auf den Freundeskreis zweiter Ordnung ausgeweitet.
Diese adaptierte *egozentrische Closeness-Zentralität* $C_{EC}(u)$ eines Knotens u berechnet sich entsprechend der Gleichung

$$C_{EC}(u) = \sum_{k \in \Gamma(u)} deg(k).$$

Bei einer adaptierten *Priorisierung nach egozentrischer Closeness* (*EGOCLOSE*) werden Knoten entsprechend der Anzahl der Freunde ihrer Nachbarknoten bevor-

zugt. Mit den Knotengraden des Freundeskreises zweiter Ordnung verwendet diese Priorisierungsstrategie Informationen über den Freundeskreis dritter Ordnung (vgl. Kap. 4.5.2). Die Gefahr einer Deanonymisierung von Beziehungen zwischen Mitgliedern des Freundeskreises dritter Ordnung ist jedoch nur noch für wenige Extremfälle gegeben.
Die Veröffentlichung der Summe der Knotengrade von Freunden widerspricht den Prinzipien von Vegas. Dennoch wird auch diese Priorisierungsstrategie in Erwägung gezogen. Abhängig von den Ergebnissen besteht immer noch die Möglichkeit, zu Gunsten der Effizienz auf einen gewissen Grad der Anonymität zu verzichten.

4.5.5 Priorisierung nach Betweenness

Das Maß der *Betweenness-Zentralität* [75, 172] beschreibt, wie häufig ein Knoten auf dem kürzesten Pfad zwischen allen möglichen Kombinationen zweier Knoten des sozialen Graphen liegt. Die Betweenness-Zentralität $C_{SB}(u)$ eines Knotens u berechnet sich entsprechend der Gleichung

$$C_{SB}(u) = \sum_{u \neq v \neq w \in V} \frac{\sigma_{vw}(u)}{\sigma_{vw}}.$$

$\sigma_{vw}(u)$ entspricht der Anzahl kürzester Pfade zwischen den Knoten v und w, welche den Knoten u beinhalten. σ_{vw} entspricht der gesamten Anzahl kürzester Pfade zwischen v und w.
Bei einer *Priorisierung nach Betweenness* (*BETW*) werden Knoten entsprechend ansteigender Betweenness-Zentralität bevorzugt.
Die Berechnung der Betweenness-Zentralität basiert auf dem globalen Wissen der Struktur des sozialen Graphen. Als Priorisierungsstrategie für Vegas kommt dieses Maß nicht in Frage. Es dient jedoch als Grundwahrheit für die nachfolgende Priorisierungsstrategie.

4.5.6 Priorisierung nach egozentrischer Betweenness

Die *egozentrische Betweenness-Zentralität* stellt eine zuverlässige Approximation für die soziozentrische Betweenness-Zentralität dar [143]. Sie berechnet sich als die Anzahl der Paare von Nachbarn eines Knotens, die untereinander keine gemeinsame direkte Verbindung besitzen.
Die egozentrische Betweenness-Zentralität $C_{EB}(u)$ eines Knoten u berechnet sich entsprechend der Gleichung

$$C_{EB}(u) = |\{(v, w) : v \neq w \in \Gamma(u) \wedge (v, w) \notin E\}|.$$

Bei einer *Priorisierung nach egozentrischer Betweenness* (*EGOBETW*) werden Knoten entsprechend ansteigender egozentrischer Betweenness-Zentralität bevorzugt. Um die Anzahl der gemeinsamen Freunde zweier Nachbarn zu bestimmen, benötigt ein Knoten das Wissen über die Vernetzung seiner Nachbarn. Stellt die

Nachbarschaft in Vegas ihre Relationship-Hashes (vgl. Kap. 3.3.7) zur Verfügung, kann ein Nutzer indirekt auf seine egozentrische Betweenness schließen. Veröffentlicht ein Freund seine Betweenness-Zentralität, handelt es sich aus Sicht des Nutzers um Informationen über seinen Freundeskreis dritter Ordnung.

4.5.7 Priorisierung nach Clustering-Koeffizient

Neben den oben vorgestellten Zentralitätsmaßen kann auch der lokale Clustering-Koeffizient (vgl. Kap. 2.5.2.4) zur Priorisierung von Knoten herangezogen werden. Für einen Knoten u berechnet sich der Clustering-Koeffizient $CK(u)$ entsprechend der Gleichung

$$CK(u) = \frac{|\{(v,w) : v \neq w \in \Gamma(u) \wedge (v,w) \in E\}|}{\frac{1}{2}|\Gamma(u)|(|\Gamma(u)| - 1)}.$$

Knoten mit einem niedrigen Clustering-Koeffizient können aufgrund der geringen Vernetzung ihrer Nachbarschaft meist keiner einzelnen Gemeinschaft zugeordnet werden. Dies bedeutet im Umkehrschluss, dass solche Knoten eine höhere Wahrscheinlichkeit besitzen, als ein Verbindungsglied zwischen mehreren verschiedenen Clustern zu fungieren. Knoten mit einem hohen Clustering-Koeffizienten werden als „eingesperrt“ innerhalb ihrer stark vernetzten Nachbarschaft betrachtet. Bezogen auf das Weiterleiten von Suchanfragen könnten solche Knoten dazu führen, dass viele dieser Anfragen ohne Aussicht auf Erfolg mehrfach von den Cluster-Mitgliedern verarbeitet werden.
Eine *Priorisierung nach Clustering-Koeffizienten* (*LOWCC*), bei der Knoten entsprechend absteigender Clustering-Koeffizienten bevorzugt werden, könnte also dazu führen, dass Suchanfragen sich effizienter über den sozialen Graphen verteilen.
Um die Anzahl der gemeinsamen Freunde zweier Nachbarn zu bestimmen, benötigt ein Knoten Informationen über die Vernetzung seiner Nachbarn. Stellt die Nachbarschaft in Vegas ihre Relationship-Hashes zur Verfügung, kann man direkt seinen lokalen Clustering-Koeffizienten berechnen. Beim Clustering-Koeffizienten eines Nachbarn handelt es sich aus Sicht eines Nutzers um Informationen über den Freundeskreis dritter Ordnung.

4.5.8 Priorisierung nach Knotenähnlichkeit

Im Kontext dieser Arbeit definiert sich die *Knotenähnlichkeit* als die Anzahl der gemeinsamen Profilinformationen zweier Knoten. Der Vergleich basiert z.B. auf persönlichen Interessen, der Lieblingsmusik, bevorzugten Büchern oder Filmen. Zudem können auch Kontextinformationen hinzugezogen werden wie z.B. die örtliche Nähe zweier Nutzer.
Es wird angenommen, dass Nutzer vorzugsweise Suchanfragen stellen, die im engen Bezug zu ihren Interessen stehen. Damit steigt auch die Wahrscheinlichkeit, dass Knoten mit ähnlichen Interessen eine Suchanfrage an einen Knoten weiterleiten, der diese beantworten kann.

Bei der *Priorisierung nach Knotenähnlichkeit* (*NODESIM*) werden Knoten mit ansteigender Anzahl ihrer gemeinsamen Profilinformationen bevorzugt. Diese Strategie erfordert lediglich Wissen aus dem eigenen Egonetzwerk.

4.5.9 Priorisierung nach schwachen Verbindungen

Die Theorie der Stärke schwacher Verbindungen (vgl. Kap. 4.3.3.2) besagt, dass schwach gebundene Freunde oft eine Vielzahl an Bekanntschaften pflegen, die man selbst nicht kennt. Stark gebundene Freunde hingegen besitzen einen großen Anteil gemeinsamer Nachbarn. Die Stärke einer Verbindung lässt sich über die Anzahl gemeinsamer Nachbarn von zwei Knoten definieren.
In dieser Form ist eine Priorisierung jedoch nicht sinnvoll. Hat ein Knoten u lediglich einen Nachbarn v, kann u keinen weiteren gemeinsamen Nachbarn mit v besitzen. Eine entsprechende Priorisierung führt zu einer unverhältnismäßig starken Überbewertung von u.
Durch eine geeignete Normalisierung lässt sich die Priorisierung extrem niedriggradiger Knoten vermeiden. Ausgehend von einem Knoten v berechnet sich der Grad $GV(u)$ einer Verbindung (u, v) zu einem Nachbarn $u \in \Gamma(v)$ entsprechend der Gleichung

$$GV(u) = \frac{|\{(u, w) : u \neq w \in \Gamma(v) \wedge (u, w) \in E\}| + 1}{deg(u)}.$$

Bei der *Priorisierung nach schwachen Verbindungen* (*WEAKTIE*) werden die Nachbarn in absteigender Reihenfolge entsprechend der Anzahl gemeinsamer Freunde bevorzugt.
Um die Anzahl der gemeinsamen Freunde zweier Nachbarn zu bestimmen, benötigt ein Knoten das Wissen über die Vernetzung seiner Nachbarn. Stellt die Nachbarschaft in Vegas ihre Relationship-Hashes zur Verfügung, kann ein Nutzer direkt den Grad seiner Verbindungen zu seinen Freunden berechnen. Aus der Sicht des Nutzers handelt es sich um Informationen über den Freundeskreis zweiter Ordnung.

4.6 Datenbasis

Zum Zeitpunkt der Untersuchung der vorgestellten Priorisierungsstrategien standen keine Datensätze dezentraler OSNs zur Verfügung. Einerseits haben dezentrale OSNs nicht denselben Zulauf an Nutzern wie etablierte zentralisierte OSNs. Andererseits verhindern die Dezentralität sowie entsprechende Maßnahmen zum Schutz der Privatsphäre ein Crawling solcher Informationen. In Abhängigkeit des Zulaufs an Nutzern könnte ein entsprechender Datensatz für Vegas jedoch in naher Zukunft verfügbar sein (vgl. Kap. 3.4).
Im Folgenden werden die Alternativen vorgestellt, die stattdessen als Datenbasis für die Evaluation der Priorisierungsstrategien dienen.

4.6.1 Überblick über die verwendete Datenbasis

Untersuchungen zentralisierter OSNs haben gezeigt, dass Interaktionsgraphen genauere Aussagen über die Bedeutung sozialer Verbindungen zulassen als die isolierte Betrachtung der Kontaktgraphen [217]. Die Anzahl der Verbindungen im OSN korreliert in der Regel nicht mit der Anzahl der Freunde, mit denen ein Nutzer regelmäßig interagiert (vgl. Kap. 2.5.1.2). Um diesen Sachverhalt zu berücksichtigen, wurden aus zwei Datensätzen zentralisierter OSNs die Interaktionsgraphen extrahiert. Sie basieren auf Crawls der OSNs Flickr und Last.fm und dienen als Grundlage für die Auswertung der Priorisierungsstrategien.

Bei Vegas handelt es sich um ein unstrukturiertes P2P-Netzwerk. Es liegt daher die Vermutung nahe, dass die Verteilung von Suchanfragen mit der in anderen unstrukturierter P2P-Netzwerke beobachteten Verteilung korreliert. Obgleich einige Datensätze solcher Netzwerke existieren, liefern sie keine Informationen über den Einfluss sozialer Beziehungen auf das Suchverhalten und die Verteilung sozialer Suchanfragen.

Studien haben jedoch gezeigt, dass das Replikationsverhalten in P2P-Netzwerken der Verteilung eines Potenzgesetzes folgt [230]. Um dieses Verhalten für die unterschiedlichen Priorisierungsstrategien zu evaluieren, wurden sie auf der Grundlage eines Zufallsgraphen nach dem *Erdös–Rényi*-Modell [68] und eines skaleninvarianten Graphen nach dem *Barabási–Albert*-Modell [19] untersucht.

Zur Simulation wurden für jeden der vier Graphen 10.000 soziale Suchanfragen generiert. Vorausgehende Simulationen haben ergeben, dass eine weitere Erhöhung keine signifikanten Veränderungen in den Ergebnissen bewirkt. Für die Ausführung wurden 100 Knoten selektiert, von denen jeder 100 Suchanfragen generiert.

Tabelle 4.1 veranschaulicht die strukturellen Eigenschaften der vier verwendeten Graphen. Um eine gewisse Vergleichbarkeit der Ergebnisse zu gewährleisten, wurden alle vier Graphen auf ca. 30.000 Knoten beschränkt. Im Folgenden werden die vier Datensätze im Detail betrachtet.

	RAND	BA	FLICKR	LASTFM
Typ	modelliert	modelliert	real	real
$\|V\|$	30.000	30.000	30.000	28.324
$\|E\|$	150.000	89.994	240.089	53.508
$deg(G)$	10,00	5,99	16,01	3,78
$Verteilung$	binomial	Potenzgesetz	Potenzgesetz	Potenzgesetz
$Dia(G)$	8	8	21	29
$dkP(G)$	4,73	4,62	5,68	7,41
$CK(G)$	0,00034	0,000118	0,21752	0,07041

Tabelle 4.1: Strukturelle Eigenschaften der verwendeten Datensätze.

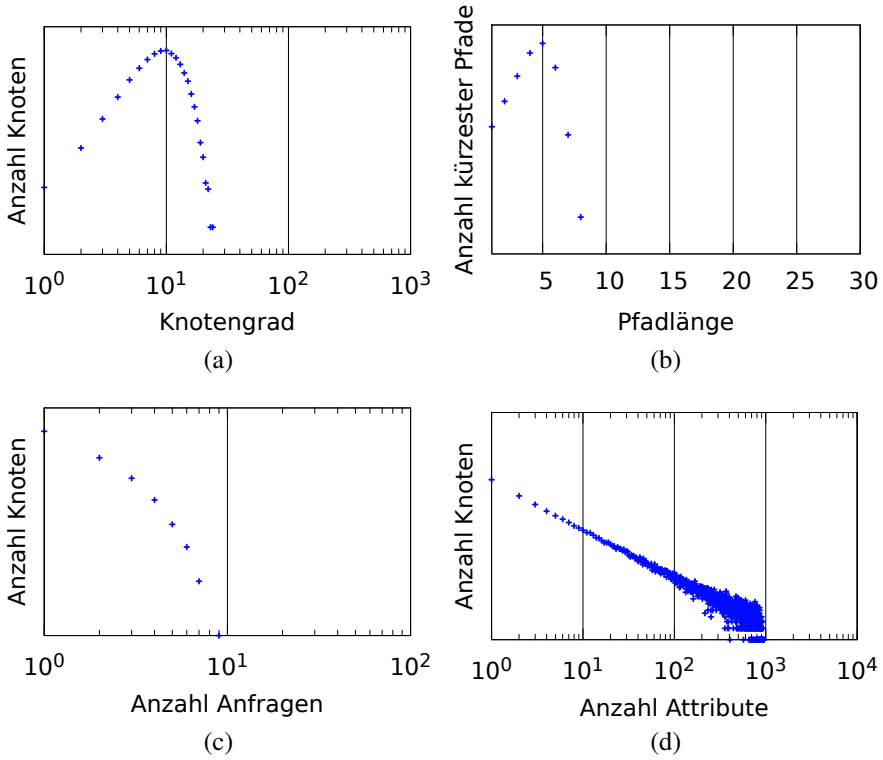


Abbildung 4.3: Verteilung der (a) Knotengrade, (b) kürzesten Pfade, (c) Suchanfragen und (d) Profilattribute für den Erdös–Rényi-Graphen RAND.

4.6.2 Erdös–Rényi-Graph

Die Analyse zahlreicher OSNs hat ergeben, dass die Gradverteilung der Knoten meist der Verteilung eines Potenzgesetzes folgt [150, 217, 122]. Viele der vorgestellten Priorisierungsstrategien sind darauf ausgelegt, von dieser Eigenschaft zu profitieren. Um die Abhängigkeit der Effektivität der Priorisierungsstrategien von den strukturellen Eigenschaften eines sozialen Graphen besser beurteilen zu können, wurden sie zusätzlich für einen Zufallsgraphen nach dem Erdös–Rényi-Modell [68] evaluiert.

Der erzeugte Interaktionsgraph *RAND* umfasst 30.000 Knoten und 150.000 Kanten. Im Durchschnitt hat jeder Knoten 10 Nachbarn. Die Verteilungen der Knotengrade und der kürzesten Pfade zwischen allen Kombinationen von Knoten werden in Abbildung 4.3(a) und 4.3(b) illustriert. Im Unterschied zu den nachfolgenden Interaktionsgraphen zeigt der Zufallsgraph eine Binomialverteilung der Knotengrade. Die Mehrzahl kürzester Pfade liegt im Bereich von vier bis sechs Hops. Im Durchschnitt hat ein kürzester Pfad eine Länge von 4,73 Hops. Der Durchmesser des Graphen umfasst 8 Hops.

Die Verteilungen der Attribute und Suchanfragen sind in Abbildungen 4.3(c) und 4.3(d) dargestellt.

4.6.3 Barabási–Albert-Graph

Das *Barabási–Albert-Modell* (*BA-Modell*) dient der künstlichen Erzeugung skaleninvarianter Graphen [19]. Es basiert auf der Theorie der *bevorzugten Bindung* (engl: *preferential attachment*), die besagt, dass sich neue Knoten bevorzugt mit hochgradigen Knoten verbinden. Das Modell besitzt lediglich einen Parameter m, der die Anzahl der Kanten vorgibt, die ein neuer Knoten beim Hinzufügen zum Graphen ausbilden soll.
Das BA-Modell erzielt eine Verteilung der Knotengrade, die einem Potenzgesetz folgt. Abgesehen vom Konfigurationsparameter m existieren keine weiteren Einflussfaktoren auf deren Verlauf. Im Zusammenhang mit der Evaluation dient das BA-Modell dazu, herauszufinden, wie stark eine Priorisierungsstrategie von den charakteristischen Eigenschaften des Flickr- bzw. des Last.fm-Graphen profitiert.
Der nach dem BA-Modell generierte Interaktionsgraph *BA* umfasst 30.000 Knoten und 89.994 Kanten ($m = 3$). Im Durchschnitt hat jeder Knoten 6 Nachbarn. Die Verteilungen der Knotengrade und der kürzesten Pfade zwischen allen Kombinationen an Knoten sind in Abbildung 4.4(a) und 4.4(b) dargestellt. Die meisten der kürzesten Pfade liegen im Bereich von vier bis sechs Hops. Im Durchschnitt hat der kürzeste Pfad eine Länge von 4,62 Hops. Der Durchmesser des Graphen umfasst 8 Hops. Das BA-Modell erzeugt kein Kleine-Welt-Netzwerk. Der Clustering-Koeffizient liegt mit 0,00118 jedoch mehr als dreimal so hoch wie beim Zufallsgraphen RAND.
Die Verteilungen der Attribute und der Suchanfragen sind in den Abbildungen 4.4(c) und 4.4(d) illustriert.

4.6.4 Flickr-Graph

Flickr stellt eine der prominentesten Dienstleistungsplattformen zur Veröffentlichung von Fotos dar. Als typischer Vertreter eines OSNs erlaubt es Flickr seinen Nutzern, ein eigenes Profil zu pflegen, Beziehungen mit anderen Nutzern zu etablieren und nach anderen Nutzern zu suchen. Zudem können Fotos anderer Nutzer als Favoriten markiert werden.
Da bei Flickr die Veröffentlichung von Fotos im Vordergrund steht, liegt die Vermutung nahe, dass die Mehrzahl der Verbindungen auf der Basis realer Beziehungen entsteht. Beispielsweise bilden sich Verbindungen zwischen Nutzern, wenn sie Fotos vom letzten gemeinsamen Urlaub oder der letzten Familienfeier teilen. Der Flickr-Datensatz wird als mögliche Approximation für einen Interaktionsgraphen herangezogen, wie er sich in Zukunft auch in Vegas etablieren könnte.
Den Untersuchungen der Priorisierungsstrategien liegt ein Flickr-Datensatz aus dem Jahr 2008 zugrunde [45]. Dieser wurde über einen Zeitraum von 104 Tagen generiert. Der Datensatz beschreibt Verbindungen zwischen Nutzern, Statistiken über hochgeladene Fotos und Markierungen von Favoriten.
Mit Hilfe eines *Metropolis–Hastings-Random-Walk* (*MHRW*) [47] wurde aus dem Datensatz der Interaktionsgraph *FLICKR* extrahiert [293]. Für jedes Foto der verbleibenden 30.000 Knoten wurde in einer Tabelle ein Identifikator zusammen mit

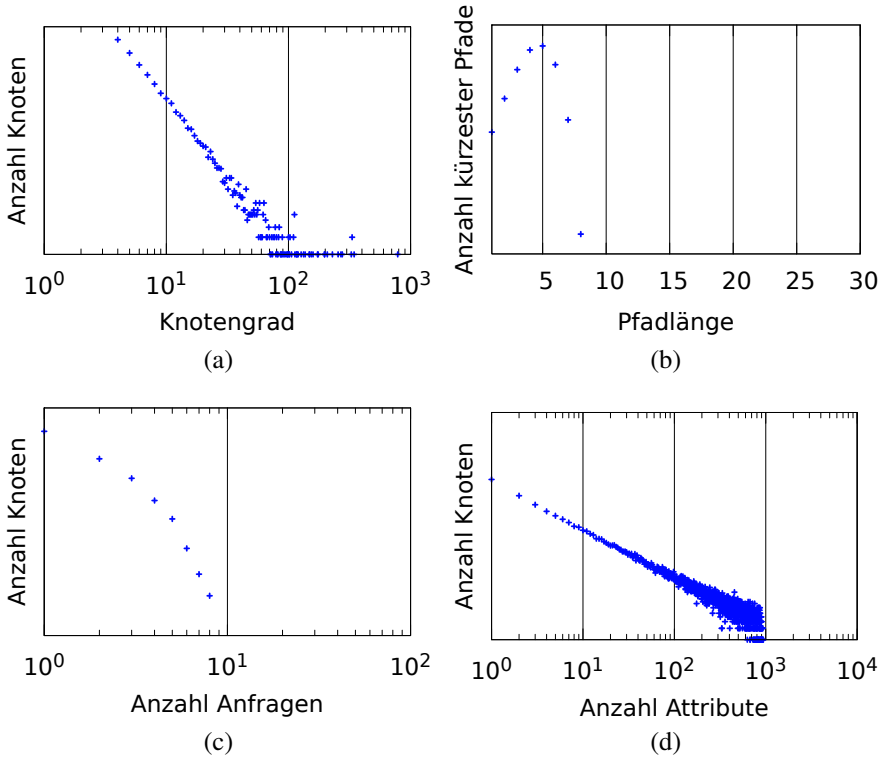


Abbildung 4.4: Verteilung der (a) Knotengrade, (b) kürzesten Pfade, (c) Suchanfragen und (d) Profilattribute für den Barabási–Albert-Graphen BA.

dem Zeitpunkt des Hochladens vermerkt. In einer weiteren Tabelle wurde für jede Markierung eines Favoriten der Identifikator des Fotos zusammen mit dem Zeitpunkt der Markierungen notiert. Auf abstrakter Ebene entspricht ein Foto einer Ressource, über die ein Nutzer Informationen besitzt. Markierungen eines Favoriten lassen sich wiederum als Zugriff auf eine Ressource interpretieren. Die erste Tabelle diente zur Herleitung der Profilattribute eines Knotens. Aus der zweiten Tabelle wurden Suchanfragen generiert.

Von 100 zufällig ausgewählten Knoten wurden 100 ihrer jüngsten Markierungen als Suchanfragen verwendet. Alle anderen Markierungen wurden in Profilattribute der Knoten transformiert. Dem Vorgehen liegt die Annahme zugrunde, dass ein Knoten, der einst eine Suchanfrage ausgeführt hat, eine Antwort erhält und die gesuchte Information kennt. Somit kann der Knoten zukünftige Anfragen nach derselben Ressource stellvertretend beantworten.

Der resultierende Interaktionsgraph besitzt 30.000 Knoten und 240.089 Kanten. Im Durchschnitt hat jeder Knoten ca. 16 Nachbarn. Die Verteilungen der Knotengrade und der kürzesten Pfade zwischen allen Kombinationen an Knoten sind in Abbildung 4.5(a) und 4.5(b) dargestellt. Die meisten der kürzesten Pfade liegen im Bereich von 4 bis 6 Hops. Im Durchschnitt hat der kürzeste Pfad zwischen zwei Knoten eine Länge von 5,68 Hops. Mit einem Clustering-Koeffizienten von 0,21752 besitzt FLICKR die Eigenschaften eines Kleine-Welt-Netzwerks. Der Durchmes-

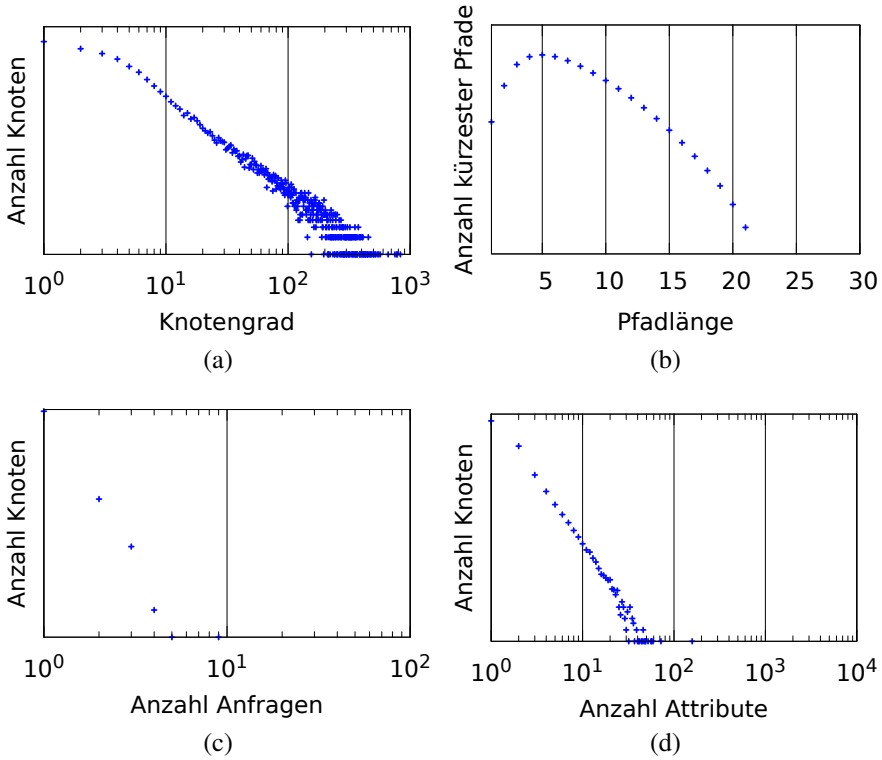


Abbildung 4.5: Verteilung der (a) Knotengrade, (b) kürzesten Pfade, (c) Suchanfragen und (d) Profilattribute für den Flickr Graphen FLICKR.

ser umfasst mit 21 Hops deutlich mehr Knoten als die künstlich erzeugten Graphen RAND und BA.
Die Verteilungen der Attribute und Suchanfragen sind in den Abbildungen 4.5(c) und 4.5(d) dargestellt. Beide Verteilungen folgen einem Potenzgesetz.

4.6.5 Last.fm-Graph

Bei Last.fm handelt es sich um eine Dienstleistungsplattform zum Tracken und Empfehlen von Musikstücken. Nutzer können ihre Lieblingskünstler und ihre favorisierten Lieder kennzeichnen. Über eine Tracking-Funktion lässt sich nachvollziehen, welche Musikstücke bereits angehört wurden. Als typischer Vertreter eines OSNs erlaubt Last.fm das Auffinden interessanter Künstler und die Ausbildung von Verbindungen zu Nutzern, die den gleichen oder einen ähnlichen Musikgeschmack teilen.
Im Gegensatz zu Flickr kann man bei Last.fm nur bedingt auf die Existenz einer realen Beziehung spekulieren, falls im OSN eine virtuelle Verbindung besteht. Eine wichtige Funktion von Last.fm stellt das Empfehlen anderer Personen mit demselben Musikgeschmack dar. Man muss davon ausgehen, dass eine Vielzahl von Verbindungen ausschließlich wegen der Ähnlichkeit von Knoten und nicht auf der Basis eines in der Realität vorhandenen wechselseitigen Interesses entsteht.

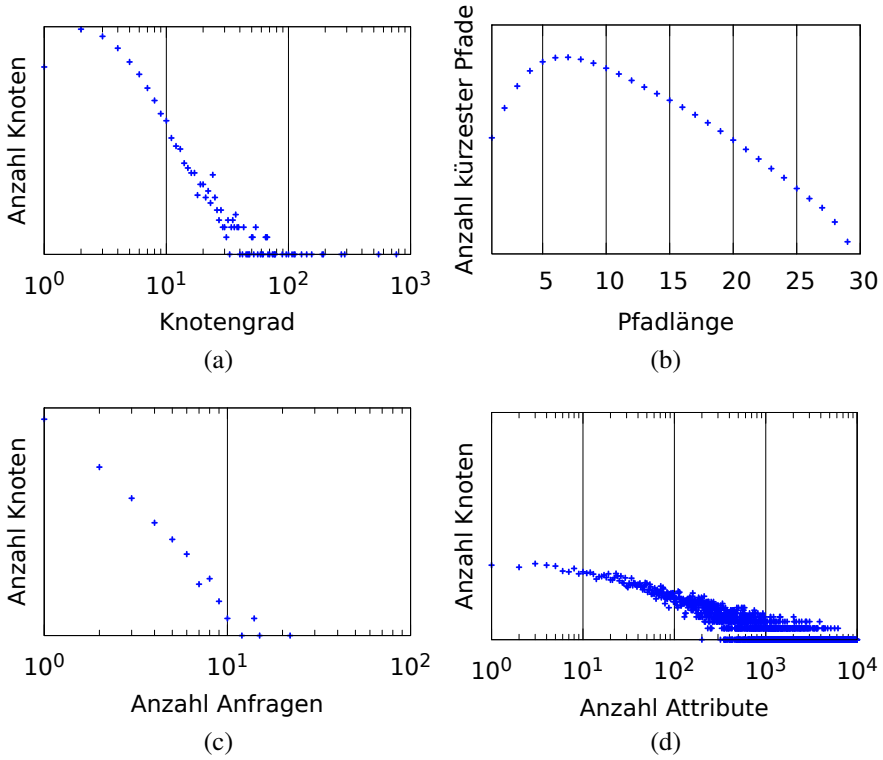


Abbildung 4.6: Verteilung der (a) Knotengrade, (b) kürzesten Pfade, (c) Suchanfragen und (d) Profilattribute für den Last.fm Graphen LASTFM.

Als Datenbasis für einen Interaktionsgraphen wurde Last.fm im Zeitraum vom 17.08.2011 bis 30.08.2011 mittels MHRW (vgl. Kap. 4.6.4) zwei Mal gecrawlt. In Analogie zu dem Vorgehen bei Flickr wurden Profilattribute und Suchanfragen auf der Basis der *Listen der am besten bewerteten Künstler* (engl: *top artist lists*) hergeleitet. Die Bewertungen des ersten Crawls dienten als Profilattribute. Die Differenz zwischen dem ersten und dem zweiten Crawl wurde als Menge der Suchanfragen interpretiert. Dabei handelte es sich um alle diejenigen Künstler, die seit dem ersten Crawl in den Top Artist Lists hinzugekommen waren. Dem Vorgehen liegt die Annahme zugrunde, dass ein Nutzer einen Künstler zunächst einmal gesucht haben muss, bevor er diesen als Top Artist in seiner Liste vermerkt. Die verbleibenden Listeneinträge wurden in Attribute transformiert.

Der resultierende Interaktionsgraph *LASTFM* besitzt 28.324 Knoten und 53.508 Kanten. Im Durchschnitt hat jeder Knoten 3 bis 4 Nachbarn. Die Verteilungen der Knotengrade und der kürzesten Pfade zwischen allen Kombinationen an Knoten sind in Abbildung 4.6(a) und 4.6(b) dargestellt. Die Mehrzahl kürzester Pfade liegt im Bereich von 5 bis 8 Hops. Im Durchschnitt hat der kürzeste Pfad zwischen zwei Knoten eine Länge von 5,68 Hops. Mit einem Clustering-Koeffizienten von 0,07041 besitzt LASTFM ebenfalls die Eigenschaften eines Kleine-Welt-Netzwerks. Der Durchmesser liegt mit 29 Hops deutlich über dem des FLICKR-Datensatzes.

Die Verteilungen der Attribute und Suchanfragen sind in den Abbildungen 4.5(c) und 4.5(d) dargestellt. Beide Verteilungen folgen einem Potenzgesetz.

4.7 Evaluation

Abhängig von der jeweiligen Priorisierungsstrategie lässt sich die soziale Suche auf unterschiedliche Messgrößen hin optimieren. Neben dem prozentualen Anteil an erfolgreich beantworteten Suchanfragen spielen die durchschnittliche Antwortzeit, die globale Verteilung der Suchanfragen sowie das Nachrichtenaufkommen eine wichtige Rolle.
Im Detail werden die Priorisierungsstrategien im Hinblick auf die folgenden Messgrößen untersucht.

Erfolgsrate Die *Erfolgsrate* definiert sich als der prozentuale Anteil aller gestarteten Suchanfragen, die mindestens in einer erfolgreichen Antwortnachricht resultieren. Wie viele Antwortnachrichten auf ein und dieselbe Suchanfrage bei einem Knoten eingehen, wird bei der Erfolgsrate nicht weiter berücksichtigt. Die Erfolgsrate gibt also Auskunft darüber, wie viele Suchanfragen eine Priorisierungsstrategie erfolgreich beantworten kann.

Durchschnittlicher kürzester Suchpfad Der *durchschnittlich kürzeste Suchpfad* definiert sich durch die minimale Anzahl von Hops, die eine Suchanfrage im Durchschnitt weitergeleitet wird, bevor diese einen Knoten erreicht, der die passende Antwort liefern kann. Wie bei der Erfolgsrate wird unter allen eingehenden Antworten lediglich diejenige mit dem kürzesten Suchpfad berücksichtigt. Der durchschnittlich kürzeste Suchpfad gibt indirekt Auskunft über die Antwortzeit einer Priorisierungsstrategie. In Vegas hängt die Antwortzeit aber auch stark von der Implementierung der eingesetzten Exchanger und dem individuellen Nutzerverhalten ab.

Verteilung der Suchanfragen Eine Priorisierung kann dazu führen, dass bestimmte Knoten mit einem sehr hohen Nachrichtenaufkommen umgehen müssen. Andere Knoten erhalten hingegen niemals eine Suchanfrage. Eine hohe Last einzelner Knoten sollte im Interesse aller Nutzer möglichst vermieden werden. Einerseits können stark ausgelastete Knoten zu Engpässen werden. Andererseits besteht die Gefahr, dass die Bereitschaft einzelner Nutzer, Suchanfragen weiterzuleiten, bei hoher Last schnell sinkt.

Eine weitere Messgröße für die Bewertung einer Priorisierungsstrategie stellt daher ihre Tendenz zur Bildung von Hot Spot Nodes dar (vgl. Kap. 4.3.3.1). Die *Verteilung von Suchanfragen* gibt Auskunft darüber, wie sich eine Priorisierungsstrategie auf die Auslastung einzelner Knoten auswirkt. Eine faire Priorisierungsstrategie erzeugt keinerlei isolierte Hot Spot Nodes.

Offensichtlich bedingt die Optimierung der dritten Messgröße eine Degradierung der ersten beiden. Priorisierungsstrategien sind dazu ausgelegt, bestimmte Knoten

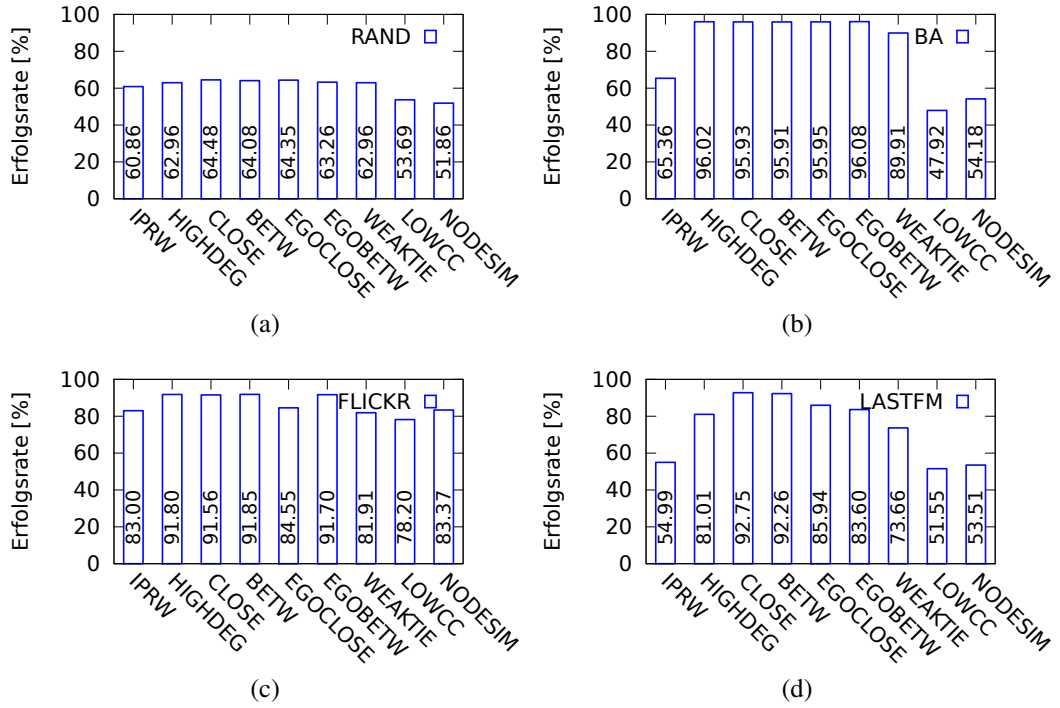


Abbildung 4.7: Erfolgsraten für die verschiedenen Priorisierungsstrategien für (a) RAND, (b) BA, (c) FLICKR und (d) LASTFM.

bei der Weiterleitung einer Suchanfrage zu bevorzugen. Es kommt also automatisch zu einer Ungleichverteilung der Suchanfragen. Wie gravierend sich eine Optimierung auswirkt, hängt stark von der zugrunde liegenden Graphstruktur ab. Letztendlich muss für jeden Anwendungsfall gesondert entschieden werden, welche Strategie sich am besten eignet.

Alle Priorisierungsstrategien werden für die im vorhergehenden Kapitel vorgestellten Datensätze auf der Basis des IPW-Algorithmus evaluiert (vgl. Kap. 4.4.1). Dabei wird genau ein Walker verwendet, der TTL-Wert wird stets auf 10 gesetzt und fehlgeschlagene Suchanfragen werden nicht wiederholt. In die Berechnung der Messgrößen fließt ausschließlich der Weg der Suchanfragen ein. Anfrageantworten bleiben unberücksichtigt.

4.7.1 Erfolgsraten

Zunächst wurden für alle Priorisierungsstrategien die Erfolgsraten untersucht. Die Ergebnisse sind in den Abbildungen 4.7(a) bis 4.7(d) dargestellt. Die exakten prozentualen Werte der erreichten Erfolgsraten sind für jede Priorisierungsstrategie im jeweiligen Histogramm vermerkt.

4.7.1.1 Zufällige Priorisierung

Bevor eine Suchanfrage an einen Nachbarn weitergeleitet wird, führt der IPW-Algorithmus eine lokale Indexsuche durch. Je mehr Nachbarn ein Knoten besitzt, desto höher liegt auch die Wahrscheinlichkeit, dass unter diesen einer existiert, der die Suchanfrage beantworten kann. Diese Tatsache spiegelt sich direkt in der zufälligen Priorisierungsstrategie IPRW wider. Mit zunehmendem durchschnittlichen Knotengrad (vgl. Tab. 4.1) steigt die Erfolgsrate an. Betrachtet man das Verhältnis der Erfolgsraten zu den durchschnittlichen Knotengraden von FLICKR ($\sim$83% : 16,01), RAND ($\sim$61% : 10,00) und LASTFM ($\sim$55% : 3,78), so würde man jedoch eine noch viel geringere Erfolgsrate für LASTFM erwarten.
Eine Erklärung für diese Beobachtung liefert der globale Clustering-Koeffizienten. Dieser liegt für FLICKR (0,21752) in etwa dreimal so hoch wie für LASTFM (0,07041). Das höhere Clustering kann dazu führen, dass Walker häufig in einem lokalen Cluster „hängen bleiben“. Da immer wieder dieselben Knoten selektiert werden, verringern sich auch die Möglichkeiten, eine Suchanfrage zu beantworten. Obwohl der durchschnittliche Knotengrad von BA (5,99) geringer ausfällt, als der von RAND (10,00), liegt die Erfolgsrate von BA ($\sim$65%) leicht oberhalb der von RAND ($\sim$61%). Diese Beobachtung lässt vermuten, dass der Effekt des Clusterings (BA: 0,000118; RAND: 0,00034) im Falle einer rein zufälligen Priorisierung von Knoten einen stärkeren Einfluss auf den Verlauf eines Walkers hat als der durchschnittliche Knotengrad.

4.7.1.2 Zentralitätsbasierte Priorisierung

Die zentralitätsbasierten Priorisierungsstrategien profitieren davon, dass die Verteilung der Knotengrade in BA, FLICKR und LASTFM einem Potenzgesetz folgt.
RAND besitzt eine binomiale Verteilung der Knotengrade. Die Erfolgsraten von HIGHDEG, (EGO-)CLOSE und (EGO-)BETW liegen hier zwischen 62,0% und 64,5%. Gegenüber IPRW (60,9%) führen zentralitätsbasierte Priorisierungsstrategien offensichtlich nur zu marginalen Verbesserungen.
Im Hinblick auf BA ($>$ 24% – 30%), FLICKR ($>$ 2% – 9%), und LASTFM ($>$ 26% – 37%) lassen sich beachtliche Verbesserungen beobachten. Der Grund für das gute Abschneiden der Strategien BETW und CLOSE liegt darin begründet, dass hochgradige Knoten in der Regel auch eine hohe Betweenness und Closeness besitzen.
Für BA zeigen die egozentrischen Priorisierungsstrategien nahezu die gleichen Erfolgsraten wie das jeweils soziozentrische Pendant. Für LASTFM ergibt sich hingegen ein etwas differenzierteres Ergebnis. Die Priorisierungsstrategie CLOSE liefert noch vor BETW die besten Ergebnisse. EGOCLOSE schneidet bereits deutlich schlechter ab als BETW, aber verhält sich immer noch besser als EGOBETW. Aufgrund der geringen durchschnittlichen Knotengrade liefert HIGHDEG in LASTFM die schlechtesten Ergebnisse unter allen Strategien. Es liegt die Vermutung nahe, dass in LASTFM die Mehrzahl der Nachbarn eines niedriggradigen Knotens ebenfalls einen niedrigen Grad besitzen.
Abbildung 4.8 verdeutlicht das Verhalten der verschiedenen Strategien an einem

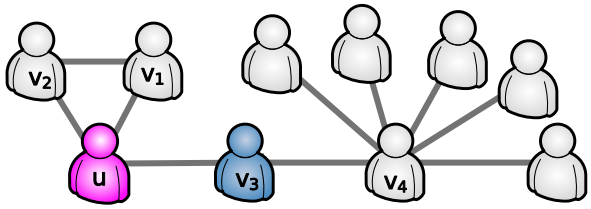


Abbildung 4.8: Beispiel für die Auswirkungen lokaler Knotengrade bei Anwendung der Priorisierungsstrategie HIGHDEG, CLOSE und BETW im Falle von LASTFM.

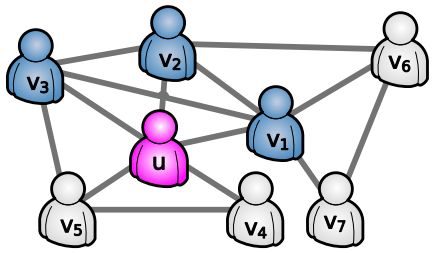


Abbildung 4.9: Beispiel für die Auswirkung lokaler Knotengrade bei Anwendung der Priorisierungsstrategie EGOCLOSE im Falle von FLICKR.

konkreten Beispiel. Wendet der Knoten u die Priorisierungsstrategie HIGHDEG an, dann liegt die Wahrscheinlichkeit, dass der Knoten v_1 als nächster Hop für die Weiterleitung einer Suchanfrage selektiert wird, genau so hoch wie für die Knoten v_2 bzw. v_3. Im Falle von CLOSE und BETW hingegen würde stets der Knoten v_3 ausgewählt werden. Da v_3 im Gegensatz zu v_1 und v_2 mit v_4 einen sehr hochgradigen Nachbarn besitzt, steigt auch die Wahrscheinlichkeit, dass sich in der Nachbarschaft von v_4 ein weiterer Knoten befindet, der die Suchanfrage erfolgreich beantworten kann.

Auch die Ergebnisse für EGOCLOSE ($\sim$85,9%) und EGOBTW ($\sim$83,6%) bestätigen diese Vermutung. EGOBETW betrachtet lediglich die Abwesenheit von Verbindungen zwischen zwei Nachbarn. EGOCLOSE hingegen errechnet sich über die Knotengrade der Nachbarn. Im Gegensatz zu EGOBETW verwendet ein Nutzer im Falle von EGOCLOSE auch Informationen über seinen Freundeskreis dritter Ordnung (vgl. Kap. 4.5.4).

Auch in FLICKR besitzen Knoten im Durchschnitt eine hohe Betweenness und Closeness. EGOCLOSE liefert hier jedoch nur eine Erfolgsrate von $\sim$84,5% und liegt damit deutlich unter allen anderen Priorisierungsstrategien wie z.B. EGOBTW mit einer Erfolgsrate von $\sim$91,7%. Verantwortlich dafür ist der hohe Clustering-Koeffizient von FLICKR.

Abbildung 4.9 verdeutlicht den Einfluss an einem weiteren Beispiel. Verwendet Knoten u EGOCLOSE als Priorisierungsstrategie, werden die Knoten v_1, v_2 und v_3 mit größerer Wahrscheinlichkeit für die Weiterleitung einer Suchanfrage in Betracht gezogen als die Knoten v_4 bis v_7. Wird z.B. v_1 ausgewählt, dann besitzen die Knoten v_2 und v_3 eine höhere Wahrscheinlichkeit wieder ausgewählt zu werden als z.B. die

Knoten v_6 und v_7. Die Wahrscheinlichkeit, dass sich ein Walker in einem Verband von Knoten „verfängt“, steigt mit Zunahme des Clusterings unter Umständen an.

4.7.1.3 Clustering-basierte Priorisierung

Bei der Priorisierungsstrategie LOWCC werden Knoten mit kleinen Clustering-Koeffizienten bevorzugt. Entgegen der theoretischen Erwartungen liefert diese Strategie durchwegs schlechtere Ergebnisse als die anderen Ansätze. LOWCC besitzt sogar schlechtere Erfolgsraten als die zufällige Auswahl von Knoten mit IPRW. Offensichtlich existiert nur selten die Situation, dass ein nur gering geclusterter Knoten als Verbindungsglied zwischen zwei sehr stark geclusterten Knoten fungiert. Die meisten solcher Knoten befinden sich außerhalb der Peripherie des sozialen Graphen und dienen lediglich als „Brücke“ für Knoten mit nur einem Nachbarn.

4.7.1.4 Weak-Tie-basierte Priorisierung

Die Stärke einer schwachen Verbindung zweier Knoten u und v verhält sich proportional zu der Anzahl ihrer gemeinsamen Freunde $|\Gamma(u) \cap \Gamma(v)|$. Ein geringer globaler Clustering-Koeffizient liefert ein erstes Indiz dafür, dass es auch wenige Knoten gibt, die eine hohe Anzahl gemeinsamer Freunde teilen. Unterscheiden sich die meisten Knoten nicht gravierend in ihrer gemeinsamen Anzahl von Freunden, verhält sich die Priorisierungsstrategie WEAKTIE ähnlich wie HIGHDEG. Erwartungsgemäß besitzt WEAKTIE im Fall von RAND mit $\sim$63% eine ähnliche Erfolgsrate wie die zentralitätsbasierten Priorisierungsstrategien. Anders verhält es sich für BA, FLICKR und LASTFM, deren Knotengrade jeweils der Verteilung eines Potenzgesetzes folgen. Im Vergleich zu IPRW verbessert WEAKTIE die Erfolgsrate für BA um $\sim$34% und für LASTFM um $\sim$18%. Obwohl FLICKR von den drei Graphen den größten Clustering-Koeffizienten besitzt, zeigt WEAKTIE hier keinerlei Verbesserung gegenüber IPRW. Diese Beobachtung ist nicht zuletzt der Tatsache geschuldet, dass IPRW sowieso schon eine relativ hohe Erfolgsrate für FLICKR liefert.
In allen drei Graphen rangiert die Erfolgsrate von WEAKTIE $\sim$6% – 10% unterhalb der von HIGHDEG. Sowohl WEAKTIE als auch EGOBTW versuchen auf der Basis der gemeinsamen Nachbarschaft zweier Knoten, eine Verbesserung der Erfolgsrate zu erzielen. Im Gegensatz zu EGOBTW schafft es WEAKTIE in keinem Fall, die Erfolgsrate von HIGHDEG zu verbessern.

4.7.1.5 Ähnlichkeitsbasierte Priorisierung

Für RAND und BA erfolgt die Zuweisung der Attribute entsprechend einem Potenzgesetz. Unter den Attributen selbst existiert jedoch keine weitere Korrelation. Dennoch lassen sich bei beiden Graphen für die Priorisierungsstrategie NODESIM Erfolgsraten beobachten, die $\sim$13% – 41% schlechter ausfallen als im Falle der zentralitätsbasierten Priorisierungsstrategien. Auch IPRW liefert mindestens um $\sim$9% bessere Erfolgsraten als NODESIM.

Das schlechte Abschneiden von NODESIM wird auf den folgenden Seiteneffekt zurückgeführt: Jedes gemeinsame Attribut zweier Knoten u und v verringert die Wahrscheinlichkeit, dass u eine Suchanfrage für eine Ressource weiterleitet, die v bereitstellen kann. Dieser Effekt wurde auch in P2P-Netzwerken beobachtet [90].
Für FLICKR und LASTFM liefert NODESIM nur eine leichte Verbesserung der Erfolgsraten. Entgegen den Erwartungen beträgt der Unterschied zu IPRW weniger als 1,5%. Die Korrelationen zwischen den Attributen in LASTFM und FLICKR wirkt sich offensichtlich nur wenig positiv bei der Priorisierung durch NODESIM aus. Dieser Einfluss kann den zuvor erläuterten Seiteneffekt nicht kompensieren.
Einen weiteren Grund für das schlechte Abschneiden stellt die fehlende Transitivität der Ähnlichkeit von Knoten dar. Sind sich die Knoten u und v sowie v und w in ihren Attributen jeweils sehr ähnlich, dann trifft dies nicht zwangsläufig auch für die Attribute von u und w zu. Der Effekt einer Priorisierung mit NODESIM nimmt mit zunehmender Anzahl von Hops daher ab.

4.7.2 Durchschnittlich kürzester Suchpfad

Neben der Erfolgsrate stellt die durchschnittliche Antwortzeit ein zweites wesentliches Kriterium bei der Bewertung einer Priorisierungsstrategie dar. In Bezug auf Vegas hängt diese stark von der Implementierung des verwendeten Exchangers bzw. dem individuellen Nutzerverhalten ab. Daher wird hier als Maß für die Beurteilung der Antwortzeit die minimale Anzahl von Hops verwendet, die eine Suchanfrage weitergeleitet werden muss, bevor sie auf einen Knoten trifft, der eine Antwort liefern kann. Die benötigte Anzahl von Hops wird über alle getätigten Suchanfragen gemittelt. Es handelt sich bei diesem Wert also um den durchschnittlich kürzesten Suchpfad.
Die Resultate aller Priorisierungsstrategien sind in den Abbildungen 4.10(a) bis 4.10(d) zusammengefasst. Neben den Ergebnissen für die einzelnen Priorisierungsstrategien enthält jedes Histogramm auch das Ergebnis für ein gewöhnliches Flooding (FL). Da Flooding das theoretische Optimum für den durchschnittlich kürzesten Suchpfad liefert, lassen sich die Aussagen über die Effizienz der Priorisierungsstrategien präzisieren.
Der TTL-Wert wurde für FL auf 7 gesetzt. Dies entspricht dem Standardwert von Gnutella [254]. Die Parametrisierung der anderen Priorisierungsstrategie erfolgte derart, dass mindestens 95% aller Suchanfragen erfolgreich beantwortet wurden, die im Falle von FL generiert wurden. Diese untere Schranke war notwendig, da man den IPW-Algorithmus nicht so parametrisieren kann, dass dieser eine konstante Erfolgsrate erzielt. In einigen Voruntersuchungen hat sich gezeigt, dass die geringe Anzahl sehr langer Suchpfade keinen signifikanten Einfluss auf die Berechnung des durchschnittlich kürzesten Suchpfades hat. Da sich die Auswertung auf bis zu 95% der Suchanfragen reduziert, liefern manche Strategien ein besseres Ergebnis, als das theoretische Optimum von FL erwarten lässt.
Erwartungsgemäß erkennt man bei allen Interaktionsgraphen eine starke Abhängigkeit zwischen den durchschnittlichen Knotengraden und der errechneten Länge

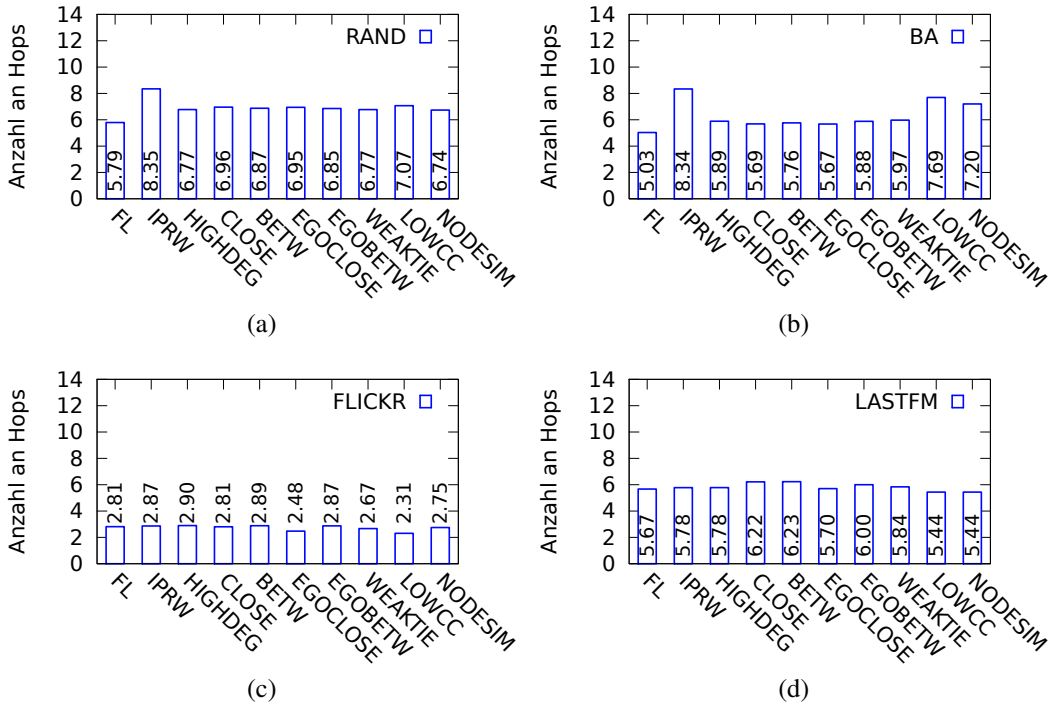


Abbildung 4.10: Ermittelte durchschnittliche kürzeste Pfade für die Interaktionsgraphen (a) RAND, (b) BA, (c) FLICKR und (d) LASTFM.

der durchschnittlich kürzesten Suchpfade. Für FLICKR liegt der durchschnittliche Knotengrad bei 16,01 und eine Suchanfrage findet nach 2,31 - 2,94 Hops die erste Antwort. LASTFM hat mit 3,78 den kleinsten durchschnittlichen Knotengrad. Hier kann eine Suchanfrage im Durchschnitt erst nach 5,44 - 7,06 Hops beantwortet werden.

Das Ergebnis für BA lässt vermuten, dass der Clustering-Koeffizient wieder wesentlichen Einfluss auf die benötigte Anzahl von Hops hat. Die durchschnittlichen Knotengrade von BA (5,99) und LASTFM (3,78) korrelieren nicht mit dem relativ geringen Unterschied der durchschnittlichen Anzahl von Hops. Offensichtlich führen sehr kleine Clustering-Koeffizienten in BA dazu, dass Suchanfragen längere Wege in Kauf nehmen müssen, bevor sie beantwortet werden können. Diese Vermutung wird auch durch das Ergebnis der Priorisierungsstrategie LOWCC gestützt. Für LASTFM liefert LOWCC neben WEAKTIE den kürzesten durchschnittlichen Suchpfad, während das Ergebnis im Falle von BA mit zu den schlechtesten zählt.

Im Fall von FLICKR erklären das hohe Clustering und die hohen Knotengrade auch die ähnlichen Ergebnisse aller Priorisierungsstrategien für die durchschnittliche Anzahl von Hops. Der Einfluss der Priorisierungsstrategie geht aufgrund dieser beiden Einflussfaktoren nahezu gänzlich verloren.

4.7.3 Verteilung der Suchanfragen

In der lokalen Umgebung eines Knotens führt eine Priorisierungsstrategie dazu, dass sich die Last asymmetrisch über die Nachbarknoten verteilt. Dieses Verhalten kann zur Ausbildung von Hot Spot Nodes führen.

Während Hot Spot Nodes auf lokaler Ebene noch vernachlässigbar sind, sollten sie auf globaler Ebene möglichst vermieden werden. Zum einen können Knoten für längere Zeit offline sein, was dazu führt, dass ein großer Teil aller Suchanfragen einfach verloren geht. Zum anderen kann eine hohe Last dazu führen, dass die betroffenen Knoten keine Bereitschaft mehr zeigen, Suchanfragen weiterzuleiten.

Neben der Erfolgsrate und der durchschnittlich kürzesten Suchpfade müssen Priorisierungsstrategien daher auch auf die Tendenz zur Bildung von Hot Spot Nodes untersucht werden. Einige Voruntersuchungen zur Bildung von Hot Spot Nodes haben gezeigt, dass sich die verschiedenen Priorisierungsstrategien für BA, FLICKR und LASTFM generell sehr ähnlich verhalten.

Mit Ausnahme von EGOCLOSE lassen sich die Priorisierungsstrategien in zwei unterschiedliche Gruppen unterteilen. HIGHDEG, BETW und CLOSE tendieren zur starken Ausbildung von Hot Spot Nodes. EGOBETW, LOWCC, WAKTIE und NODESIM hingegen zeigen dafür eine geringere Tendenz.

Stellvertretend für beide Gruppen veranschaulicht Abbildung 4.11 die absoluten Verteilungen der Suchanfragen für LASTFM am Beispiel der Priorisierungsstrategien HIGHDEG und WEAKTIE. HIGHDEG zeigt die stärkste Tendenz zur Bildung von Hot Spot Nodes. Konkret sind bei Anwendung von HIGHDEG für mehr als 85% aller Suchanfragen eine Hand voll Knoten immer involviert. Ein ganz ähnliches Verhalten zeigen die Priorisierungsstrategien CLOSE und BETW. Die Bildung der wenigen sehr hoch frequentierten Hot Spot Nodes liegt vermutlich darin begründet, dass HIGHDEG auf dem globalen Maß der Degree-Zentralität basiert. Es werden immer dieselben Knoten am stärksten priorisiert. Die Priorisierung hängt nicht von der lokalen Sicht desjenigen Knotens ab, der die Gewichtung seiner Nachbarn gerade vornimmt.

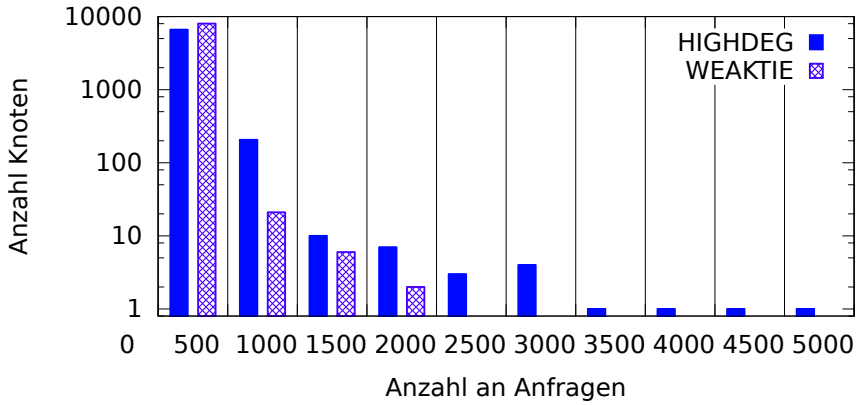


Abbildung 4.11: Verteilung der Hot Spot Nodes für LASTFM am Beispiel der Priorisierungsstrategien HIGHDEG und WEAKTIE.

Beispielsweise wird der Knoten mit der höchsten Degree-Zentralität immer von allen seinen Nachbarn als nächster Hop selektiert. Die lokale Umgebung der Nachbarn spielt dabei keine Rolle.

Als Folge dieser statischen Bevorzugung bewegt sich ein Walker immer in Richtung derselben Knoten. Insgesamt kommt es zu einer stark verzerrten Auslastung der Knoten.

Tendenziell generiert die Priorisierungsstrategie WEAKTIE zwar mehr, dafür aber weitaus weniger stark frequentierte Hot Spot Nodes. Im Falle von WEAKTIE werden an keinen Knoten mehr als 45% der Suchanfragen weitergeleitet. Die priorisierte Anordnung hängt von der lokalen Umgebung eines jeden Knotens ab. Die Auswahl des nächsten Hops erfolgt direkt auf der Basis der lokalen Berechnungen eines Knotens. Ein vergleichbares Verhalten kann für alle Priorisierungsstrategien beobachtet werden, die auf einer lokal priorisierten Anordnung der Knoten basieren.

Das Ergebnis für die Priorisierungsstrategie EGOCLOSE liegt inmitten der beiden betrachteten Gruppierungen. Diese Beobachtung lässt sich dadurch erklären, dass es sich bei EGOCLOSE im Wesentlichen um eine Kombination der Ansätze HIGHDEG und WEAKTIE handelt. Im Vergleich zu WEAKTIE führen Informationen aus dem Freundeskreis zweiter Ordnung dazu, dass die priorisierte Anordnung sich bei EGOCLOSE weniger dynamisch verhält. Dennoch ist sie noch lange nicht so statisch wie im Falle von HIGHDEG.

4.7.4 Diskussion

Sowohl für die Erfolgsraten als auch für die durchschnittlich kürzesten Suchpfade liefern zentralitätsbasierte Priorisierungsstrategien deutlich bessere Ergebnisse als die ähnlichkeitsbasierte Bevorzugung von Knoten. Die Verteilungen der Profilattribute und der Suchanfragen hat keinen signifikanten Einfluss auf die Ergebnisse. Offensichtlich überwiegt der positive Einfluss statischer Grapheigenschaften gegenüber dynamischen Effekten wie der Verteilung von Suchanfragen.

Für soziale Graphen mit geringem Clustering und hohen durchschnittlichen Knotengraden erzielt IPRW bereits beachtliche Erfolgsraten. Für hohe durchschnittliche Knotengrade eignet sich stets HIGHDEG als Priorisierungsstrategie.

Mit abnehmenden durchschnittlichen Knotengraden gewinnt das Clustering verstärkt an Bedeutung. Ein hoher Clustering-Koeffizient wirkt sich negativ auf die Erfolgsrate von HIGHDEG aus. Besitzt ein sozialer Graph hohe durchschnittliche Knotengrade und einen hohen Clustering-Koeffizienten, dann erzielt die Priorisierungsstrategie EGOBETW ähnlich gute Ergebnisse wie BETW. Diese Beobachtung ist insofern interessant, weil sich der egozentrische Ansatz ohne Vernachlässigung der Privatsphäre auf der Basis eines lokalen Index implementieren lässt. EGOBETW eignet sich auch für sehr restriktive dezentrale OSNs wie Vegas.

Für kleinere durchschnittliche Knotengrade liefert EGOCLOSE eine ähnlich gute Erfolgsrate wie HIGHDEG. EGOCLOSE beruht auf Informationen des Freundes-

kreises dritter Ordnung. Für eine Umsetzung in Vegas müsste man Abstriche in Bezug auf den Grad der Anonymität der Nutzer in Kauf nehmen.
Über die Priorisierungsstrategie WEAKTIE lassen sich keine vergleichbaren Erfolgsraten erzielen. Steht die Vermeidung von Hot Spot Nodes im Vordergrund, stellt der Ansatz jedoch eine interessante Alternative dar.
Die durchschnittliche Pfadlänge liegt bei den meisten Priorisierungsstrategien nahe am theoretischen Optimum eines gewöhnlichen Floodings. Diese Messgröße spielt bei der Entscheidung für eine Priorisierungsstrategie daher nur einer untergeordnete Rolle.

4.8 Zusammenfassung

Soziale Graphen stellen eine wertvolle Quelle für die Suche nach Informationen im WWW dar. Zum einen ermöglicht das Wissen über die Vernetzung der Nutzer die gezielte Suche nach Informationen in OSNs. Zum anderen erlaubt die Berücksichtigung sozialer Informationen eine individualisierte Filterung wie z.B. die Ergebnislisten herkömmlicher Suchmaschinen.
Aufgrund der hohen Anforderungen an die Sicherheit und den Schutz der Privatsphäre dezentraler OSNs sind die Möglichkeiten zur Umsetzung der sozialen Suche äußerst limitiert. Primär liegt es an der eingeschränkten Sicht auf den sozialen Graphen, die den Einsatz ausgereifter Weiterleitungsstrategien verhindert.
Die Bewertung blinder und informierter Suchverfahren hat gezeigt, dass sich die Kombination gewöhnlicher Random Walks mit lokalen Indizes besonders gut als Grundlage einer Weiterleitungsstrategie eignet. Abhängig von den Restriktionen eines dezentralen OSNs lässt sich die Umgebung des lokalen Index flexibel auf den Freundeskreis der gewünschten Ordnung anpassen. Um von den Kontextinformationen sozialer Graphen zu profitieren, eignen sich Weiterleitungsstrategien auf der Basis von Grapheigenschaften. Insbesondere lassen sich Zentralitätsmaße anstelle der zufälligen Priorisierung problemlos in Random Walks integrieren.
Als generisches Routing-Verfahren für dezentrale OSNs wurde der IPW-Algorithmus entwickelt. Dieser kombiniert einen gewöhnlichen Random Walk mit einem lokalen Index, der sich auf das Egonetzwerk eines Nutzers beschränkt. Um den Einfluss sozialer Kontextinformationen auf das Routing mittels IPW zu verstehen, wurden zahlreiche Priorisierungsstrategien auf der Basis von Grapheigenschaften formuliert. Diese wurden am Beispiel von Vegas untersucht.
Zum Zeitpunkt der Untersuchungen standen keine sozialen Graphen dezentraler OSNs zur Verfügung. Die Evaluation beruht auf künstlich erzeugten Graphen nach dem Erdös–Rényi- und dem Barabási–Albert-Modell. Außerdem wurden die Ansätze auf der Basis sozialer Graphen der zentralisierten OSNs Flickr und Last.fm untersucht.
Im Fokus der Evaluation standen die Erfolgsraten, die durchschnittlich kürzesten Suchpfade und die Verteilung von Suchanfragen. Für alle Untersuchungen wurde eine uneingeschränkte Bereitschaft zur Weiterleitung von Suchanfragen als gegeben vorausgesetzt.

Priorisierungsstrategien, bei denen die Auswahl des nächsten Hops auf globalem Wissen wie der soziozentrischen Closeness oder Betweenness beruht, liefern bessere Ergebnisse als Ansätze, die auf lokalem Wissen basieren. Dennoch kommen Priorisierungsstrategien wie z.B. die egozentrische Betweenness den Ergebnissen globaler Ansätze sehr nahe. Gerade für den Einsatz in dezentralen OSNs stellen sie eine sinnvolle Alternative dar.
Priorisierungsstrategien, die eine globale Messgröße, wie die Degree-Zentralität verwenden, tendieren zur Bildung von Hot Spot Nodes. Ansätze, die auf lokalem Wissen basieren, liefern oftmals schlechtere Erfolgsraten. Steht die Vermeidung von Hot Spot Nodes im Vordergrund, stellen sie dennoch eine mögliche Alternative dar.
Statische Grapheigenschaften haben einen größeren Einfluss auf das Routing als die Dynamik der Kommunikation. Man muss aber davon ausgehen, dass der Kommunikation mehr Gewicht beigemessen werden muss, wenn die Bereitschaft der Nutzer sinkt, Suchanfragen weiterzuleiten.
Die Priorisierungsstrategien wurden auf der Grundlage sehr unterschiedlicher Datensätze evaluiert. Neben dem Problem, dass es sich dabei teilweise um soziale Graphen zentralisierter OSN-Crawls handelt, beinhalten die gewonnenen Ergebnisse keine statistische Konfidenz. Um diesem und zahlreichen weiteren Problemen zu begegnen, beschäftigt sich das nächste Kapitel mit der Modellierung sozialer Interaktionsgraphen.

5 Modellierung sozialer Interaktionsgraphen

Um frühzeitig ein fehlerhaftes oder ineffizientes Verhalten neuer Netzwerkprotokolle oder Softwareanwendungen zu erkennen und zu verstehen, bietet sich die vorhergehende Untersuchung in einer Simulationsumgebung an. Mit der Evaluation verschiedener Priorisierungsstrategien für das Weiterleiten von Suchanfragen in dezentralen OSNs lieferte das vorhergehende Kapitel ein typisches Beispiel für die Notwendigkeit solcher Untersuchungen.

Im Kontext von OSNs benötigt man als Grundlage für die Evaluation einen Datensatz, der den sozialen Graphen repräsentiert. Die herkömmliche Vorgehensweise zur Herleitung besteht im Crawling des entsprechenden OSNs. Um die Vergleichbarkeit der Ergebnisse eines Simulationslaufs gewährleisten zu können, muss jeder Lauf auf demselben Datensatzes simuliert werden.

Aus Angst vor der Deanonymisierung [157, 218, 173] der Mitglieder eines OSNs existieren bisher nur wenige publizierte und ausreichend anonymisierte Datensätze [150, 217, 82, 200]. Zudem handelt es sich beim Crawling um eine sehr zeitaufwendige Prozedur. Crawler benötigen häufig mehrere Wochen oder Monate, um einen sozialen Graphen der gewünschten Größe zu generieren. Das Ergebnis ist ein verzerrtes Abbild der tatsächlichen Situation im OSN. Knoten werden vom Crawler unter Umständen stark zeitverzögert besucht, so dass sich die Struktur des rekonstruierten Graphen stets aus zeitlich voneinander abweichenden Teilzustände des tatsächlichen sozialen Graphen zusammensetzt. Da jeder Crawl immer nur die Momentaufnahme eines OSNs zu einer bestimmten Zeit repräsentiert, mangelt es den Simulationsergebnissen an statistischer Konfidenz [185]. Aufgrund zunehmender Möglichkeiten zur Einstellung der Sichtbarkeit bestimmter Profilattribute und Inhalte, wird es auch immer schwieriger, repräsentative Crawls eines OSNs zu generieren.

Ein alternativer Ansatz zur Analyse neuer Algorithmen und Protokolle beruht auf der Verwendung künstlich generierter sozialer Graphen. Zum einen besteht keinerlei Gefahr einer Deanonymisierung des sozialen Graphen, zum anderen gewährleistet die Verwendung mehrerer nach demselben Modell erzeugter sozialer Graphen die statistische Konfidenz der Ergebnisse.

Dieses Kapitel beschäftigt sich mit der Herleitung und Evaluation eines generischen Modells zur Erzeugung sozialer Interaktionsgraphen für zentralisierte und dezentrale OSNs [295]. Nach der Betrachtung verschiedener Anforderungen an das zu entwickelnde Modell werden zahlreiche Verfahren zur Erzeugung von Graphen präsentiert. Basierend auf der Kombination zweier Ansätze wird das generische Modell

zur Erzeugung sozialer Interaktionsgraphen konzipiert. Dieses wird im Detail auf den Einfluss seiner Konfigurationsparameter auf die Eigenschaften der generierten Graphen hin untersucht. Zuletzt erfolgt die Evaluation des Modells am Beispiel der Crawling-Datensätze zweier ausgewählter OSNs.

5.1 Anforderungen an ein generisches Modell zur Erzeugung sozialer Interaktionsgraphen

Die Idee zur Modellierung sozialer Interaktionsgraphen entstand bei der Analyse der Priorisierungsstrategien für Suchanfragen innerhalb dezentraler OSNs (vgl. Kap. 4). Da zu diesem Zeitpunkt keine sozialen Graphen zur Verfügung standen, erfolgten die Untersuchungen auf der Basis von Crawling-Datensätzen zentralisierter OSNs (vgl. Kap. 4.6). Dabei hat sich gezeigt, dass bereits geringe Unterschiede in den strukturellen Eigenschaften der OSNs sich relativ stark auf die Ergebnisse einer Priorisierungsstrategie auswirken können. Um weitere theoretische Überlegungen zur Struktur dezentraler OSNs überprüfen zu können, bedarf es eines flexiblen Modells zur Erzeugung sozialer Interaktionsgraphen.
Bei der Entwicklung eines Modells stehen theoretische Überlegungen zur Struktur und zum Interaktionsverhalten der Nutzer dezentraler OSNs im Vordergrund. Dazu zählen insbesondere Annahmen über die spezifischen Eigenschaften von Vegas. Da sich Vegas auf die Abbildung realer sozialer Beziehungen auf virtuelle Kontakte fokussiert, liegt die Vermutung nahe, dass sich im sozialen Graphen eher die Eigenschaften von RSNs als diejenigen zentralisierter OSNs widerspiegeln. Diesem Sachverhalt kommt bei der Konzeption des Modells eine hohe Bedeutung zu.
Im Folgenden werden die Anforderungen an ein generisches Modell [295] vorgestellt, das die Erzeugung sozialer Interaktionsgraphen mit den Eigenschaften von RSNs und OSNs erlaubt. Neben der Berücksichtigung struktureller Unterschiede zentralisierter und dezentraler OSNs stehen das Netzwerkwachstum und das Interaktionsaufkommen im Vordergrund.

5.1.1 Netzwerkwachstum

Abhängig vom angebotenen Dienst und der mit diesem verbundenen Zielgruppe existiert für jedes OSN in der Regel ein bestimmtes Muster, nach welchem der Beitritt eines neuen Nutzers erfolgt. Nahezu alle zentralisierten OSNs wie z.B. Facebook oder Twitter unterliegen dem sogenannten *Netzwerkeffekt* [147]. Dieser zeichnet sich dadurch aus, dass der wahrgenommene Nutzen eines OSNs mit ansteigender Anzahl von Mitgliedern wächst.
In der Startphase eins OSNs tritt zunächst nur eine kleine Anzahl von Nutzern dem Netzwerk bei. Eine verstärkte Nutzung führt daraufhin zu einer positiven Rückkopplung, die schließlich in einem exponentiellen Wachstum der Nutzerzahlen mündet. Nach einer maximalen Wachstumsphase sinkt die Beitrittsrate neuer Mitgliedern wieder. Abbildung 5.1 veranschaulicht die Entwicklung der Mitglie-

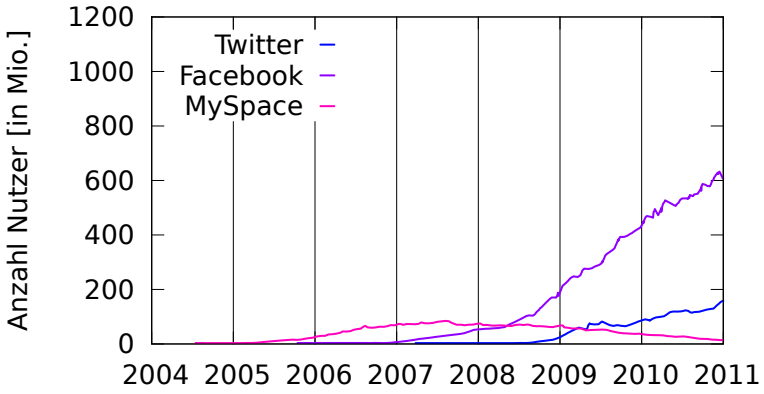


Abbildung 5.1: Wachstum und Anzahl registrierter Nutzer von Facebook, Twitter und MySpace bis zum 30. August 2011 (aufbereitet auf Grundlage von [258]).

derzahlen von Facebook, Twitter und MySpace seit der Gründung des jeweiligen OSNs bis zum 30. August 2011. Während Twitter sich im Jahr 2011 noch vor der exponentiellen Wachstumsphase bewegt, scheint Facebook diese bereits überschritten zu haben. MySpace hingegen hat sein Maximum schon längst erreicht und verliert seit 2007 kontinuierlich an Mitgliedern. Es sei darauf hingewiesen, dass obige Wachstumsprognose und die jüngsten Beitrittszahlen von Facebook nicht mehr korrelieren [238, 251].
Die Ergebnisse der Priorisierungsstrategien von Suchanfragen in dezentralen OSNs (vgl. Kap. 4.7) lassen vermuten, dass die gegenwärtige Wachstumsphase maßgeblichen Einfluss auf das Verhalten eines Algorithmus haben kann. Die meisten Modelle machen keinerlei Annahmen über die Wachstumsphasen eines Netzwerks [19, 164, 146, 117, 142, 91, 128]. Um den Einfluss solcher Phasen genauer studieren zu können, soll ein generisches Modell die Simulation des Netzwerkeffekts ermöglichen.

5.1.2 Verteilung von Knotengraden

Zahlreiche Analysen haben gezeigt, dass es sich bei OSNs meist um skaleninvariante Kleine-Welt-Netzwerke handelt [150, 217, 122]. Die Verteilung der Knotengrade folgt jeweils einem Potenzgesetz.
Dieses Verhalten gilt im Allgemeinen nicht für RSNs. Im Gegensatz zu OSNs muss man eine gewisse Zeit und Anstrengung investieren, um eine Freundschaft aufrecht zu erhalten. Derartige Einflussfaktoren werden unter dem Begriff der *wiederkehrenden Kosten* (engl: *recurring costs*) [106] zusammengefasst. In der Realität unterhält eine Person eine Vielzahl oftmals flüchtiger Bekanntschaften, von denen nur ein geringer Anteil zum engeren Freundeskreis zählt. Wiederkehrende Kosten werden jedoch nur durch engere Freundschaften verursacht. Daher existieren keine extrem hochgradigen Knoten in den sozialen Graphen von RSNs. Unter der Prämisse wiederkehrender Kosten pflegt eine Person in der Realität im Durchschnitt maximal 150 Kontakte [66]. Die Knotengrade in RSNs sind in der Regel gaußverteilt [24, 70, 6].

In Vegas ist der Blickwinkel eines Nutzers auf das eigene Egonetzwerk beschränkt. Die Gelegenheiten, Freundschaften mit Nutzern zu etablieren, die sich nicht auch im realen Leben über die Arbeit, ein gemeinsames Hobby oder eine andere gemeinsame soziale Aktivität nahe stehen, sind stark limitiert (vgl. Kap. 3.1.2). Es liegt die Vermutung nahe, dass der Kontaktgraph von Vegas ähnliche strukturelle Eigenschaften besitzt wie der eines RSNs.
Ein generisches Modell soll in der Lage sein, Knotengrade so zu verteilen, dass sie mit den Beobachtungen in gewöhnlichen zentralisierten oder mit den Annahmen für sehr restriktive dezentrale OSNs wie Vegas korrelieren.

5.1.3 Ausbildung von Gemeinschaften

Unsere Gesellschaft zeichnet sich durch die Ausbildung von Gruppen bzw. Gemeinschaften aus (vgl. Kap. 2.5.3.3). [50]. Welche Formen solche Gemeinschaften annehmen, nach welchem Muster sie sich bilden und wie deren Mitglieder miteinander interagieren, zählt zu den grundlegenden Fragestellungen bei der Erforschung von RSNs und OSNs [211, 161, 168, 160].
In diesem Zusammenhang wurde eine Vielzahl an Algorithmen entwickelt, welche die Klassifizierung [160, 72, 129] bzw. das Auffinden von Gemeinschaften [49, 163, 65, 29, 221] bewerkstelligen. Ohne konkretes Vorwissen über die Beschaffenheit einer bestimmten Gemeinschaft fällt deren Identifikation im sozialen Graphen jedoch schwer [13]. Unabhängig von der genauen Definition einer Gemeinschaft haben Untersuchung von Kollaborationsnetzwerken [164] und OSNs [13] jedoch gezeigt, dass die Wahrscheinlichkeit, einer Gemeinschaft beizutreten, sublinear mit der sich bereits in der Gemeinschaft befindlichen Anzahl von Freunden steigt. Zur Vorhersage dieser Tatsache wurden unterschiedliche theoretische Modelle entwickelt [207, 62, 112].
Neben der Anzahl gemeinsamer Freunde steht auch die Beschaffenheit schon existierender Verbindungen im Vordergrund. Beispielsweise steigt die Wahrscheinlichkeit für den Beitritt zu einer Gemeinschaft nicht nur mit der Anzahl gemeinsamer Freunde innerhalb dieser Gemeinschaft, sondern auch mit dem Grad der Vernetzung der sich schon in der Gemeinschaft befindlichen Freunde [13]. Auch ohne die Existenz bestimmter Gemeinschaften zu beachten, wurde schon früh der Einfluss gemeinsamer Freunde bei der Ausbildung einer Beziehung nachgewiesen [161].
Unabhängig davon, ob sich Gemeinschaften innerhalb eines OSNs identifizieren lassen oder nicht, soll ein generisches Modell die Möglichkeit bieten, einen Zwang zur Bildung von Gemeinschaften sowie einen Zwang zum Beitritt zu solchen Gemeinschaften auszuüben. Der Grad der vorliegenden Gemeinschaftsstruktur wird dabei über das Maß der Modularität quantifiziert (vgl. Kap. 2.5.3.3).

5.1.4 Verteilung der Netzwerkinteraktionen

Studien über OSNs haben gezeigt, dass hochgradige Knoten für den größten Anteil auftretender Interaktionen verantwortlich sind [217, 123]. Da die Verteilung

der Knotengrade in den meisten zentralisierten OSNs einem Potenzgesetz folgt, generiert ein hochgradiger Knoten im Durchschnitt auch mehr Interaktionen als ein Knoten niedrigen Grades.

Unabhängig davon, ob ein sozialer Graph mit den strukturellen Eigenschaften eines OSNs oder eines RSNs gefordert wird, soll ein generisches Modell die Möglichkeit bieten, die Korrelation zwischen Knotengraden und Interaktionen aktiv zu beeinflussen. Die Verteilung von Interaktionen auf Knoten in Abhängigkeit ihres Grades wird im Folgenden auch als *Verteilung der Netzwerkinteraktion* bezeichnet.

5.1.5 Verteilung der Knoteninteraktionen

Das Interaktionsaufkommen innerhalb eines OSNs ist stark abhängig von seiner Funktionalität. Vergleicht man beispielsweise Facebook und Twitter, so lässt sich ein sehr unterschiedliches Kommunikationsparadigma erkennen. Während man in Facebook Inhalte eines Mitglieds z.B. mit einem Pinnwandeintrag aktiv kommentiert, ermöglicht Twitter die Rundsendung von Informationen an einen Kreis interessierter Abonnenten. Neben der Betrachtung der Netzwerkinteraktionen macht es daher Sinn, die Verteilung der Interaktionen eines einzelnen Knotens auf seine unmittelbaren Nachbarn zu betrachten [217].

In OSNs existieren keine wiederkehrenden Kosten. Daher liegt die Vermutung nahe, dass ein Knoten eine Vielzahl an flüchtigen Bekanntschaften unterhält, mit denen er kaum interagiert. Bei einem dezentralen OSN wie Vegas existieren vermutlich viel weniger bzw. überhaupt keine flüchtigen Beziehungen (vgl. Kap. 5.1.2).

Ein generisches Modell soll daher die Möglichkeit bieten, auch andere Verteilungen wie z.B. eine Gleichverteilung der Interaktionen eines Knotens zu gewährleisten. Die Verteilung der Interaktionen eines Knotens mit seiner Nachbarschaft wird im Folgenden als *Verteilung der Knoteninteraktion* bezeichnet.

5.2 Existierende Ansätze zur Modellierung sozialer Graphen

Der Ansatz, OSNs auf das Interaktionsverhalten seiner Nutzer hin zu untersuchen, ist noch relativ jung. Die Mehrzahl der Modelle beschränkt sich auf die Erzeugung eines Kontaktgraphen. Die Entwicklung der meisten Ansätze fällt in eine Zeit, in der OSNs in ihrer heutigen Ausprägung noch nicht existierten.

Viele Modelle zielen auf die Repräsentation von Kollaborations-, Zitations- oder Web-Graphen ab [213, 19, 20, 21]. Obwohl die Semantik einer Bekanntschaft höchst unterschiedlich ausfallen kann, produzieren alle Modelle Graphen mit strukturellen Eigenschaften, die sich auch in RSNs bzw. OSNs wiederfinden lassen. Die meisten Ansätze lassen sich daher auch zur Erzeugung sozialer Graphen einsetzen [164].

5.2.1 Überblick über die existierenden Modelle

Bei den meisten der im Nachfolgenden vorgestellten Ansätze handelt es sich um *Wachstums-* bzw. *Evolutionsmodelle*. Während erstere einen reinen Wachstumsprozess durch das kontinuierliche Hinzufügen von Knoten und Kanten forcieren, versuchen letztere einen Evolutionsprozess zu imitieren. Neben der reinen Bildung wird hier auch das Entfernen bzw. das Absterben von Verbindungen simuliert.
Viele dieser Modelle basieren auf dem Prinzip des Preferential-Attachments [19], welches die Wahrscheinlichkeit für die Auswahl eines existierenden Knotens in Abhängigkeit seines Grades beschreibt (vgl. Kap. 4.6.3). Eine angestrebte Eigenschaft stellt die Fähigkeit zur Bildung lokaler Cluster dar, welche z.B. durch den direkten Einfluss der Nachbarschaft entstehen.
Bei der Analyse von RSNs hat man bereits früh erkannt, dass Beziehungen zwischen Personen viel häufiger auftreten, wenn sie gemeinsame Eigenschaften teilen [125]. Dieses Phänomen wird als *Homophilie* (engl: *homophily*) bezeichnet und beschreibt eine der robustesten und am häufigsten auftretenden Eigenschaften in RSNs [146].
Auf der Grundlage der Homophilie lassen sich viele Hypothesen aufstellen. Eine Erklärung für das Auftreten sozialer Korrelationen zwischen den Mitgliedern eines OSNs könnte sein, dass sich die entsprechenden Personen über OSNs zunächst nur kennenlernen, später dann aber auch im realen Leben zu engen Freunden werden [91]. Umgekehrt kann Homophilie die Wahrscheinlichkeit zur Ausbildung einer Verbindung auch reduzieren, z.B. wenn es sich um unvereinbare Profilattribute wie das Geschlecht oder die Religion eines Individuums handelt.
Getrieben von dieser Beobachtung haben sich neben Wachstums- und Evolutionsmodellen auch Modelle zur *Verbindungsvorhersage* (engl: *link prediction*) etabliert. Ganz allgemein handelt es sich um ein Link-Prediction-Modell, wenn es zum Zeitpunkt t die Wahrscheinlichkeit für die Ausbildung einer neuen Verbindung zwischen zwei Knoten zum Zeitpunkt $t + 1$ vorhersagt [93].
Im Gegensatz zu Wachstums- und Evolutionsmodellen berücksichtigen Link-Prediction-Modelle nicht nur strukturelle Eigenschaften des sozialen Graphen, sondern auch zielgruppenspezifische Attribute seiner Knoten. In den einfachsten Fällen handelt es sich um allgemeine Attribute wie z.B. Alter, Geschlecht, finanzieller Wohlstand oder religiöse Überzeugung einer Person. Neben der Homophilie spielen bei Link-Prediction-Modellen zeitliche Aspekte, wie die Häufigkeit bestimmter Interaktionen innerhalb eines festgelegten Zeitintervalls, eine Rolle. Diese Betrachtungsweise ist relativ neu. In Bezug auf OSNs existieren bisher nur wenige Ansätze, die sich dieser Charakteristik widmen [204, 2].
Viele Link-Prediction-Modelle beschränken sich bei der Vorhersage auf die Auswertung rein struktureller Attribute wie den Knotengrad oder die Anzahl gemeinsamer Nachbarn. Insofern können Link-Prediction-Modelle auch als Generalisierung von Wachstums- und Evolutionsmodellen verstanden werden. Modelle dieser Art werden zur Klasse der *eigenschaftsbasierten* Link-Prediction-Modelle gezählt [93]. Daneben lassen sich Link-Prediction-Modell in *bayessche* sowie *relationale* Wahrscheinlichkeitsmodelle und in *linear algebraische* Modelle unterteilen [93].

5.2.2 Eingrenzung der betrachteten Modelle

Das zu entwickelnde Modell soll eine geeignete Verteilung der Knotengrade, die Ausbildung von Gemeinschaften sowie die Verteilung von Netzwerk- und Knoteninteraktionen simulieren (vgl. Kap. 5.1). Wachstums- und Evolutionsmodelle beschränken sich auf die strukturellen Eigenschaften eines Graphen. Einige Modelle erreichen nur eine Skaleninvarianz [19]. Andere wiederum forcieren nur das Clustering [213]. Viele Modelle versuchen sowohl eine Verteilung der Knotengrade entsprechend eines Potenzgesetzes als auch ein hohes Clustering wie in Kleine-Welt-Netzwerken zu erreichen. Die meisten Wachstums- und Evolutionsmodelle teilen dabei die Eigenschaft, dass sie eine geringe Berechnungskomplexität aufweisen.
Link-Prediction-Modelle hingegen werden meist auf der Grundlage existierender Netzwerkdaten hergeleitet. Dabei dient der soziale Graph zum Zeitpunkt t als Trainingsdatensatz für einen Klassifikator. Das Modell selbst versucht den Zustand des sozialen Graphen zum Zeitpunkt $t + 1$ vorherzusagen.
Abhängig vom gewählten Ähnlichkeitsmaß stimmt das Ergebnis dann mehr oder weniger gut mit dem gewünschten Graphen überein. Ein Problem bei der Erzeugung eines Klassifikators auf der Grundlage der Homophilie stellt die Größe des Netzwerkes dar. Die Anzahl möglicher Kanten wächst quadratisch mit der Anzahl von Knoten, so dass eine paarweise Berechnung der Ähnlichkeit zweier Knoten nicht skaliert [93].
Für dezentrale OSNs wie Vegas existieren keine Crawling-Datensätze, die für das Training eines Klassifikators dienen können. Daher werden Link-Prediction-Modelle im Kontext dieser Arbeit zur Erzeugung sozialer Interaktionsgraphen nicht betrachtet.
Im Folgenden werden einige Modelle vorgestellt und auf ihre Anwendbarkeit für die Erzeugung sozialer Graphen hin untersucht. Diese Ansätze versuchen strukturelle Eigenschaften wie die Verteilung der Knotengrade oder das Clustering-Verhalten zu modellieren.
Da sich die meisten Modelle auf Datensätze virtueller und realer sozialer Netzwerke beziehen, wird im Folgenden einfach nur von sozialen Netzwerken gesprochen (vgl. Kap. 2.2). Nur wenn sich ein Ansatz explizit auf die Modellierung von RSNs bzw. OSNs bezieht, werden die Begriffe RSN und OSN auch explizit verwendet.

5.2.3 Random Walk Modell

Das *Random Walk Modell* (*RW-Modell*) [206] versucht, soziale Netzwerke auf der Basis eines Zufallsprozesses zu modellieren. Es basiert auf dem beobachteten Browsing-Verhalten im WWW. Zum einen erreicht man neue Webseiten über Hyperlinks, zum anderen über den Einsatz von Suchmaschinen. Das Random Walk Modell adaptiert diese Beobachtungen, indem es das Folgen von Hyperlinks als Random Walk modelliert.
Es existieren Regeln für das Hinzufügen neuer Knoten und die Ausbildung weiterer Kanten. Der konstante Parameter q bestimmt die Wahrscheinlichkeit dafür, welche der beiden Regeln angewendet werden soll. Mit der Wahrscheinlichkeit $1 - q$ wird

ein neuer Knoten u erzeugt, der sich mit einem zufällig ausgewählten Knoten v verbindet. Mit der Wahrscheinlichkeit q verbindet sich Knoten u, der seine letzte Verbindung zum Knoten v etabliert hat, zu einem zufällig selektierten Knoten $w \in \Gamma(v)$.
q entspricht der Wahrscheinlichkeit, dass der Random Walk fortgesetzt bzw. ein neuer Knoten hinzugefügt wird. Die Länge des Random Walks bestimmt also den Grad des Knotens u. Das Ergebnis ist eine Verteilung der Knotengrade, die einem Potenzgesetz folgt.
Auch die Verteilung des Clustering-Koeffizienten folgt einem Potenzgesetz. Die exakte Verteilung lässt sich über den Parameter q regulieren. Leider liefert das Modell keine schlüssige Erklärung für die Korrelation zwischen der Verteilung der Knotengrade eines Knotens u und der Knotengrade seiner Nachbarschaft $\Gamma(u)$. Es wird vermutet, dass die Wahrscheinlichkeitsverteilung nur von q, nicht aber von den Eigenschaften eines Knotens selbst abhängt [206].
Eine Erweiterung des RW-Modells stellt das *Rekursive Suchmodell* (*RS-Modell*) dar [206]. Der Random Walk wird so verändert, dass ein Knoten u zu allen Nachbarn $\Gamma(v)$ eines besuchten Knotens v eine neue Verbindung ausbildet. Für kleine Werte von q entstehen dadurch weniger hochgradige Knoten, die dafür häufiger sehr ähnliche Knotengrade aufweisen. Insgesamt nimmt damit der Anteil hochgradiger Knoten sehr schnell ab. Die Verteilung hochgradiger Knoten unterliegt einem nicht näher spezifizierten Cut-Off.
Sowohl das RW- als auch das RS-Modell generieren eine Knotengradverteilung, die einem Potenzgesetz folgt. Im RS-Modell führt die komplette Vernetzung mit der Nachbarschaft eines besuchten Knotens zur Verteilung der Knotengrade, die einem Potenzgesetz mit Cut-Off folgt. In beiden Modellen lässt sich die Verteilung der lokalen Clustering-Koeffizienten indirekt beeinflussen. Netzwerkeffekte und Interaktionen werden hingegen nicht berücksichtigt.

5.2.4 Preferential-Attachment-Modelle

Die Untersuchung zahlreicher Computernetzwerke hat gezeigt, dass ein Knoten neue Verbindungen in Proportion zu seinen schon existierenden Kanten ausbildet [5, 19, 118, 105]. Insgesamt führt dieses Verhalten zu einem multiplikativen Prozess, der eine Verteilung der Knotengrade verursacht, die einem Potenzgesetz folgt. Dieser Prozess führt zur Bildung skaleninvarianter Graphen und wird als Preferential Attachment bezeichnet (vgl. Kap. 4.6.3).

5.2.4.1 Barabási–Albert-Modell

Das *Barabási–Albert-Modell* (*BA-Modell*) wurde bereits für die künstliche Erzeugung sozialer Graphen zur Evaluation der Priorisierungsstrategien von Suchanfragen in dezentralen OSNs verwendet (vgl. Kap. 4.6.3).
Im BA-Modell verbindet sich ein neuer Knoten mit dem Graphen in Anlehnung an den *The-rich-get-richer*-Prozess [190]: Die Wahrscheinlichkeit, mit der ein neu-

er Knoten u einem existierenden Knoten v beitritt, verhält sich proportional zum Knotengrad $deg(v)$.

Die Auswahlwahrscheinlichkeit $P(deg(v))$ berechnet sich entsprechend der Gleichung $P(deg(v)) = \frac{deg(v)}{\sum_{w \in V} deg(w)}$. Die Anzahl m an Verbindungen, die ein Knoten bei seinem Beitritt etabliert, ist ein konstanter Parameter des Modells.

Die nummerische Analyse hat gezeigt, dass das BA-Modell einen skaleninvarianten Graphen erzeugt. Die Verteilung der Knotengrade folgt einem Potenzgesetz mit konstantem Koeffizienten $\gamma = 3$ [4]. Die Länge des durchschnittlich kürzesten Pfades l wächst logarithmisch mit der Anzahl der Knoten und folgt der Form $l \sim \frac{\ln |V|}{\ln \ln |V|}$ [4]. Der globale Clustering-Koeffizient liegt in der Regel deutlich über dem eines Zufallsgraphen. Im Gegensatz zu Kleine-Welt-Netzen skaliert dieser ebenfalls mit der Anzahl der Knoten. Die Verteilung der lokalen Clustering-Koeffizienten folgt annähernd einem Potenzgesetz der Form $CK(v) \sim |V|^{-0,75}$ [4].

Bezogen auf OSNs sorgt das BA-Modell dafür, dass ein neues Mitglied bevorzugt Freundschaften mit Nutzern eingeht, die sich bereits einer hohen Popularität innerhalb des Netzwerks erfreuen.

Zusammenfassend lässt sich festhalten, dass das BA-Modell eine Verteilung der Knotengrade generiert, die einem Potenzgesetz folgt. Abgesehen von der Anzahl der Knoten sind die Clustering-Eigenschaften jedoch nicht beeinflussbar. Netzwerkeffekte und Interaktionen werden vom BA-Modell nicht berücksichtigt.

5.2.4.2 Attraktivitätsmodell

Das BA-Modell ist in seiner Anwendbarkeit stark limitiert. Die Verteilung der Knotengrade folgt stets einem Potenzgesetz mit konstantem Koeffizienten $\gamma = 3$. Um eine flexible Verteilung der Knotengrade zu erzielen, führt das *Attraktivitätsmodell* (*AT-Modell*) [63] die Attraktivität als Attribut eines Knotens ein. Mit zunehmender Attraktivität eines Knotens steigt die Wahrscheinlichkeit, dass eine weitere Kante zu diesem ausgebildet wird.

Die Attraktivität A_u eines Knotens u wird mit der konstanten Attraktivität $A_0 \geq 0$ initialisiert. A_u wächst in Abhängigkeit der Anzahl eingehender Kanten $deg(u)$. Die Wahrscheinlichkeit, dass eine neue Kante zum Knoten u ausgebildet wird, ist proportional zu $A_u = A_0 + deg(u)$. Über A_0 lässt sich also steuern, inwiefern neue Kanten verstärkt mit jungen Knoten ausgebildet werden.

Im Gegensatz zum BA-Modell dürfen neue Kanten auch zwischen bereits existierenden Knoten etabliert werden. Abhängig von der Verteilung der Attraktivität führt dieser Umstand zu unterschiedlichen Koeffizienten γ.

Im Unterschied zum BA-Modell generiert das AT-Modell eine Verteilung der Knotengrade, die einem Potenzgesetz mit variablen Koeffizienten γ folgt. Die Berücksichtigung der Attraktivität hat nur einen indirekten Einfluss auf die Clustering-Eigenschaften der generierten Graphen. Netzwerkeffekte und Interaktionen werden vom AT-Modell nicht berücksichtigt.

5.2.4.3 Aktivitätsmodell

In Analogie zum AT-Modell forciert das *Aktivitätsmodell* (*AK-Modell*) [6] die aktive Reduktion hochgradiger Knoten. Dafür werden die Kosten zur Ausbildung von Kanten und das Altern von Knoten berücksichtigt.

Das AK-Modell differenziert zwischen *aktiven* und *inaktiven* Knoten. Zu inaktiven Knoten können keine weiteren Verbindungen aufgebaut werden. Neue Knoten werden zunächst als aktiv gekennzeichnet und nach Ablauf einer gewissen Zeitspanne als inaktiv markiert.

Ein Wechsel von aktiv zu inaktiv wird durch zwei verschiedene Bedingungen provoziert. Bei der *Alterungsbedingung* bestimmt eine konstante Wahrscheinlichkeit p_i, zu welchem Zeitpunkt ein Knoten inaktiv wird. Die Wahrscheinlichkeit, dass ein Knoten aktiv bleibt, nimmt mit fortschreitender Zeit exponentiell ab. Unter Berücksichtigung der *Kostenbedingung* wird ein Knoten inaktiv, sobald er eine maximale Anzahl k_{max} an Kanten ausgebildet hat.

Beide Bedingungen führen dazu, dass extrem hochgradige Knoten nicht mehr gebildet werden. Die Verteilung der Knotengrade folgt einem Potenzgesetz mit exponentialverteiltem Cut-Off. Bei hinreichend starker Einflussnahme beider Bedingungen lässt sich auch eine gaußsche Verteilung der Knotengrade erzwingen. Wie im AT-Modell sind Clustering-Eigenschaften jedoch weiterhin nicht direkt beeinflussbar. Auf Netzwerkeffekte und Interaktionen wird im AK-Modell nicht eingegangen.

5.2.4.4 Informationsfiltermodell

Das BA-Modell geht davon aus, dass ein neuer Knoten das globale Wissen über den Zustand des gesamten Graphen besitzt. Dies schließt insbesondere die Kenntnis aller Knotengrade mit ein. Um dieser unrealistischen Annahme zu begegnen, reduziert das *Informationsfiltermodell* (*IF-Modell*) [152] die Möglichkeiten zur Ausbildung neuer Kanten.

Wird ein neuer Knoten u dem Graphen hinzugefügt, darf u nur mit bestimmten existierenden Knoten $v \in U$ eine neue Kante etablieren. $U \subseteq V$ beschreibt die Menge aller Knoten, die u als „interessant“ bewertet. Im einfachsten Fall wird U rein zufällig aus V erzeugt. Innerhalb von U findet die Auswahl in Analogie zum BA-Modell statt.

Offensichtlich werden neue Kanten zufällig etabliert, wenn der Anteil $f = \frac{|U|}{|V|}$ interessanter Knoten nahe bei 0 liegt. Das Ergebnis ist ein Zufallsgraph mit einer exponentiell abfallenden Verteilung der Knotengrade. Liegt f nahe bei 1, werden Kanten entsprechend dem BA-Modell erzeugt. In diesem Fall folgt die Verteilung der Knotengrade einem Potenzgesetz. Dazwischenliegende Werte bewirken eine Verteilung entsprechend eines Potenzgesetzes mit exponentiellem Cut-Off. Die Clustering-Eigenschaften sind wie in den vorhergehenden Modellen nicht direkt beeinflussbar. Netzwerkeffekte und Interaktionen bleiben weiterhin unberücksichtigt.

5.2.4.5 Fitnessmodell

Im BA-Modell besitzen ältere Knoten automatisch eine höhere Wahrscheinlichkeit, einen hohen Grad zu erlangen. Für jeden existierenden Knoten u steigt die Anzahl seiner Verbindungen mit der Zeit. Dieses Verhalten lässt sich mit der *Konnektivität* $k_u(t) = (t/t_u)^b$ beschreiben, wobei t_u dem Zeitpunkt entspricht, an dem u zum Graphen hinzugefügt wurde. Im BA-Modell gilt $b = 0,5$ [26]. Da die ältesten Knoten das längste Zeitfenster besitzen, neue Kanten zu etablieren, erzielen sie tendenziell auch die höchste Anzahl von Verbindungen.
Das *Fitnessmodell* (*FI-Modell*) [26] begegnet dieser unrealistischen Annahme, indem es die Wahrscheinlichkeit für die Ausbildung einer neuen Kante auf einen inhärenten Qualitätsunterschied zwischen den Knoten zurückführt.
Der Fitness-Parameter η beschreibt die Fähigkeit eines Knotens, auf Kosten anderer Knoten um Verbindungen zu konkurrieren. Knoten werden nicht nur aufgrund der Anzahl ihrer bereits etablierten Verbindungen, sondern auch in Abhängigkeit ihres Fitness-Parameters ausgewählt. Dies führt im Umkehrschluss dazu, dass Knoten mit einer niedrigen Fitness trotz ihres hohen Grades im Vergleich zum gewöhnlichen BA-Modell bei der Auswahl benachteiligt werden.
In jedem Schritt wird dem Graphen ein Knoten u mit der Fitness η_u hinzugefügt. η_u berechnet sich aus einer zuvor festgelegten Verteilung $\rho(\eta_u)$. u bildet m neue Verbindungen zu bereits existierenden Knoten aus. Die Wahrscheinlichkeit $P(u, v)$, dass u sich mit einem existierenden Knoten v verbindet, hängt von der Konnektivität k_v und der Fitness η_v ab. Sie berechnet sich entsprechend der Gleichung $P(u, v) = \frac{\eta_v k_v}{\sum_{w \in V} \eta_w k_w}$.
Ein hoher Fitness-Parameter erlaubt es auch relativ jungen und wenig vernetzten Knoten, mit hoher Frequenz neue Verbindungen auszubilden. Offensichtlich können Knoten mit höherer Fitness dem Graphen später beitreten und dennoch eine stärkere Konnektivität erzielen als solche, die schon sehr früh hinzugefügt wurden. Dieser Sachverhalt lässt sich nicht für beliebige Verteilungen $\rho(\eta_u)$ beobachten. Statt einer Verteilung entsprechend eines Potenzgesetzes, führt die Wahl einer Exponentialverteilung für $\rho(\eta_u)$ zu einer verzerrten Exponentialverteilung der Knotengrade. Bisher konnte keine Systematik in der Verteilung der Knotengrade in Abhängigkeit der verwendeten Verteilung $\rho(\eta_u)$ gefunden werden.
Zusammenfassend lässt sich festhalten, dass das FI-Modell die automatische Bevorzugung älterer Knoten bei Ausbildung neuer Kanten vermeidet. Der Einsatz des Fitness-Parameters liefert weiterhin eine Verteilung der Knotengrade entsprechend eines Potenzgesetzes. Die Clustering-Eigenschaften sind wie in den vorhergehenden Modellen nicht direkt beeinflussbar. Netzwerkeffekte und Interaktionen werden vom FI-Modell nicht modelliert.

5.2.5 Clustering-Modelle

Neben der Skaleninvarianz lässt sich in sozialen Netzwerken die Tendenz zur Bildung Kleiner-Welt-Netzwerke beobachten [212, 165, 166, 167]. Diese zeichnen sich durch ihre kurzen durchschnittlichen Pfade sowie ein erhöhtes Clustering aus.

Die Länge des durchschnittlich kürzesten Pfades skaliert fast ausschließlich logarithmisch mit der Anzahl der Knoten. Eine Erklärung für das erhöhte Clustering basiert auf der Annahme, dass zwei Personen, die sich zunächst nicht kennen, sich oftmals über einen ihrer gemeinsamen Bekannten kennenlernen und später ebenfalls eine Freundschaft etablieren [212].
Die Analyse wissenschaftlicher Kollaborationsnetzwerke [165, 166, 167] hat gezeigt, dass deren Wachstum stark vom Clustering der Knoten beeinflusst wird. Es existiert ein Zusammenhang zwischen der Anzahl der Nachbarn zweier Personen u und v zum Zeitpunkt t und der Wahrscheinlichkeit, dass diese zum Zeitpunkt $t+1$ miteinander kollaborieren. Die Wahrscheinlichkeit $P(com(u,v))$, dass zwei existierende Knoten u und v eine Verbindung etablieren, wächst mit der Anzahl ihrer gemeinsamen Kollaborateure $com(u,v) = |\Gamma(u) \cap \Gamma(v)|$. Sie lässt sich mit einer kumulativen Exponentialverteilung entsprechend der Gleichung $P(com(u,v)) = 1 - \alpha \cdot e^{-\frac{com(u,v)}{L}}$ berechnen [164]. Eine Vernetzung der Knoten, die dieser Wahrscheinlichkeitsverteilung folgt, bezeichnet man als *Newman-Clustering*.
Im Folgenden werden einige Modelle vorgestellt, welche die Bildung von Netzwerken mit Kleiner-Welt-Eigenschaften forcieren. Darunter befindet sich auch ein Ansatz, der die Theorie des Preferential-Attachments mit dem Newman-Clustering kombiniert.

5.2.5.1 Connecting-Nearest-Neighbor-Modell

Mit Einführung *potentieller* Kanten liefert das *Connecting-Nearest-Neighbor-Modell* (*CN-Modell*) [206] ein mathematisches Konzept für die Theorie des Newman-Clusterings.
Zwei Knoten u und w besitzen eine potentielle Kante $e_{pot}(u,w)$, wenn sie keine gemeinsame Kante $e(u,w)$ und mindestens einen gemeinsamen Nachbarn $v \in \Gamma(u) \cap \Gamma(w)$ besitzen. Zwei Regeln beeinflussen das Wachstum des Graphen:

1. Mit der Wahrscheinlichkeit $1-p$ wird ein neuer Knoten u mit einem zufällig selektierten existierenden Knoten v verbunden. Dies führt implizit zur Ausbildung von $|\Gamma(v)| - 1$ potentiellen Kanten $e_{pot}(u,w)$ mit $w \in \Gamma(v)$.

2. Mit der Wahrscheinlichkeit p wird eine zufällig selektierte potentielle Kante $e_{pot}(u,w)$ in eine echte Kante $e(u,w)$ umgewandelt.

Das Modell lässt sich über die Anzahl der Knoten und den Parameter p konfigurieren.
Simulationen haben gezeigt, dass die Verteilung mittlerer Knotengrade einem Potenzgesetz folgt. Für $p \to 1$ nimmt der Koeffizient γ ab, was sich auch in der niedrigen Zuwachsrate neuer Knoten widerspiegelt. Umgekehrt nimmt γ für $p \to 0$ zu, was einer Erhöhung der Zuwachsrate neuer Knoten entspricht.
Für hohe Knotengrade und $\gamma < 1$ folgt die Verteilung der lokalen Clustering-Koeffizienten einem Potenzgesetz. Zudem steigt mit zunehmendem Grad eines Knotens der durchschnittliche Grad seiner Nachbarknoten.

Wie das AT-Modell generiert das CN-Modell eine Verteilung der Knotengrade, die einem Potenzgesetz mit variablen Koeffizienten γ folgt. Die Tendenz zur Bildung extrem hochgradiger Knoten bleibt weiterhin bestehen. Auch das CN-Modell ermöglicht nur eine indirekte Einflussnahme auf die Clustering-Eigenschaften der generierten Graphen. Netzwerkeffekte und Interaktionen werden vom CN-Modell nicht berücksichtigt.

5.2.5.2 Aquaintance-Network-Modell

Im Unterschied zu den bisher betrachteten Ansätzen handelt es sich beim *Aquaintance-Network-Modell* (*AN-Modell*) [59] um ein Evolutionsmodell. Verbindungen werden nicht nur gebildet, sondern auch wieder entfernt. Ausgehend von einer zu Beginn schon existierenden Anzahl von Individuen entsteht eine neue Bekanntschaft dadurch, dass ein Individuum zwei seiner Bekannten einander vorstellt. Im Modell werden dazu die folgenden beiden Schritte iterativ ausgeführt:

1. Ein zufällig selektierter Knoten u wählt zufällig zwei seiner Nachbarn v und w aus. Falls bisher noch keine Verbindung zwischen v und w existiert, wird die Kante $e(v, w)$ generiert. Falls u keine zwei Nachbarn besitzt, wählt u einen beliebigen anderen Knoten v aus und erzeugt die Kante $e(u, v)$.

2. Mit der Wahrscheinlichkeit p wird ein Knoten u zufällig ausgewählt und inklusive all seiner Verbindungen $e(u, w)$ mit $w \in \Gamma(u)$ vom Netzwerk entfernt. Zusätzlich wird ein neuer Knoten v hinzugefügt, der eine neue Verbindung mit einem zufällig ausgewählten Knoten etabliert.

Simulationen haben gezeigt, dass für $p \to 0$ die Verteilung der Knotengrade einem Potenzgesetz folgt. Für $p \to 1$ lässt sich eine Exponentialverteilung beobachten. Für $p \ll 1$ wird die Verteilung der Knotengrade durch Schritt 1 des Modells dominiert. Der Bereich, in dem die Verteilung einem Potenzgesetz folgt, wächst mit kleiner werdendem p an.
Aufgrund der durch p limitierten Lebenszeit können die Knotengrade einzelner Knoten nicht unendlich anwachsen. Dies spiegelt sich in einer Verteilung der Knotengrade entsprechend eines Potenzgesetzes mit Cut-Off wider. Für große Werte von p konkurrieren die Schritte 1 und 2, was sich in einer Exponentialverteilung der Knotengrade niederschlägt.
Betrachtet man den Zeitraum, in dem Personen sozialen Netzwerken beitreten bzw. diese wieder verlassen, so bewegt man sich tendenziell im Bereich von Jahren bzw. Jahrzehnten. Zur Modellierung sozialer Netzwerke eignen sich daher sehr kleine Werte von p. Die durch p limitierte Lebenszeit der Knoten führt zu einem stationären Zustand, der als Approximation der Situation innerhalb sozialer Netzwerke dient.
In Analogie zum IF-Modell sind mit dem AN-Modell Verteilungen möglich, die einer Exponentialfunktion oder einem Potenzgesetzes mit bzw. ohne exponentiellem Cut-Off folgen. Im Ergebnis handelt es sich um stark geclusterte Graphen mit einem kleinen durchschnittlich kürzesten Pfad. Es ist jedoch nicht möglich, den

Clustering-Koeffizienten komplett unabhängig von der Gradverteilung der Knoten zu beeinflussen. Da der modellierte Graph nicht wachsen kann, muss die gewünschte Anzahl von Knoten von Anfang an existieren. Netzwerkeffekte und Interaktionen werden vom AN-Modell nicht berücksichtigt.

5.2.5.3 Tunable-Clustering-Modell

Mit dem Konzept der *Triadenformation* erweitert das *Tunable-Clustering-Modell* (*TC-Modell*) [96] das BA-Modell um die Möglichkeit zur direkten Einflussnahme auf den Wert des globalen Clustering-Koeffizienten.
Eine Triadenformation ergibt sich wie folgt. Wird im Zuge des Preferential-Attachments zwischen einem neuen Knoten u und einem existierenden Knoten v eine neue Kante $e(u, v)$ etabliert, wird eine weitere Kante $e(u, w)$ von u zu einem zufällig ausgewählten Knoten $w \in \Gamma(v)$ gebildet. Falls u bereits mit allen Knoten $w \in \Gamma(v)$ verbunden ist, wird eine weitere Kante entsprechend des Preferential-Attachments hinzugefügt.
Im TC-Modell konkurriert die Triadenformation mit dem gewöhnlichen Preferential-Attachment. Soll dem Graphen ein neuer Knoten u mit m Kanten hinzugefügt werden, verbindet sich u in Analogie zum BA-Modell mit einem existierenden Knoten v. Mit der Wahrscheinlichkeit p wird daraufhin eine Triadenformation durchgeführt. Mit der Wahrscheinlichkeit $1-p$ wird der nächste Knoten hinzugefügt. Durchschnittlich werden pro Knoten $m_t = (m-1) \cdot p$ Triadenformationen ausgeführt. Weiterhin werden pro Knoten genau m Kanten generiert. Unabhängig vom Wert m_t liefert das TC-Modell mit dem Koeffizienten $\gamma = 3$ eine Verteilung der Knotengrade, die der des BA-Modells entspricht.
Für $m_t \to m$ lässt sich der Clustering-Koeffizient auf sein theoretisches Maximum erhöhen (vgl. Kap. 5.2.4.1). Für eine wachsende Anzahl der Knoten nähert sich der Clustering-Koeffizient zudem an einen von m_t linear abhängigen konstanten Wert an. Mit $m_t \to m$ wächst die Länge des durchschnittlich kürzesten Pfades bei konstanter Anzahl von Knoten an. Für eine konstante Wahl von m_t wächst dieser logarithmisch mit der Anzahl der Knoten.
Als Ergebnis liefert das TC-Modell eine Verteilung der Knotengrade, die einem Potenzgesetz folgt. Im Gegensatz zu den bisher betrachteten Modellen ist es zudem möglich, das Clustering-Verhalten der erzeugten Graphen direkt zu beeinflussen. Da sich die Wahrscheinlichkeit zur Verbindung weit auseinander liegender Knoten durch eine gehäufte Ausbildung transitiver Verbindungen verringert, sorgt die Triadenformation für einen längeren durchschnittlich kürzesten Pfad. Das TC-Modell berücksichtigt weder Netzwerkeffekte noch Interaktionen.

5.2.5.4 Communitites-Modell

Die meisten Wachstums- und Evolutionsmodelle generieren eine Verteilung der Knotengrade, die einem Potenzgesetz folgt. Dieses Verhalten ist in sozialen Netzwerken jedoch nicht zwangsläufig gegeben [106]. Ein abweichendes Verhalten konnte z.B. für Kollaborationsnetzwerke [6, 169] und für OSNs wie Cyworld [3],

Twitter [123] und in Bezug auf das Interaktionsaufkommen auch für Facebook [205, 11] beobachtet werden.

In RSNs verteilen sich Knotengrade tendenziell um einen bestimmten Mittelwert. Als Ursache für dieses Verhalten lassen sich die wiederkehrenden Kosten identifizieren (vgl. Kap. 5.1.2). Da die Verteilung der Knotengrade keinem Potenzgesetz folgt, liegt die Vermutung nahe, dass das Preferential-Attachment keine wesentliche Rolle beim Wachstum von RSNs spielt. Offensichtlich stellt die Fähigkeit zur Erzeugung von Clustern den dominierenden Faktor bei deren Modellierung dar.

Aus diesem Grund fokussiert sich der im Folgenden als *Communitites-Modell* (*CO-Modell*) bezeichnete Ansatz explizit auf die Bildung von Clustern. Es handelt sich um ein Evolutionsmodell mit einer zu Beginn festgelegten Anzahl von Knoten.

Die Berechnung der Wahrscheinlichkeit zur Ausbildung einer neuen Verbindung zwischen zwei Knoten u und v basiert auf der Anzahl ihrer gemeinsamen Nachbarn $com(u,v) = |\Gamma(u) \cap \Gamma(v)|$ [106]. Wiederkehrende Kosten werden durch eine maximal erlaubte Anzahl von Nachbarn modelliert. Nur solange zwischen zwei Knoten u und v regelmäßige Interaktionen auftreten, bleibt die Kante $e(u,v)$ bestehen.

Die Wahrscheinlichkeit $P(u,v)$, dass die Knoten u und v interagieren, hängt von der Gesamtanzahl der Freunde $|\Gamma(u) \cup \Gamma(v)|$ sowie der Anzahl gemeinsamer Freunde $com(u,v)$ von u und v ab. Sie berechnet sich entsprechend der Gleichung $P(u,v) = f(\Gamma(u)) \cdot f(\Gamma(v)) \cdot g(com(u,v))$. Dabei bestimmt die Funktion f jeweils den Einfluss der Anzahl von Freunden auf $P(u,v)$ und berechnet sich entsprechend der Gleichung $f(z) = \frac{1}{e^{\beta(z-z^*)}+1}$. z^* definiert die maximale Anzahl möglicher Freunde. f liefert den Wert 1,0 für kleine Werte von z und fällt stark ab für $z \to z^*$. Der Parameter β regelt die Schärfe des Abfalls. Die Funktion g basiert auf der kumulativen Exponentialverteilung für die Wahrscheinlichkeit zur Ausbildung einer neuen Kante zwischen zwei Knoten in Abhängigkeit der Anzahl ihrer gemeinsamen Nachbarn [162]. Sie berechnet sich entsprechend der Gleichung $g(m) = 1 - (1 - p_0) \cdot e^{-\alpha m}$, wobei p_0 die Wahrscheinlichkeit einer Interaktion zwischen zwei Knoten beschreibt, die bisher keine Verbindung besitzen. α regelt das Wachstum von g.

Schließlich führt das CO-Modell für jede existierende Kante $e(u,v)$ ein Kantengewicht $h(u,v)$ ein, das die Stärke einer Freundschaft simuliert. Dieses wird bei Bildung von $e(u,v)$ mit 1 initialisiert und fällt exponentiell entsprechend der Funktion $h(u,v) = e^{-k\Delta t}$ mit der Zeit ab. Tritt zwischen u und v eine Interaktion auf, wird $w(u,v)$ wieder auf 1 gesetzt. Δt entspricht der vergangenen Zeitspanne seit der letzten Interaktion.

Im Ergebnis erzeugt das CO-Modell Graphen mit zahlreichen strukturellen Eigenschaften von RSNs. Die Verteilung der Knotengrade gehorcht keinem Potenzgesetz sondern erfolgt um einen Mittelwert. Das CO-Modell besitzt die Fähigkeit, das Clustering von Knoten direkt zu beeinflussen. Das Ergebnis spiegelt sich auch in der starken Tendenz zur Ausbildung von Gemeinschaften wider. Im Gegensatz zu allen bisher betrachteten Ansätzen berücksichtigt das CO-Modell das Auftreten von Interaktionen. Netzwerkeffekte werden jedoch weiterhin nicht modelliert.

5.2.6 Beurteilung der betrachteten Modelle

Obwohl viele der vorgestellten Ansätze nicht explizit für die Modellierung sozialer Netzwerke entwickelt wurden, generiert jeder Ansatz Graphen mit strukturellen Eigenschaften, die sich zum Teil auch in RSNs bzw. OSNs wiederfinden. Im Hinblick auf die Bereitstellung eines generischen Modells erfüllt jedoch keiner der Ansätze alle geforderten Anforderungen (vgl. Kap. 5.1).
Das RW- und das RS-Modell ermöglichen eine flexible Verteilung der Knotengrade entsprechend eines Potenzgesetzes. Zudem unterstützen beide die Erzeugung von Graphen mit unterschiedlichen Clustering-Koeffizienten. Das Clustering korreliert jedoch mit der Verteilung der Knotengrade und kann im jeweiligen Modell nicht isoliert beeinflusst werden.
Alle Preferential-Attachment-Modelle unterstützen die Erzeugung von Graphen, deren Verteilung der Knotengrade einem Potenzgesetz folgt. Je nach Ausprägung generieren sie eine Verteilung mit variablen Koeffizienten, einem (exponentiellen) Cut-Off für hochgradige Knoten oder auch eine Gaußverteilung. Abgesehen von der Größe eines Graphen bietet keiner der Ansätze die Möglichkeit, unabhängig von der Verteilung der Knotengrade, Einfluss auf das Clustering-Verhalten bzw. die Tendenz zur Ausbildung von Gemeinschaften zu nehmen.
Während Preferential-Attachment-Modelle die Erzeugung skaleninvarianter Netzwerke forcieren, zielen Clustering-Modelle auf die Bereitstellung Kleiner-Welt-Netzwerke ab. Mit Ausnahme des CO-Modells generieren alle Clustering-Ansätze jedoch ebenfalls Verteilungen der Knotengrade, die einem Potenzgesetz folgen. Je nach Modell wird das Clustering durch die Gradverteilung der Knoten oder aber über einen konkreten Parameter reguliert. Im TC-Modell wird die Skaleninvarianz hingegen durch das Preferential-Attachment entsprechend des BA-Modells gesteuert. Unabhängig von der Verteilung der Knotengrade unterstützt das TC-Modell eine direkte Einflussnahme auf das Clustering-Verhalten. Mit dem CO-Modell ist es möglich, einen direkten Zwang zur Bildung von Gemeinschaften auszuüben. Das Modell unterscheidet sich von allen anderen Ansätzen insbesondere dadurch, dass sich die Knotengrade um einen bestimmten Mittelwert und nicht entsprechend eines Potenzgesetztes verteilen.
Keines der vorgestellten Modelle berücksichtigt bei der Modellierung Netzwerkeffekte. Beim überwiegenden Teil handelt es sich um einfache Wachstumsmodelle. Obwohl keines davon die Modellierung unterschiedlicher Wachstumsphasen forciert, stellt die Integration eines zeitlichen Verlaufs keine besondere Schwierigkeit dar. Mit dem AN- und dem CO-Modell wurden zwei Evolutionsmodelle diskutiert. In beiden Modellen nähert sich die Verteilung der Knotengrade jedoch einem stationären Verlauf an. Ohne entsprechende Modifikationen eignen sie sich nicht für die Modellierung von Netzwerkeffekten.
Interaktionen spielen lediglich im CO-Modell eine tragende Rolle. Konkret werden sie dort zur Simulation wiederkehrender Kosten eingesetzt. Treten über einen längeren Zeitraum hinweg keine weiteren Interaktionen auf, wird die Verbindung zwischen den betroffenen Knoten getrennt.
Für die Konzeption eines generischen Modells bietet sich eine Kombination aus

dem BA- und dem CO-Modell an. Während ersteres die Modellierung typischer Charakteristika von OSNs unterstützt, liefert letzteres die Möglichkeit zur direkten Einflussnahme auf das Clustering-Verhalten und damit zur Modellierung der Eigenschaften von RSNs.
Das folgende Kapitel beschreibt das Konzept zur Erzeugung generischer Interaktionsgraphen auf der Basis dieser Kombination.

5.3 Konzeption des generischen Modells

Das im Folgenden beschriebene generische Modell unterstützt die Erzeugung von Interaktionsgraphen sozialer Netzwerke mit den Eigenschaften von RSNs bzw. OSNs. Neben der Einflussnahme auf strukturelle Eigenschaften zentralisierter und dezentraler OSNs erlaubt dieser Ansatz die Modellierung von Netzwerkeffekten und Nutzerinteraktionen. Das Modell ist so generisch konzipiert, dass es auch in Bezug auf zukünftige Erkenntnisse über die Struktur und das Interaktionsverhalten innerhalb dezentraler OSNs wie Vegas konfiguriert werden kann. Die Umsetzung verfolgt eine unabhängige Einflussnahme auf die Verteilung der Knotengrade, das Clustering-Verhalten, die Länge des durchschnittlich kürzesten Pfades und die Bildung von Gemeinschaften.
Das Konzept kombiniert die Idee des Preferential-Attachments entsprechend des BA-Modells (vgl. Kap. 5.2.4.1) und das Newman-Clustering in Analogie zum CO-Modell (vgl. Kap. 5.2.5.4). Um eine gezielte Steuerung der unterschiedlichen Grapheigenschaften zu ermöglichen, führt das Modell einen direkten Zwang zur Ausbildung von Gemeinschaften ein. Des Weiteren modelliert es das Wachstumsverhalten entsprechend eines Netzwerkeffekts und ein konfigurierbares Interaktionsverhalten.
Das Modell unterteilt sich in eine strukturelle und eine interaktive Komponente. Während sich erstere mit der Modellierung der rein strukturellen Grapheigenschaften befasst, zielt letztere auf die Interaktionen zwischen den verschiedenen Knoten in Abhängigkeit von der Zeit ab.

5.3.1 Strukturelles Modell

Das Wachstum eines sozialen Netzwerks basiert auf dem kontinuierlichen Hinzufügen neuer Mitglieder und ihrer Verbindungen untereinander. Bezogen auf den zugrunde liegenden sozialen Graphen stellt der Beitritt eines neuen Knotens bzw. die Ausbildung einer neuen Kante eine Interaktion dar. Da sich die Auswirkungen dieser Interaktionen direkt in den Grapheigenschaften niederschlagen, wird das Netzwerkwachstum dem strukturellen Modell zugeordnet. Zudem beschreibt das strukturelle Modell, wie ein neuer Knoten einen existierenden Knoten zur Ausbildung einer Verbindung selektiert und wie weitere Kanten zwischen zwei existierenden Knoten etabliert werden.

5.3.1.1 Netzwerkwachstum

Wie bereits in Kapitel 5.1.1 erwähnt, unterliegen nahezu alle untersuchten OSNs einem Netzwerkeffekt [147]. Netzwerkeffekte lassen sich mit einer logistischen Funktion der Form

$$f(t) = \frac{G}{1 + e^{k \cdot (a-t)}} \tag{5.1}$$

modellieren [97, 44]. Eine logistische Funktion beschreibt das S-förmige Wachstum einer Population in Abhängigkeit der gegenwärtigen Populationsdichte [208]. Während G die maximale Anzahl der Knoten definiert, beschreibt a den Zeitpunkt und k den Grad des maximalen Wachstums. Abhängig von der Charakteristik eines sozialen Netzwerks regulieren diese Parameter dessen zeitlichen Wachstumsverlauf.
Die Anzahl der im Zeitintervall $[t_{i-1}; t_i]$ zu modellierenden Knoten $N(t)$ lässt sich für $G = |V|$ über die Ableitung

$$N(t_i) = \frac{G \cdot k \cdot e^{a-t_i}}{(1 + e^{k \cdot (a-t_i)})^2} \tag{5.2}$$

der Funktion 5.1 berechnen. Die Berechnung der Anzahl zu erzeugender Kanten erfolgt analog.

5.3.1.2 Auswahl eines existierenden Knotens

Für das Hinzufügen neuer Knoten kommt eine leicht modifizierte Version des BA-Modells zum Einsatz (vgl. Kap. 5.2.4.1). Die Wahrscheinlichkeit $P(deg(v))$, dass ein neuer Knoten u eine Verbindung zu einem existierenden Knoten v etabliert, berechnet sich entsprechend der Gleichung

$$P(deg(v)) = \frac{deg(v) + 1}{\sum_{w \in V} deg(w)}. \tag{5.3}$$

Das Addieren einer 1 zum Knotengrad von v dient dazu, existierende Knoten ohne Nachbarn zu berücksichtigen. Ein Knoten v verliert alle seine Nachbarn, wenn v alle Verbindungen zu seinen Freunden $\Gamma(v)$ entfernt oder umgekehrt, wenn ein Freund $w \in \Gamma(v)$ die Beziehung zu v beendet. Ein Grund dafür stellt beispielsweise ein boshaftes Verhalten von v gegenüber seinen Freunden dar. Die Addition ermöglicht es den Knoten vom Grad 0, weiterhin Freunde in das Netzwerk einzuladen.
Die genaue Anzahl neuer Knoten, die innerhalb eines Zeitschritts dem Netzwerk hinzugefügt werden, wird durch die logistische Funktion 5.1 bestimmt. Die Kombination des Preferential-Attachments mit dem Netzwerkwachstum ermöglicht die Simulation logistischen Wachstums in Abhängigkeit von der Zeit.

5.3.1.3 Erzeugung einer Gemeinschaft

Neben dem Hinzufügen neuer Knoten sollen weitere Kanten so ausgebildet werden, dass der modellierte Graph ein hohes Clustering bzw. die Tendenz zur Bildung von

Gemeinschaften aufweist. Dazu bedient sich das Modell der Idee des Newman-Clusterings [164] und verfolgt eine Umsetzung in Anlehnung an das CO-Modell (vgl. Kap. 5.2.5.4).
Die Wahrscheinlichkeit, dass zwei Knoten u und v mit der Anzahl gemeinsamer Freunde $com(u,v) = |\Gamma(u) \cap \Gamma(v)|$ eine Freundschaft eingehen, berechnet sich entsprechend der Gleichung

$$P(com(u,v)) = 1 - \alpha \cdot e^{-\frac{com(u,v)}{L}}. \tag{5.4}$$

Der Parameter L regelt, wie stark die Wahrscheinlichkeitsverteilung für die Auswahl zweier Knoten u und v mit $e(u,v) \notin \mathrm{E}$ von der Anzahl ihrer gemeinsamen Nachbarn $com(u,v)$ abhängt. Abbildung 5.2 veranschaulicht den Funktionsverlauf der Wahrscheinlichkeitsverteilungen für unterschiedliche Werte von L. Mit anwachsendem L hat die Anzahl gemeinsamer Nachbarn immer weniger Einfluss auf die Wahrscheinlichkeit für die Ausbildung einer neuen Kante. Um auch solche Knoten bei der Auswahl berücksichtigen zu können, die zuvor alle Freunde verloren haben (vgl. Kap. 5.3.1.2), erhält jede Kante $e(u,v) \notin \mathrm{E}$ unabhängig von der Anzahl gemeinsamer Nachbarn der Knoten u und v eine gewisse Grundwahrscheinlichkeit.

5.3.1.4 Auswahl einer Gemeinschaft

Bisher basiert das Modell auf einer Kombination des BA- und des CO-Modells. Weder das BA- noch das CO-Modell erlauben jedoch eine direkte, jeweils unabhängige Konfiguration der Verteilung der Knotengrade, des globalen Clustering-Koeffizienten, der Modularität oder des durchschnittlich kürzesten Pfades. Um das Modell möglichst flexibel auf die strukturellen Eigenschaften von RSNs und OSNs adaptieren zu können, führt es zusätzlich einen Zwang zur expliziten Ausbildung von Gemeinschaften ein.
In Anlehnung an die Theorie des Newman-Clusterings bewirkt dieser Zwang, dass Knoten eher dazu neigen, eine Verbindung mit Knoten aus derselben Gemeinschaft auszubilden.

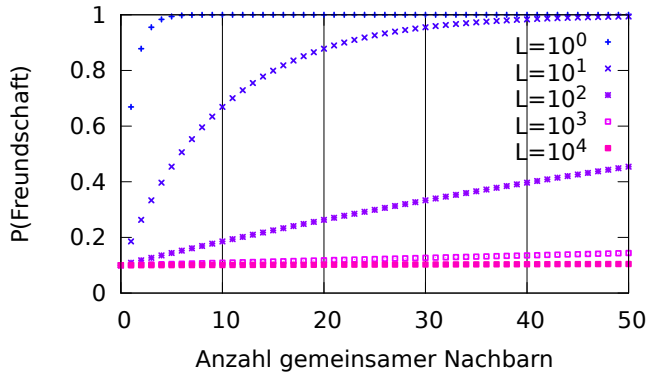


Abbildung 5.2: Unterschiedliche Wahrscheinlichkeitsverteilungen für die Ausbildung einer neuen Kante in Abhängigkeit der Anzahl gemeinsamer Nachbarn.

Eine Gemeinschaft ist in diesem Zusammenhang als abstraktes Konzept zu verstehen. Gemeinschaften werden unabhängig von der Struktur oder den homophilen Eigenschaften des Graphen definiert. Der Zwang wird durch die folgenden beiden Schritte modelliert.

Beitritt eines neuen Knotens Sei $U_i \subset G$ eine Gemeinschaft von Knoten und sei $CS(t) = \bigcup_i U_i$ die *Menge aller existierenden Gemeinschaften* (*CS*, engl: *community set*) zum Zeitpunkt t mit $U_i \cap U_j = \emptyset$ für $i \neq j$. Zu Beginn ($t = t_0$) besteht der Graph aus genau einer Gemeinschaft $U_0 = CS(t_0)$. Soll zum Zeitpunkt $t+1$ ein neuer Knoten u dem Graphen hinzugefügt werden, so tritt u mit der Wahrscheinlichkeit $P(X \geq N)$ einer existierenden Gemeinschaft $U_i \subseteq CS(t)$ bei. Mit der Wahrscheinlichkeit $P(X < N)$ erzeugt u eine neue Gemeinschaft U_{i+1} und wird ihr Mitglied. Es gilt $CS(t+1) = CS(t) \cup U_{i+1}$.
$N \in [0,0; 1,0]$ stellt einen Konfigurationsparameter des Modells dar. Der numerische Wert von N entspricht der Wahrscheinlichkeit zur Ausbildung einer neuen Gemeinschaft. Je kleiner die Wahl von N ausfällt, desto weniger Gemeinschaften werden etabliert.
Die Wahrscheinlichkeit, dass ein neuer Knoten u eine neue Gemeinschaft U_{i+1} generiert und sich mit einem existierenden Knoten $v \in U_i$ verbindet, berechnet sich entsprechend der Gleichung

$$P_{U_{i+1}}(deg(v)|X) = P(deg(v)) \cdot P(X < N). \tag{5.5}$$

Die Wahrscheinlichkeit, dass ein neuer Knoten u einer existierenden Gemeinschaft $U_i \subseteq CS(t)$ beitritt und sich zu einem existierenden Knoten $v \in U_i$ verbindet, berechnet sich entsprechend der Gleichung

$$P_{U_i}(deg(v)|X) = P(deg(v)) \cdot P(X \geq N). \tag{5.6}$$

Das Preferential-Attachment wird also an die Bedingung zum Beitritt zu einer bestimmten Gemeinschaft geknüpft.

Ausbildung einer neuen Verbindung Sei wieder $U_i \subset G$ eine Gemeinschaft von Knoten und sei $CS(t) = \bigcup_i U_i$ die Menge aller existierenden Gemeinschaften zum Zeitpunkt t mit $U_i \cap U_j = \emptyset$ für $i \neq j$. Soll zum Zeitpunkt $t+1$ eine neue Kante $e(u,v)$ zwischen zwei existierenden Knoten u und v dem Graphen hinzugefügt werden, so geschieht dies mit der Wahrscheinlichkeit $P(X \geq O)$ für $u, v \in U_i$ und $U_i \subset CS(t)$. Mit der Wahrscheinlichkeit $P(X < O)$ erfolgt die Wahl von $u \in U_i$ und $v \in U_j$ mit $U_i, U_j \subset CS(t)$ und $i \neq j$.
Die Wahrscheinlichkeit, dass eine neue Kante $e(u,v)$ zwischen zwei Knoten $u \in U_i$ und $v \in U_i$ mit $com(u,v)$ gemeinsamen Freunden gebildet wird, berechnet sich entsprechend der Gleichung

$$P_{U_{i=j}}(com(u,v)|X) = P(com(u,v)) \cdot P(X \leq O). \tag{5.7}$$

Die Wahrscheinlichkeit, dass eine neue Kante $e(u, v)$ zwischen zwei Knoten $u \in U_i$ und $v \in U_j$ für $i \neq j$ mit $com(u, v)$ gemeinsamen Freunden gebildet wird, berechnet sich entsprechend der Gleichung

$$P_{U_{i \neq j}}(com(u, v)|X) = P(com(u, v)) \cdot P(X > O). \tag{5.8}$$

Die Auswahl zweier Knoten entsprechend des Newman-Clusterings wird somit an die Bedingung der Mitgliedschaft in derselben Gemeinschaft geknüpft.

5.3.1.5 Entfernen von Knoten

Neben dem kontinuierlichen Beitritt neuer Mitglieder kommt es während der Wachstumsphase eines OSNs auch zu vereinzelten Austritten. Um dieses Verhalten zu modellieren, verwendet das Modell einen weiteren Konfigurationsparameter $E \in [0,0; 1,0]$. Dieser bestimmt in Abhängigkeit der Gesamtanzahl den Anteil an Knoten, der alle seine Verbindungen während der Wachstumsphase des Graphen verliert. Nachdem ein neuer Knoten hinzugefügt wurde, wird mit der Wahrscheinlichkeit

$$P(E) = \frac{1}{|V|} \cdot E \tag{5.9}$$

ein zufällig ausgewählter Knoten wieder entfernt.

5.3.2 Interaktives Modell

Die Aufgabe des interaktiven Modells besteht in der Simulation kontinuierlicher Interaktionen zwischen den Teilnehmern eines OSNs. Derzeit existiert kein Modell, das sich der expliziten Modellierung von Interaktionen annimmt (vgl. Kap. 5.2). Sieht man vom Prozess der Ausbildung neuer Verbindungen ab, werden lediglich im CO-Modell (vgl. Kap. 5.2.5.4) Interaktionen generiert. Sie dienen jedoch der Simulation wiederkehrender Kosten, also als Entscheidungsgrundlage dafür, ob eine existierende Verbindung aufrecht erhalten werden soll oder nicht.
Im Folgenden wird ein sehr einfacher Ansatz zur Modellierung von Interaktionen vorgestellt. Um unabhängig vom Wissen über die Homophilie einzelner Knoten zu bleiben, basiert die Bestimmung der nächsten Interaktion ausschließlich auf den rein strukturellen Eigenschaften des sozialen Graphen. Das strukturelle Modell wirkt sich somit unmittelbar auf das Verhalten des interaktiven Modells aus.
Der hier gewählte Ansatz untergliedert sich in zwei Schritte. Zunächst wird ein Knoten u ausgewählt, der die Interaktion initiiert. Danach erfolgt die Auswahl eines Nachbarn $v \in \Gamma(u)$, mit dem die Interaktion durchgeführt wird. Schließlich legt das Modell die Gesamtanzahl durchzuführender Interaktionen fest.

5.3.2.1 Auswahl eines existierenden Knotens

Wie in Kapitel 5.1.4 bereits erläutert, existiert in sozialen Netzwerken die Tendenz, dass hochgradige Knoten auch den höchsten Anteil von Interaktionen generieren.

In Anlehnung an die Auswahl eines existierenden Knotens (vgl. Kap. 5.3.1.2) werden auch Interaktionen in Abhängigkeit des Knotengrades zugewiesen. Die Wahrscheinlichkeit $P(deg(u))$, dass ein Knoten u für die Initiierung einer Interaktion ausgewählt wird, berechnet sich entsprechend der Gleichung

$$P(deg(u)) = \frac{deg(u)}{\sum_{v \in V} deg(v)}. \tag{5.10}$$

Dieser Teil des interaktiven Modells regelt die Verteilung der Netzwerkinteraktionen (vgl. Kap. 5.1.4).

5.3.2.2 Auswahl einer existierenden Kante

Nach der Selektion eines existierenden Knotens muss ein entsprechender Interaktionspartner ausgewählt werden. Wie in Kapitel 5.1.5 bereits erläutert sind mehrere Vorgehensweisen denkbar. Der hier gewählte Ansatz versucht die Verteilung der Interaktionen entsprechend den Beobachtungen in Facebook zu modellieren [217]. Jede Kante $e(u, v)$ zwischen zwei Knoten u und v erhält ein Kantengewicht $ia(u, v)$. Es beruht auf der Anzahl der bereits zwischen u und v durchgeführten Interaktionen. Die Wahrscheinlichkeit einer Interaktion zwischen u und v wächst proportional zur Anzahl bereits durchgeführter Interaktionen $ia(u, v)$. Die Wahrscheinlichkeit $P(e(u, v))$, dass eine Interaktion der Kante $e(u, v)$ zugewiesen wird, berechnet sich über die Gleichung

$$P(e(u, v)) = 1 - e^{-\frac{ia(u,v)}{Z}}. \tag{5.11}$$

Der Funktionsverlauf der Wahrscheinlichkeitsverteilung gleicht der aus Abbildung 5.2. Offensichtlich lässt sich über den Konfigurationsparameter Z der Einfluss des Kantengewichts $ia(u, v)$ regulieren. Für große Werte von Z kann auch eine Gleichverteilung erzwungen werden.
Dieser Teil des interaktiven Modells regelt die Verteilung der Knoteninteraktionen (vgl. Kap. 5.1.5).

5.3.2.3 Bestimmung der Anzahl von Interaktionen

Neben Interaktionen wie dem Versand von E-Mails, Tweets, Wall-Posts oder Kommentaren stellt auch das Schließen einer Freundschaft eine Interaktion dar. Für Facebook wurde ermittelt, dass es sich bei 45% der täglich auftretenden Interaktionen lediglich um das Schließen einer Freundschaft handelt [217]. Da sich dieses Verhältnis relativ einfach aus einem OSN-Datensatz extrahieren lässt, erfolgt die Bestimmung der Menge auftretender Interaktionen über die Gesamtanzahl von Verbindungen.
Im weiteren Verlauf werden Interaktionen, die dem Schließen einer Freundschaft entsprechen, als *Befreundungsinteraktionen* referenziert. Alle anderen Interaktionen werden als *allgemeine Interaktionen* bezeichnet. Um das passende Verhältnis an Befreundungs- und allgemeinen Interaktionen zu modellieren, führt das Modell

Parameter	*Beschreibung*
t	Bestimmt die Anzahl der Wachstumsphasen. Je kleiner t, desto mehr Knoten, Kanten und Interaktionen werden dem Graphen innerhalb einer Wachstumsphase hinzugefügt.
n	Bestimmt die Anzahl der Knoten am Ende der letzten Wachstumsphase.
e	Bestimmt die Anzahl der Kanten am Ende der letzten Wachstumsphase.
N	Bestimmt die Wahrscheinlichkeit zur Erzeugung einer neuen Gemeinschaft während des Beitritts eines neuen Knotens.
O	Bestimmt die Wahrscheinlichkeit für die Auswahl zweier Knoten derselben Gemeinschaft während der Ausbildung einer neuen Kante.
L	Beeinflusst den Verlauf der Funktion zur Berechnung der Wahrscheinlichkeitsverteilung für die Auswahl einer neu zu bildenden Kante.
E	Bestimmt den Anteil der Knoten, die alle ihre Verbindungen verlieren.
Z	Beeinflusst den Verlauf der Funktion zur Berechnung der Wahrscheinlichkeitsverteilung für die Auswahl einer Kante für die nächste Interaktion.
p	Bestimmt den Anteil der Interaktionen, die reine Befreundungsinteraktionen darstellen.

Tabelle 5.1: Auflistung aller Konfigurationsparameter des Modells.

einen letzten Parameter $p \in [0,0;1,0]$ ein. Die Gesamtanzahl I_a allgemeiner Interaktionen berechnet sich entsprechend der Gleichung

$$I_a = \frac{|E|}{p} - |E|. \tag{5.12}$$

Analog zur Anzahl der Knoten und Kanten (vgl. Kap. 5.3.1.1) berechnet sich die Anzahl allgemeiner Interaktionen $I_a(t_i)$, die im Schritt t_i dem Graphen hinzugefügt werden, entsprechend Gleichung 5.2 mit der Wahl $G = I_a$.
Tabelle 5.1 gibt noch einmal einen Überblick über alle Konfigurationsparameter des Modells.

5.4 Evaluation des generischen Modells

Die Evaluation des Modells unterteilt sich in drei Schritte. Zunächst werden die Auswirkungen der einzelnen Konfigurationsparameter auf die strukturellen Eigenschaften der generierten Graphen untersucht, bevor sich die Auswertung mit der

Parameter	G_{allg}	$G_{5:50}$	$G_{1:5}$	$G_{1:10}$
t	10	10	100	100
n	*	500	1.000	1.000
e	*	5.000	5.000	10.000
N	*	*	*	*
O	*	*	*	*
L	10	10	10	10
E	0	0	0	0
Z	100	100	100	100
p	50	50	50	50

Tabelle 5.2: Auflistung der Standardparametrisierung für die Erzeugung häufig referenzierter Graphen. Parameter, die mit dem Zeichen „*“ gekennzeichnet sind bzw. vom Standardwert abweichen, werden bei der Referenzierung mit angegeben.

Abbildung von Interaktionen beschäftigt. Anschließend erfolgt eine Analyse im Hinblick auf die Eignung zur Modellierung der Charakteristika existierender sozialer Netzwerke.

Als Kenngrößen sozialer Graphen werden die Verteilungen der Knotengrade, die Verteilung lokaler Clustering-Koeffizient, der globale Clustering-Koeffizient, die Länge des durchschnittlich kürzesten Pfades und die Modularität untersucht. Im Fokus der Untersuchungen stehen die Möglichkeiten zur Ausübung eines Zwangs auf die Bildung von Gemeinschaften, um eine dieser Kenngrößen isoliert und unabhängig von den verbleibenden Grapheigenschaften zu beeinflussen.

Im weiteren Verlauf wird auf Graphen der Klassen G_{allg}, $G_{5:50}$, $G_{1:5}$ und $G_{1:10}$ Bezug genommen. Die Klassen definieren sich durch die Standardkonfigurationen aus Tabelle 5.2. Die nicht spezifizierten Werte („*“) werden für die konkreten Ausprägungen einer Klasse explizit angegeben.

Nahezu alle Ergebnisse beruhen auf dem Mittelwert der Resultate von jeweils 30 Simulationsläufen. Auf Abweichungen davon wird gegebenenfalls ausdrücklich hingewiesen.

5.4.1 Auswirkungen der Parameter auf das strukturelle Modell

Die Evaluation des strukturellen Modells beinhaltet den Einfluss der Anzahl von Netzwerkzuständen, die Auswirkungen des Zwangs zur Bildung von Gemeinschaften und die Verteilung der Grapheigenschaften in Abhängigkeit der Anzahl von Knoten und Kanten.

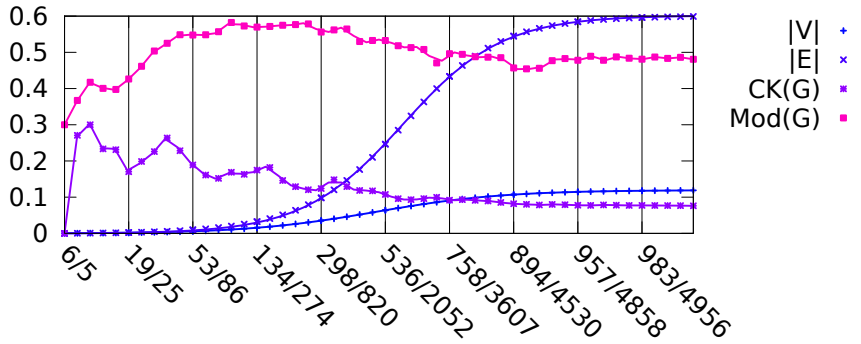
5.4.1.1 Einfluss der Anzahl von Netzwerkzuständen

Mit der Möglichkeit zur Parametrisierung der Anzahl gewünschter Netzwerkzustände unterstützt das Modell die isolierte Betrachtung einzelner Wachstumsphasen (Parameter t). Damit lässt sich die Differenz an Befreundungs- und allgemeinen Interaktionen im Intervall $[t_{i-1}; t_i]$ untersuchen.
Abbildung 5.3 veranschaulicht beispielhaft die Entwicklung des globalen Clustering-Koeffizienten, der Modularität, der Länge des durchschnittlich kürzesten Pfades, des Radius, des Durchmessers und der Anzahl von Gemeinschaften für einen Graphen der Klasse $G_{1:5}$ mit $N = 0,03$ und $O = 0,90$.
Abhängig vom gewählten Wachstumsintervall $[t_{i-1}; t_i]$ ergeben sich hier sehr unterschiedliche Werte. Die Anzahl der Gemeinschaften (vgl. Abb. 5.3(c)) steigt erwartungsgemäß kontinuierlich an. Globaler Clustering-Koeffizient und Modularität (vgl. Abb. 5.3(a)) schwanken hingegen zu Beginn sehr stark, bevor sich beide Werte zum Ende hin auf ein nahezu konstantes Niveau einpendeln. Auch die Distanzmaße (vgl. Abb. 5.3(b)) zeigen naturgemäß zu Beginn starke Schwankungen. Die Länge des durchschnittlich kürzesten Pfades pendelt sich schnell auf einen relativ konstanten Wert ein. Radius und Durchmesser zeigen auch bei abgeschwächtem Wachstum noch starke Schwankungen in ihren Werten.
Um die Robustheit und Effizienz bestimmter Algorithmen und Funktionen für soziale Netzwerke zu evaluieren, macht es offensichtlich Sinn, sie auch isoliert innerhalb unterschiedlicher Wachstumsphasen zu betrachten.
Für die finale Struktur des sozialen Graphen hat die Auswahl an Netzwerkzuständen keinen Einfluss. Die weiteren Betrachtungen struktureller Grapheigenschaften beschränken sich auf den Endzustand der modellierten Graphen.

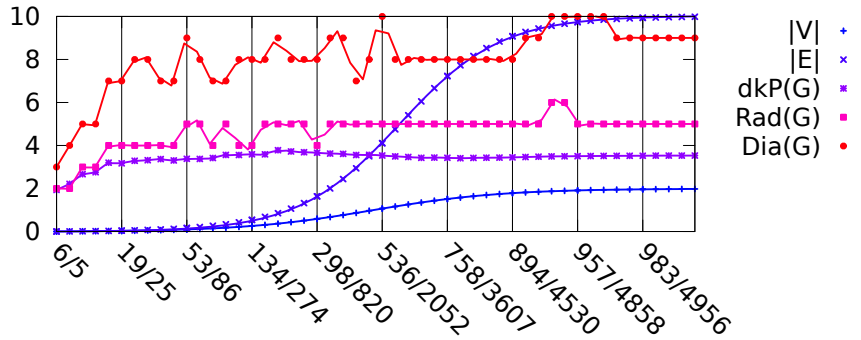
5.4.1.2 Einfluss der Auswahl von Gemeinschaften zur Ausbildung neuer Kanten

Das Modell versucht die strukturellen Grapheigenschaften durch einen Zwang zur Ausbildung von Gemeinschaften zu beeinflussen (vgl. Kap. 5.1.3). Im Folgenden werden die Auswirkungen der dafür verantwortlichen Parameter N, O und L analysiert.

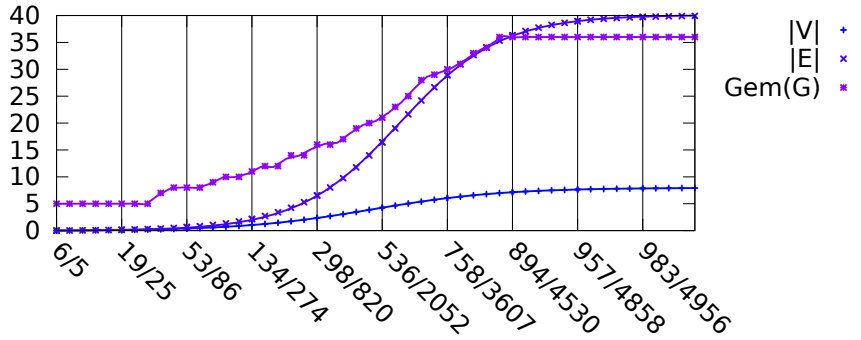
Allgemeine Beobachtungen Die Wahrscheinlichkeiten für die Erzeugung einer neuen Gemeinschaft (vgl. Gl. 5.5) sowie für den Beitritt zur eigenen Gemeinschaft (vgl. Gl. 5.6) bestimmen die Struktur der modellierten Graphen. Um einen umfassenden Überblick über das strukturelle Verhalten des Modells zu gewinnen, werden zunächst die Auswirkungen der zugehörigen Parameter N und O auf die Modularität, den globalen Clustering-Koeffizienten, den durchschnittlich kürzesten Pfad, den Radius und den Durchmesser untersucht.
Abbildung 5.4 zeigt die Verteilung dieser Grapheigenschaften für einen Graphen der Klasse $G_{5:50}$ in Abhängigkeit von N und O. Aus Gründen der Übersichtlichkeit ist jeweils nur der aus 30 Simulationsläufen berechnete Mittelwert jeder Kombination aus N und O dargestellt. Für die Modularität, den globalen Clustering-



(a) Modularität und Clustering

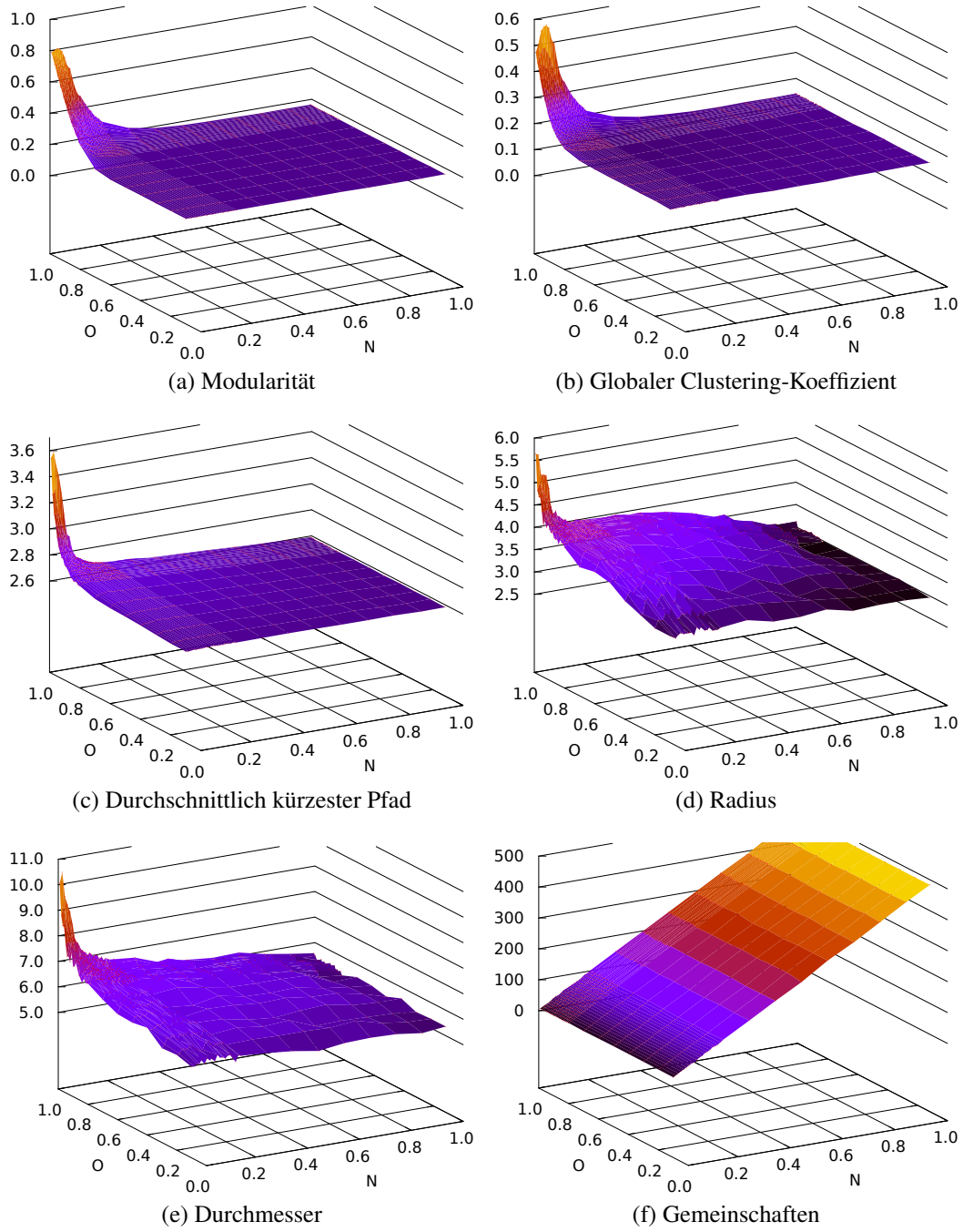


(b) Kürzeste Pfade, Radius und Durchmesser



(c) Gemeinschaften

Abbildung 5.3: Entwicklung (a) des globalen Clustering-Koeffizient und der Modularität, (b) des durchschnittlich kürzesten Pfades, des Radius und des Durchmessers sowie (c) der Anzahl von Gemeinschaften für einen Graphen der Klasse $G_{1:5}$ mit $N = 0,03$ und $O = 0,90$. Es sind 50 der 100 durchgeführten Wachstumsschritte dargestellt. Die Abszisse zeigt das gegenwärtige Knoten-zu-Kanten-Verhältnis nach je fünf Schritten. Die approximierten Kurven dienen lediglich zur besseren Visualisierung.



(a) Modularität (b) Globaler Clustering-Koeffizient

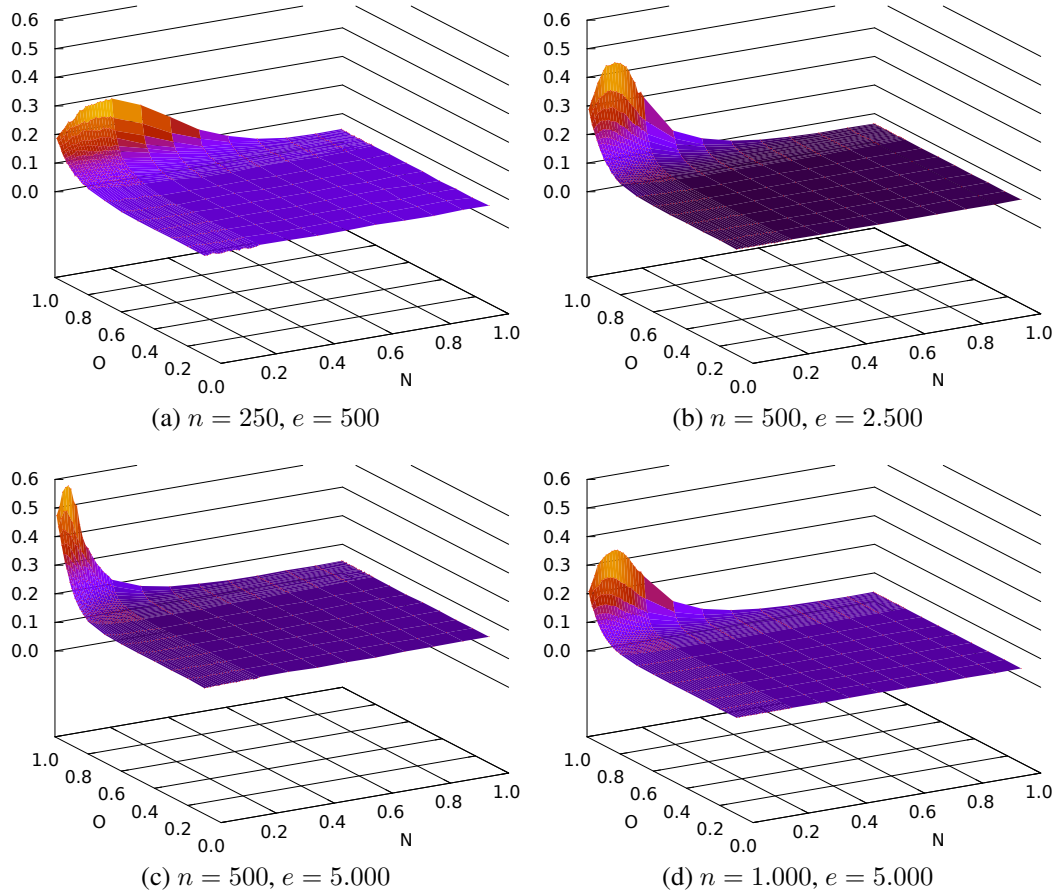
(c) Durchschnittlich kürzester Pfad (d) Radius

(e) Durchmesser (f) Gemeinschaften

Abbildung 5.4: Verteilung der Grapheigenschaften (a) Modularität, (b) globaler Clustering-Koeffizient, (c) durchschnittlich kürzester Pfad, (d) Radius, (e) Durchmesser und (f) Gemeinschaften für verschiedene Graphen der Klasse $G_{5:50}$ für $N, O \in [0,0; 1,0]$.

Koeffizient und den durchschnittlich kürzesten Pfad erkennt man sofort, dass sich der interessante Bereich durchwegs auf kleine Werte für N in Kombination mit großen Werten für O beschränkt.
Bei einer geringen Anzahl existierender Gemeinschaften ($N \in [0,0;0,2]$) führt der starke Zwang, eine Kante zu einem Knoten aus der eigenen Gemeinschaft auszubilden ($O \in [0,8;1,0]$), offensichtlich dazu, dass Knoten derselben Gemeinschaft sich viel stärker untereinander vernetzen als solche unterschiedlicher Gemeinschaften (vgl. Abb. 5.4(a)). Dies wiederum entspricht genau dem erwarteten Verhalten im Sinne der Definition der Modularität (vgl. Kap. 2.5.3.3). Für Wahrscheinlichkeiten von $N \in [0,2;1,0]$ und $O \in [0,0;0,8]$ sind keine signifikanten Unterschiede für die berechnete Modularität erkennbar.
Ein ähnliches Verhalten lässt sich auch für die Verteilung des globalen Clustering-Koeffizienten beobachten (vgl. Abb. 5.4(b)). Für $N \in [0,0;0,2]$ und $O \in [0,8;1,0]$ ergibt sich eine signifikante Steigerung.
Ein weiteres wichtiges Indiz für die Eignung des vorgeschlagenen Modells stellt die Verteilung des durchschnittlich kürzesten Pfades dar (vgl. Abb. 5.4(c)). Sie lässt sich für den betrachteten Beispielgraphen um mehr als 50% steigern.
Radius und Durchmesser besitzen nur eine bedingte Aussagekraft in Bezug auf die Charakteristik sozialer Graphen [211]. Der Grund hierfür spiegelt sich auch in den beiden Abbildungen 5.4(d) und 5.4(e) wider. Die Verteilungen der Werte beider Eigenschaften unterliegen starken Schwankungen. Dennoch lässt sich eine Zunahme beider Werte für $N \in [0,0;0,2]$ und $O \in [0,8;1,0]$ beobachten.
Je weniger Gemeinschaften der Graph besitzt, desto mehr Kanten müssen innerhalb einer Gemeinschaft ausgebildet werden. Betrachtet man eine Gemeinschaft isoliert, führt dies zu reduzierten Radien und Durchmessern innerhalb dieser Gemeinschaft. Für die Verbindung zwischen verschiedenen Gemeinschaften stehen unterdessen nur wenige Kanten zur Verfügung. Somit existieren nur wenige Pfade zwischen den einzelnen Gemeinschaften. Dies führt wiederum zu sehr großen Radien und Durchmessern. Aufgrund der starken Schwankungen ihrer Werte werden Radius und Durchmesser im weiteren Verlauf der Untersuchungen vernachlässigt.
Aus Gründen der Vollständigkeit veranschaulicht Abbildung 5.4(f) die Anzahl von Gemeinschaften in Abhängigkeit von N und O. Unabhängig von O wächst die Anzahl der Gemeinschaften proportional zu N.

Eingrenzung des Wertebereichs der Parameter N und O Für $N \in [0,2;1,0]$ und $O \in [0,0;0,8]$ weisen Modularität, globaler Clustering-Koeffizient und durchschnittlich kürzester Pfad in den Graphen der Klasse $G_{5:50}$ keine signifikanten Unterschiede auf.
Um sicherzustellen, dass die beiden Parameter generell keinen nennenswerten Einfluss auf diesen Bereich ausüben, wurden die Verteilungen dieser Grapheigenschaften für weitere Graphen unterschiedlicher Größe analysiert.
Abbildung 5.5 zeigt die Verteilung für den globalen Clustering-Koeffizienten am Beispiel von vier Graphen der Klasse G_{allg} mit unterschiedlichem Knoten-zu-Kanten-Verhältnis. Als Tendenz lässt sich erkennen, dass abhängig von der Anzahl



(a) $n = 250$, $e = 500$ (b) $n = 500$, $e = 2.500$

(c) $n = 500$, $e = 5.000$ (d) $n = 1.000$, $e = 5.000$

Abbildung 5.5: Verteilung des globalen Clustering-Koeffizienten in Abhängigkeit von N und O für vier Graphen der Klasse G_{allg} mit unterschiedlichen Knoten-zu-Kanten-Verhältnissen $n : e$.

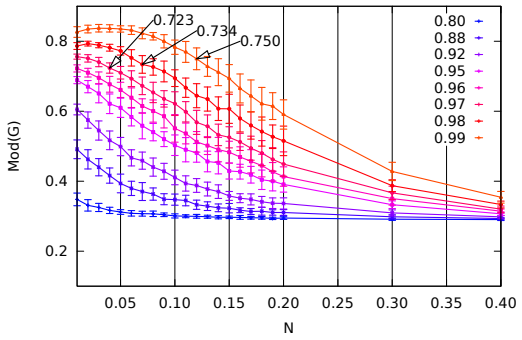
der Knoten und Kanten signifikante Unterschiede in der Steigung und der Position des Maximums existieren. Unabhängig von der Wahl von N sind für $O \in [0,0; 0,8]$ jedoch keine signifikanten Unterschiede in der Verteilung zu erkennen. Analoge Beobachtungen ergeben sich für den durchschnittlich kürzesten Pfad sowie die Modularität. Für die weiteren Betrachtungen werden nur Verteilungen für $O \in [0,8; 1,0]$ untersucht.

Korrelation von Modularität, globalem Clustering-Koeffizienten und durchschnittlich kürzestem Pfad Die Eignung des Modells zur Erzeugung unterschiedlicher sozialer Graphen misst sich an seiner Fähigkeit, unabhängig und selektiv auf die Modularität, den globalen Clustering-Koeffizienten sowie den durchschnittlich kürzesten Pfad Einfluss zu nehmen (vgl. Kap. 5.3.1.4). Konkret gilt es herauszufinden, ob es das Modell erlaubt, den Wert einer einzelnen Grapheigenschaft zu fixieren bzw. auf ein kleines Intervall zu beschränken und gleichzeitig den Wert einer der verbleibenden Grapheigenschaften stark zu variieren.

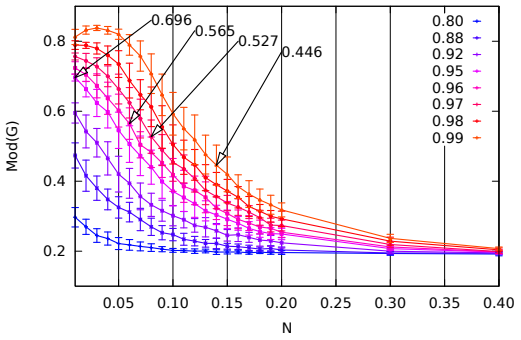
Abbildung 5.6 zeigt die Ergebnisse der entsprechenden Untersuchungen für Graphen der Klassen $G_{1:5}$ und $G_{1:10}$ für $N \in [0,0;0,2]$ und $O \in [0,8;1,0]$. Es ist jeweils nur der interessante Wertebereich von N dargestellt. Aus Gründen der Übersichtlichkeit beschränken sich die Darstellungen zudem auf eine Teilmenge der Werte von O.
Zunächst lässt sich festhalten, dass sich derselbe Wert für eine bestimmte Grapheigenschaft über unterschiedliche Konfigurationen von N und O selektieren lässt. Wird für einen Graphen der Klasse $G_{1:5}$ beispielsweise eine Länge des durchschnittlich kürzesten Pfades von ca. 4 gefordert, so lässt sich dies mit unterschiedlichen Konfigurationen von N und O bewerkstelligen (vgl. Abb. 5.6(e)). Abhängig von den gewünschten Werten für die anderen Grapheigenschaften wie z.B. den globalen Clustering-Koeffizienten (vgl. Abb. 5.6(c)) bietet sich hier die Wahl $N = 0,04$ und $O = 0,97$, $N = 0,07$ und $O = 0,98$ bzw. $N = 0,12$ und $O = 0,99$ an.
Die Abbildungen 5.6(b), 5.6(d) und 5.6(f) illustrieren ein weiteres Beispiel für Graphen der Klasse $G_{1:10}$. Um einen globalen Clustering-Koeffizienten von ca. 0,24 zu erzielen, bietet sich je nach gewünschter Modularität z.B. einer der Kombinationen $N = 0,01$ und $O = 0,95$, $N = 0,06$ und $O = 0,96$, $N = 0,08$ und $O = 0,97$ bzw. $N = 0,14$ und $O = 0,99$ an.
Natürlich lässt sich nicht jede beliebige Anforderung an die Eigenschaften eines Graphen durch die Parametrisierung mit N und O erfüllen. Dennoch erlaubt das Modell eine gezielte Einflussnahme auf die unterschiedlichen Grapheigenschaften.

Korrelation von lokalen Clustering-Koeffizienten und Knotengraden

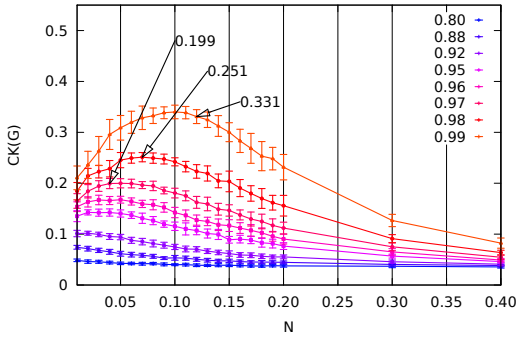
Um die Auswirkungen des Zwangs zur Bildung von Gemeinschaften besser zu verstehen, werden im Folgenden die Verteilungen der lokalen Clustering-Koeffizienten und der Knotengrade im Detail betrachtet.
Abbildung 5.7 veranschaulicht für die beiden Ausprägungen $G_{1:5}^{NO_1}$ und $G_{1:5}^{NO_2}$ der Klasse $G_{1:5}$ sowie für die Ausprägungen $G_{1:10}^{NO_1}$ und $G_{1:10}^{NO_2}$ der Klasse $G_{1:10}$ die Verteilung der Knotengrade, der lokalen Clustering-Koeffizienten und der lokalen Clustering-Koeffizienten als Funktion der Knotengrade. N und O wurden so gewählt, dass sich für die Beispielgraphen sehr ähnliche globale Clustering-Koeffizienten ergeben. Tabelle 5.3 veranschaulicht die Wahl von N und O sowie die ermittelten Eigenschaften der vier Graphen.
Modularität, globaler Clustering-Koeffizient und durchschnittlich kürzester Pfad der vier Kombinationen liegen jeweils im Konfidenzintervall der im vorhergehenden Kapitel ermittelten Ergebnisse (vgl. Abb. 5.6).
$G_{1:5}^{NO_1}$ und $G_{1:5}^{NO_2}$ besitzen nahezu den gleichen globalen Clustering-Koeffizienten, unterscheiden sich aber stark in der berechneten Modularität ($Mod(G_{1:5}^{NO_1}) = 0,687$, $Mod(G_{1:5}^{NO_2}) = 0,824$). Dieser Effekt wird offensichtlich durch den Zwang zur Bildung von Gemeinschaften provoziert. Während $G_{1:5}^{NO_1}$ 103 Gemeinschaften etabliert, sind es bei $G_{1:5}^{NO_2}$ lediglich 21. In $G_{1:5}^{NO_2}$ werden also weniger, dafür aber größere Cluster mit mehr Verbindungen innerhalb einer Gemeinschaft gebildet als in $G_{1:5}^{NO_1}$. Gleichzeitig treten in $G_{1:5}^{NO_2}$ weniger Verbindungen zwischen verschiedenen Gemeinschaften auf als in $G_{1:5}^{NO_1}$.



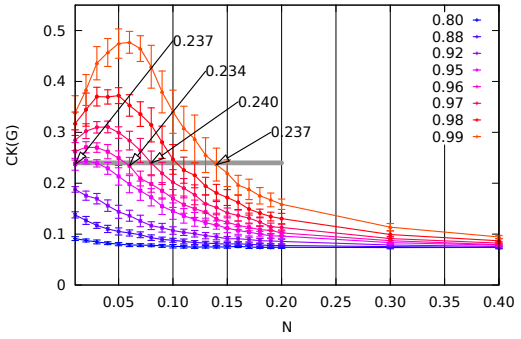
(a) Modularität von $G_{1:5}$



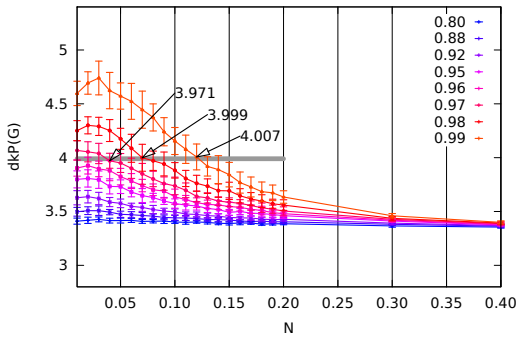
(b) Modularität von $G_{1:10}$



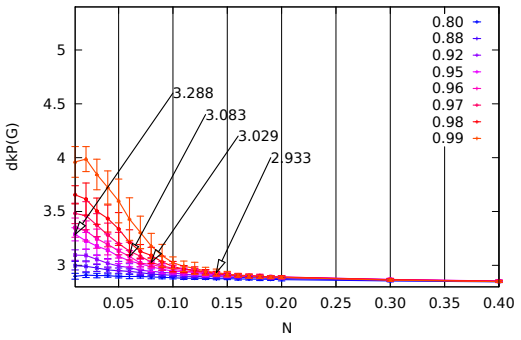
(c) Globaler Clustering-Koeffizient von $G_{1:5}$



(d) Globaler Clustering-Koeffizient von $G_{1:10}$



(e) Durchschnittlich kürzester Pfad von $G_{1:5}$



(f) Durchschnittlich kürzester Pfad von $G_{1:10}$

Abbildung 5.6: Verhalten von (a) – (b) Modularität, (c) – (d) globalem Clustering-Koeffizienten und (e) – (f) durchschnittlich kürzestem Pfad in Abhängigkeit von N und O für Graphen der Klassen $G_{1:5}$ und $G_{1:10}$ mit $N \in [0,0;0,2]$ und $O \in [0,8;1,0]$.

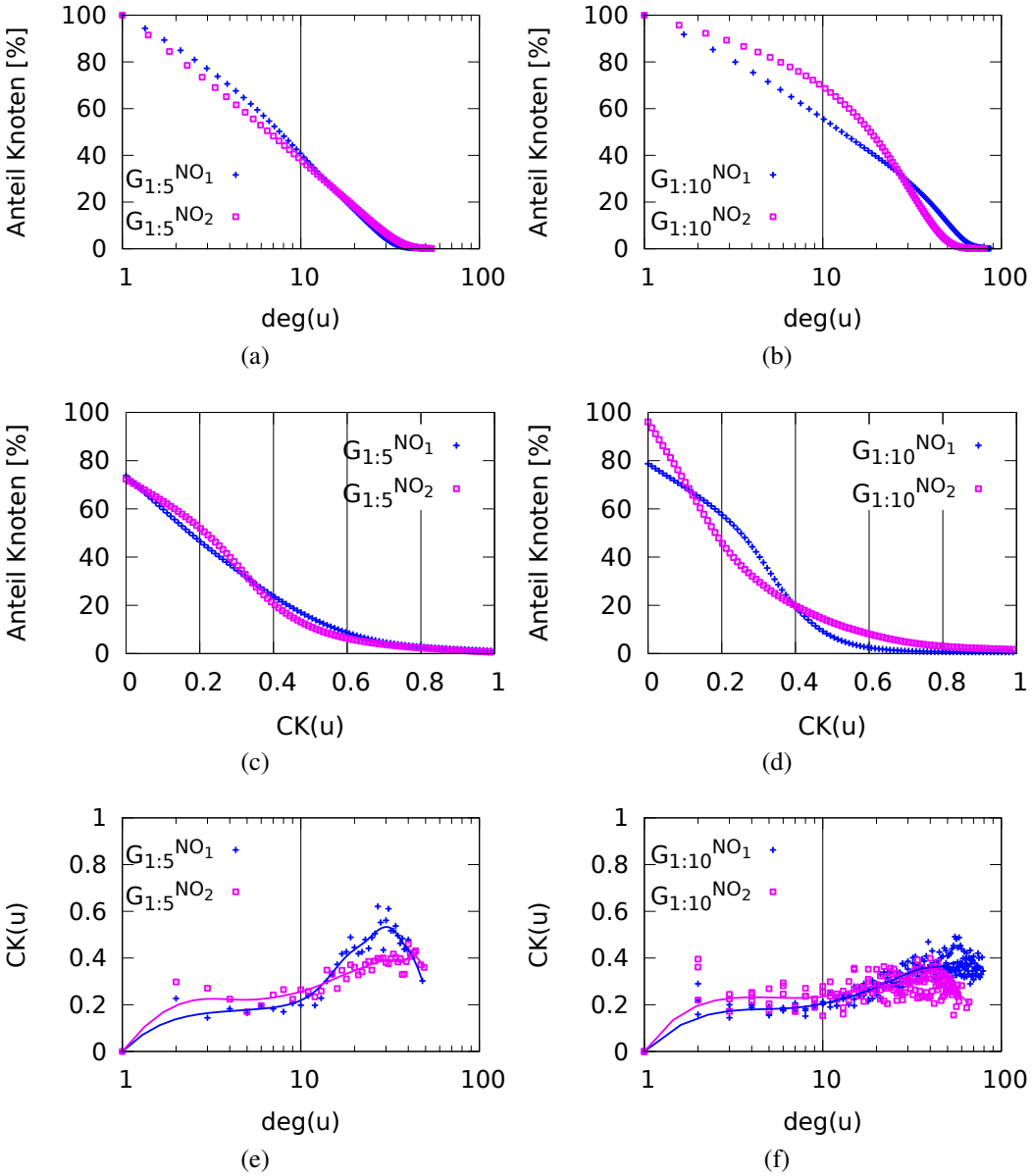


Abbildung 5.7: (a),(b) Komplementär kumulative Häufigkeitsverteilung der Knotengrade, (c),(d) der lokalen Clustering-Koeffizienten und (e),(f) Verteilung lokaler Clustering-Koeffizienten als Funktion der Knotengrade am Beispiel je zweier Graphen der Klassen $G_{1:5}$ und $G_{1:10}$.

	N	O	CK	Mod	Gem	dkP	Rad	Dia
$G_{1:5}^{NO_1}$	0,11	0,98	0,237	0,687	103	3,83	6	9
$G_{1:5}^{NO_2}$	0,02	0,99	0,238	0,824	21	4,61	6	11
$G_{1:10}^{NO_1}$	0,01	0,95	0,236	0,703	11	3,26	5	7
$G_{1:10}^{NO_2}$	0,14	0,99	0,242	0,452	133	2,93	4	7

Tabelle 5.3: Auflistung der Grapheigenschaften für $G_{1:5}^{NO_1}$ und $G_{1:5}^{NO_2}$ sowie $G_{1:10}^{NO_1}$ und $G_{1:10}^{NO_2}$. Die Graphen wurden so parametrisiert, dass sie jeweils ähnliche globale Clustering-Koeffizienten aufweisen.

Dieser Effekt lässt sich direkt an den Verteilungen der Knotengrade und der lokalen Clustering-Koeffizienten nachvollziehen. Zum einen zeigt $G_{1:5}^{NO_2}$ eine reduzierte Bildung niedriggradiger und eine gehäufte Bildung hochgradiger Knoten (vgl. Abb. 5.7(a)), zum anderen kann man in $G_{1:5}^{NO_2}$ eine Verdichtung des Anteils an Knoten mit kleinen bis mittleren Clustering-Koeffizient beobachten (vgl. Abb. 5.7(c)). Um den gleichen globalen Clustering-Koeffizienten zu erreichen, kompensiert $G_{1:5}^{NO_1}$ beide Effekte durch die Bildung eines kleineren Anteils hochgradiger Knoten, der wiederum sehr große Clustering-Koeffizienten aufweist (vgl. Abb. 5.7(e)).

Da in $G_{1:5}^{NO_2}$ weniger Verbindungen zwischen Knoten verschiedener Gemeinschaften ausgebildet werden als in $G_{1:5}^{NO_1}$, existieren insgesamt auch weniger Pfade zwischen solchen Knoten. Alle Pfade zwischen den Knoten verschiedener Cluster verlaufen stets über bestimmte Verbindungsknoten. Im Falle von $G_{1:5}^{NO_2}$ macht sich dies in einem größeren Wert für den durchschnittlich kürzesten Pfad bemerkbar.

Auch für $G_{1:10}^{NO_1}$ und $G_{1:10}^{NO_2}$ lässt sich der große Modularitätsunterschied ($Mod(G_{1:10}^{NO_1}) = 0,703$, $Mod(G_{1:10}^{NO_2}) = 0,452$) auf den Zwang zur Bildung von Gemeinschaften zurückführen. Im Gegensatz zu $G_{1:5}^{NO_1}$ und $G_{1:5}^{NO_2}$ wirkt sich dieser hier genau entgegengesetzt aus. $G_{1:10}^{NO_1}$ bildet 11, $G_{1:10}^{NO_2}$ 133 Gemeinschaften. $G_{1:10}^{NO_1}$ und $G_{1:10}^{NO_2}$ tauschen in der Verteilung der Knotengrade (vgl. Abb. 5.7(b)) und der Clustering-Koeffizienten (vgl. Abb. 5.7(d)) daher ihre Plätze.

Aufgrund des kleineren Knoten-zu-Kanten-Verhältnisses fällt die Reduktion niedriggradiger und die Häufung hochgradiger Knoten sowie die Verdichtung des Anteils an Knoten mit kleinen bis mittleren Clustering-Koeffizienten noch stärker aus als bei $G_{1:5}^{NO_1}$ und $G_{1:5}^{NO_2}$. Durch die erhöhte Anzahl von Kanten nimmt insgesamt auch die Anzahl von Knoten mit hohen Graden zu. Dieses Verhalten spiegelt sich ebenfalls in der Verteilung der lokalen Clustering-Koeffizienten als Funktion der Knotengrade wider (vgl. Abb. 5.7(f)).

Insgesamt lässt sich festhalten, dass das Modell durch geeignete Wahl von N und O eine isolierte Einflussnahme auf einzelne Grapheigenschaften erlaubt. Zudem ermöglicht es Verteilungen der Knotengrade entsprechend eines Potenzgesetzes sowohl mit ($G_{1:10}^{NO_2}$) als auch ohne ($G_{1:10}^{NO_1}$) Cut-Off.

5.4.1.3 Einfluss des Newman-Clusterings zur Ausbildung neuer Kanten

Die Auswahl eines Beitrittsknotens wird durch das Preferential-Attachment in Kombination mit dem Zwang zur Ausbildung einer neuen Gemeinschaft gesteuert. Soll eine Kante zwischen zwei existierenden Knoten generiert werden, kommt hingegen das Newman-Clustering in Kombination mit dem Zwang zum Befreunden mit einem Knoten der eigenen Gemeinschaft zum Tragen (vgl. Kap. 5.3.1). Die Wahrscheinlichkeit für die Auswahl einer Kante zwischen zwei existierenden Knoten berechnet sich entsprechend Gleichung 5.4.

Der Parameter L steuert, wie stark die Anzahl gemeinsamer Freunde $com(u, v)$ von zwei Knoten u und v die Ausbildung einer neuen Kante $e(u, v)$ beeinflusst (vgl. Abb. 5.2). Für die bisherigen Betrachtungen wurde L konstant mit dem Wert 10 initialisiert (vgl. Tab. 5.2).

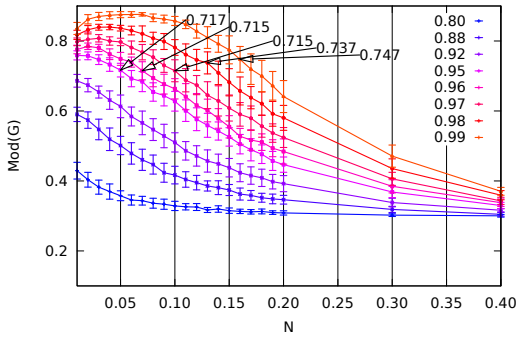
Um die Auswirkungen der Wahl von L auf die Modularität, den globalen Clustering-Koeffizienten sowie den durchschnittlich kürzesten Pfad zu analysieren, wurden Graphen der Klassen $G_{1:5}$ und $G_{1:10}$ mit $L \in \{10^0, 10^1, \ldots, 10^5\}$ generiert. Abbildung 5.8 veranschaulicht die Ergebnisse für $G_{1:5}^{L_1}$ mit $L = 1$ und $G_{1:5}^{L_2}$ mit $L = 10^5$ in Abhängigkeit von N und O.

Offensichtlich führt eine kleine Wahl von L zur Erhöhung der Werte aller dargestellten Grapheigenschaften. Diese Beobachtung gilt unabhängig von der Wahl von N und O. Interessiert man sich beispielsweise für Graphen der Klasse $G_{1:5}$ mit einem durchschnittlich kürzesten Pfad von ca. 4 Hops und einem globalen Clustering-Koeffizienten von ca. 0,2, dann lässt sich dies mit der Wahl $L = 1$ nicht realisieren (vgl. Abb. 5.8(c)). Mit der Wahl $L = 10^5$ hingegen kommt man dem gewünschten globalen Clustering-Koeffizienten bereits sehr nahe (vgl. Abb. 5.8(d)).

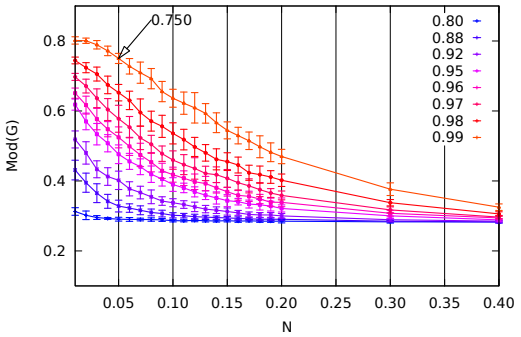
Zur weiteren Erläuterung veranschaulicht Abbildung 5.9 für $G_{1:5}^{L_1}$ und $G_{1:5}^{L_2}$ am Beispiel von $N = 0,11$ und $O = 0,98$ die Verteilung der Knotengrade (vgl. Abb. 5.9(a)), der lokalen Clustering-Koeffizienten (vgl. Abb. 5.9(b)) und die Verteilung lokaler Clustering-Koeffizienten (vgl. Abb. 5.9(c)) als Funktion der Knotengrade.

Für $G_{1:5}^{L_2}$ erfolgt die Auswahl eines Nachbarn zur Bildung einer neuen Kante nahezu gleichverteilt (vgl. Abb. 5.2). Die Anzahl gemeinsamer Freunde hat in diesem Fall keinen Einfluss auf die Auswahl einer Kante. Je größer die Wahl von L ausfällt, desto stärker wird die Bildung von Gemeinschaften durch O bestimmt. Dies spiegelt sich in der Verteilung der Knotengrade wider (vgl. Abb. 5.9(a)). Gegenüber $G_{1:5}^{L_1}$ besitzt $G_{1:5}^{L_2}$ aufgrund der nahezu gleichverteilten Auswahl einen höheren Anteil niedriggradiger und einen reduzierten Anteil hochgradiger Knoten.

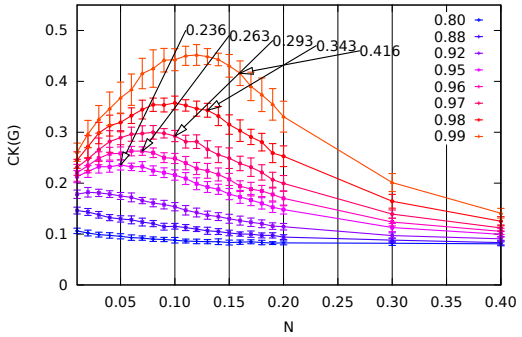
Primär bestimmt das Preferential-Attachment die Verteilung der Knotengrade. Besonders stark wird davon die Verteilung der lokalen Clustering-Koeffizienten beeinflusst (vgl. Abb. 5.9(b)). Die nahezu gleichverteilte Wahrscheinlichkeit für die Auswahl einer neuen Kante führt in $G_{1:5}^{L_2}$ zu einer ausgeglicheneren Vernetzung und somit zu einer Reduktion stark geclusterter Knoten. Auch die Verteilung lokaler Clustering-Koeffizienten als Funktion der Knotengrade verdeutlicht diesen Effekt (vgl. Abb. 5.9(c)). Durch die nahezu gleichverteilte Wahrscheinlichkeit für die Aus-



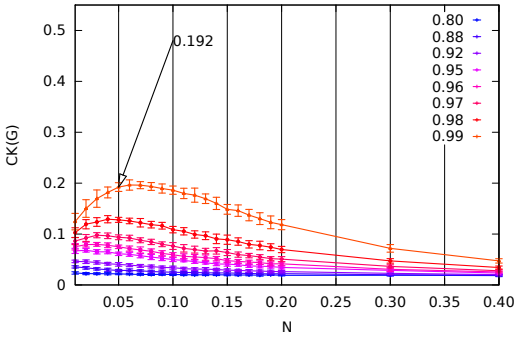
(a) Modularität von $G^{L_1}_{1:5}$



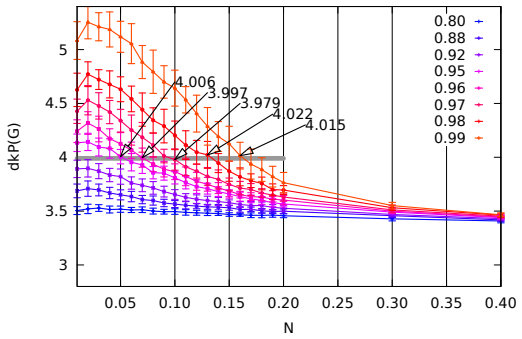
(b) Modularität von $G^{L_2}_{1:5}$



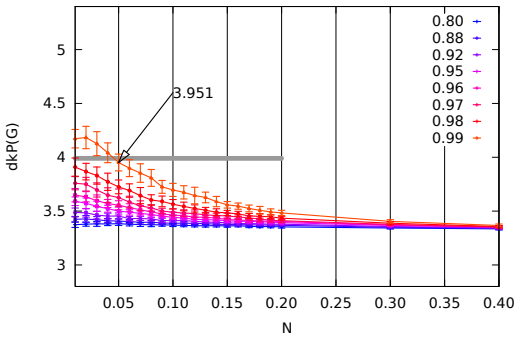
(c) Globaler Clustering-Koeffizient von $G^{L_1}_{1:5}$



(d) Globaler Clustering-Koeffizient von $G^{L_2}_{1:5}$



(e) Durchschnittlich kürzester Pfad von $G^{L_1}_{1:5}$



(f) Durchschnittlich kürzester Pfad von $G^{L_2}_{1:5}$

Abbildung 5.8: Verhalten von (a) – (b) Modularität, (c) – (d) globalem Clustering-Koeffizienten und (e) – (f) durchschnittlich kürzestem Pfad in Abhängigkeit von N und O für $G^{L_1}_{1:5}$ und $G^{L_2}_{1:5}$.

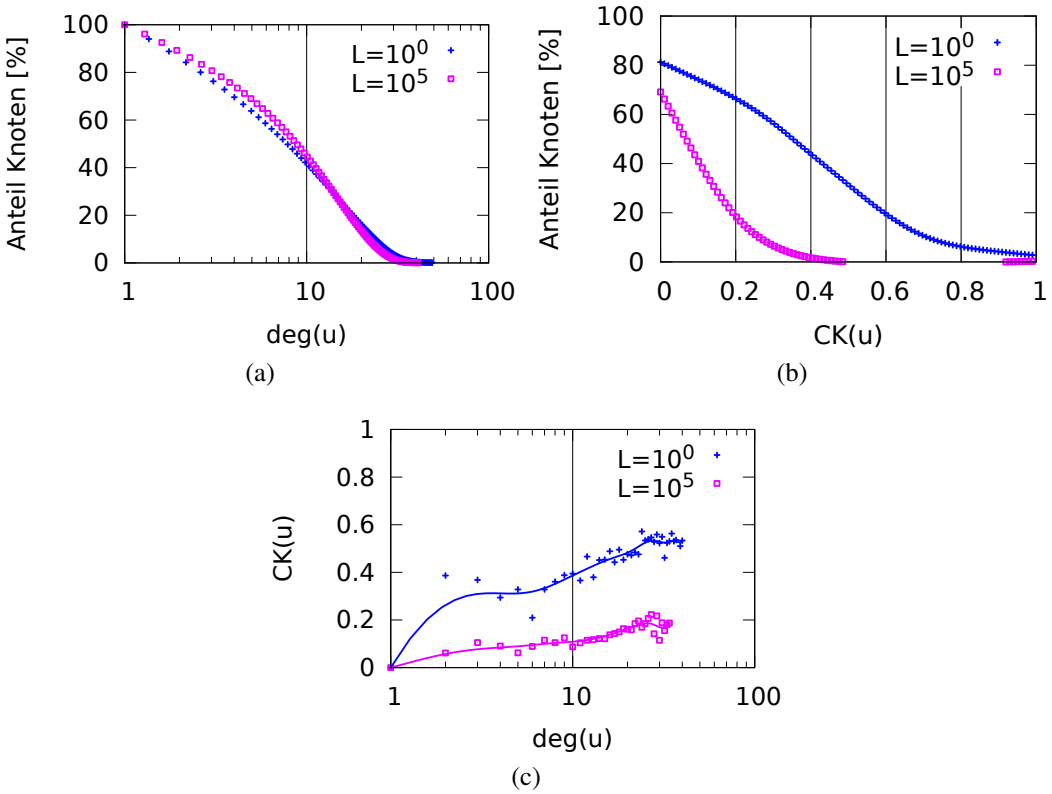


Abbildung 5.9: (a) Komplementär kumulative Häufigkeitsverteilungen der Knotengrade, (b) der lokalen Clustering-Koeffizienten und (c) Verteilung lokaler Clustering-Koeffizienten als Funktion der Knotengrade für $G_{1:5}^{L_1}$ ($L = 10^0$) und $G_{1:5}^{L_2}$ ($L = 10^5$).

wahl einer neuen Kante reduziert sich der Clustering-Koeffizient für Knoten aller Grade.

Über den Parameter L lässt sich also steuern, wie stark die Anzahl gemeinsamer Freunde innerhalb bzw. außerhalb einer Gemeinschaft die Bildung einer neuen Kante beeinflusst.

Insgesamt lässt sich festhalten, dass sich über den Parameter L alle Grapheigenschaften auf einmal verstärken bzw. abschwächen lassen.

5.4.1.4 Verhalten der Verteilung von Grapheigenschaften in Abhängigkeit der Anzahl von Knoten und Kanten

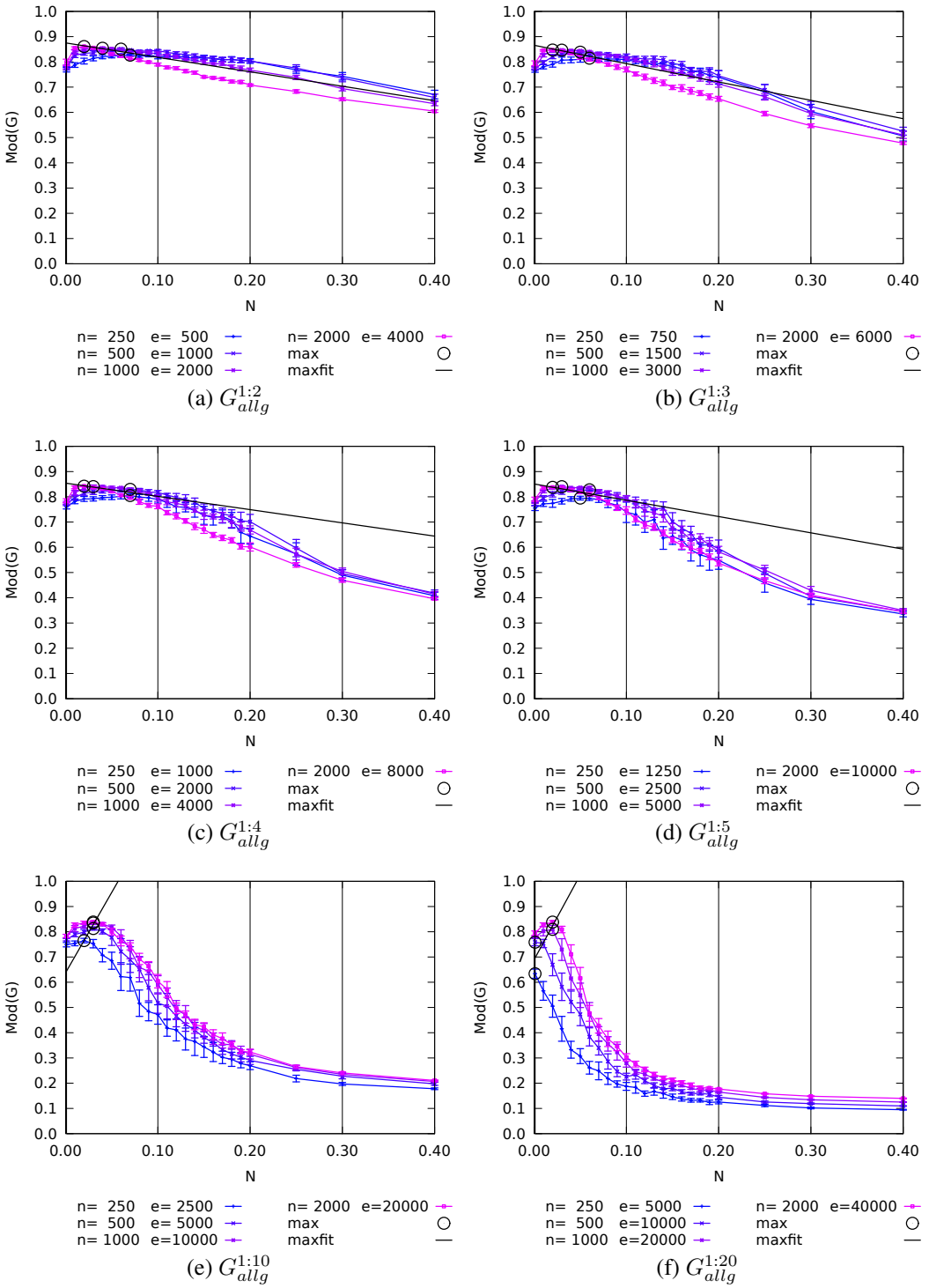
Für $O \to 1$ zeigen die zwei Beispielgraphen aus Abbildung 5.6 die Tendenz einer ansteigenden Modularität, einer anwachsenden Länge des durchschnittlich kürzesten Pfades sowie eines ansteigenden globalen Clustering-Koeffizienten. Zudem existieren für $N \in [0,0;0,2]$ lokale Maxima für alle drei Eigenschaften. Es stellt sich in diesem Zusammenhang die Frage, wie sich die verschiedenen Grapheigen-

schaften unter Berücksichtigung der Knoten-zu-Kanten-Verhältnisse unter variierenden Werten von N verhalten.
Dazu wurden Graphen der Unterklassen $G_{allg}^{n:e} \subset G_{allg}$ mit variierendem Knoten-zu-Kanten-Verhältnis $n : e$ erzeugt. Das Verhalten der Verteilungen wurde für $O \in \{0,80, 0,81, \ldots, 0,99\}$ untersucht. Unabhängig von der Wahl von O zeigen alle generierten Graphen ein ähnliches Verhalten. Da sich für die Wahl von $O = 0,99$ die deutlichsten absoluten Unterschiede ergaben, beschränkt sich die folgende Diskussion auf die für diesen Wert beobachteten Ergebnisse.
Abbildung 5.10 zeigt zunächst die Verteilung der Modularität. Bei der Betrachtung der Abbildungen 5.10(a) – 5.10(f) lassen sich im Verlauf der Verteilungen einige interessante Charakteristika beobachten:

- Mit einem ansteigenden Knoten-zu-Kanten-Verhältnis treten relativ hohe Werte der Modularität in immer kleiner werdenden Intervallen von N auf.
- Das Maximum der Verteilungen verschiebt sich hin zu immer kleiner werdenden Werten von N.
- Die Abnahme der Modularität verstärkt sich für wachsendes N und ein zunehmendes Knoten-zu-Kanten-Verhältnis.

Unabhängig vom Knoten-zu-Kanten-Verhältnis erreichen alle Graphen mit 2.000 Knoten für eine geeignete Wahl von N das gleiche Maximum der Modularität ($\approx$ 0,85). Tendenziell lässt sich also festhalten, dass der durch N und O angestrebte Zwang zur Ausbildung von Gemeinschaften funktioniert.
Betrachtet man ausschließlich Graphen der Klasse $G_{allg}^{1:20}$ (vgl. Abb. 5.10(f)), stellt man fest, dass mit zunehmender Anzahl der Knoten, die Modularität unabhängig von der Wahl von N steigt. Für Ausprägungen der Klasse $G_{allg}^{1:2}$ (vgl. Abb. 5.10(a)) hängt es hingegen stark von der Wahl von N ab, ob Graphen mit geringer Anzahl von Knoten, eine höhere oder eine niedrigere Modularität aufweisen als solche mit einer hohen Anzahl von Knoten.
Insgesamt kann man festhalten, dass sich zum Erreichen einer hohen Modularität die Werte für die Wahl von N bei Graphen mit einem kleinen Knoten-zu-Kanten-Verhältnis und einer unterschiedlichen Anzahl von Knoten stärker unterscheiden müssen, als im Falle von Graphen mit einem großen Knoten-zu-Kanten-Verhältnis und einer ähnlichen Anzahl von Knoten. Die Wahl von N fällt hier nahezu konstant aus.
In Anlehnung an die Modularität veranschaulicht Abbildung 5.11 die Verteilung des globalen Clustering-Koeffizienten für dieselben Graphen. Auf der Basis der Abbildungen 5.11(a) – 5.11(f) lassen sich die gleichen Aussagen über das Verhalten der Verteilungen des globalen Clustering-Koeffizienten treffen, wie im Falle der Modularität:

- Mit einem ansteigenden Knoten-zu-Kanten-Verhältnis treten relativ hohe Werte des globalen Clustering-Koeffizienten in immer kleiner werdenden Intervallen von N auf.

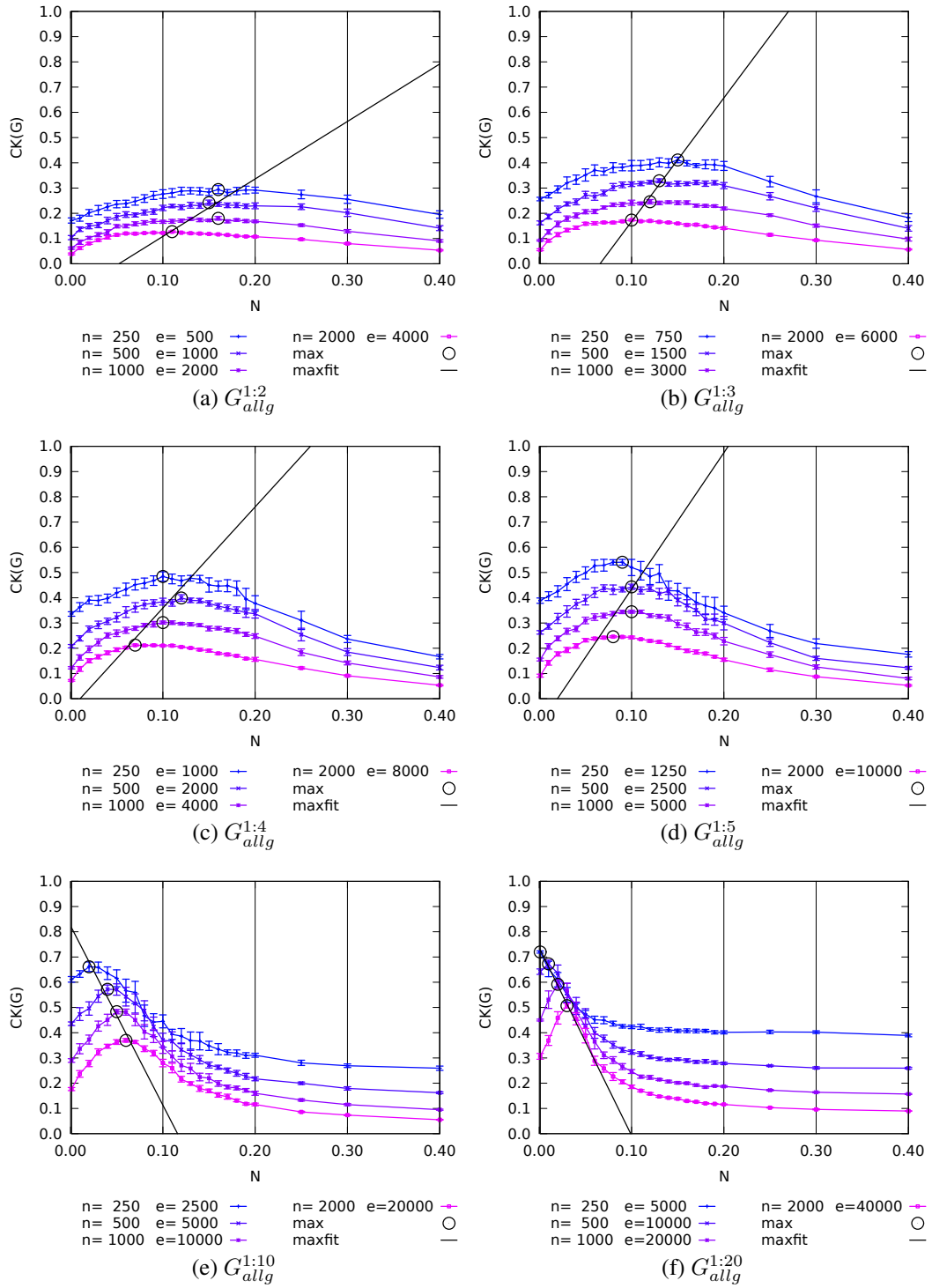


(a) $G_{allg}^{1:2}$ (b) $G_{allg}^{1:3}$

(c) $G_{allg}^{1:4}$ (d) $G_{allg}^{1:5}$

(e) $G_{allg}^{1:10}$ (f) $G_{allg}^{1:20}$

Abbildung 5.10: Verteilung der Modularität für konstante Wahl von $O = 0,99$ in Abhängigkeit von N für Graphen der Unterklassen $G_{allg}^{n:e} \subset G_{allg}$ mit variierendem Knoten-zu-Kanten-Verhältnis $n : e$. Lokale Maxima sind mit einem schwarzen Kreis gekennzeichnet. Die schwarze Linie dient zur Visualisierung des tendenziellen Verlaufs der Verteilung der lokalen Maxima.



(a) $G_{allg}^{1:2}$ (b) $G_{allg}^{1:3}$

(c) $G_{allg}^{1:4}$ (d) $G_{allg}^{1:5}$

(e) $G_{allg}^{1:10}$ (f) $G_{allg}^{1:20}$

Abbildung 5.11: Verteilung des globalen Clustering-Koeffizienten für eine konstante Wahl von $O = 0,99$ in Abhängigkeit von N für Graphen der Unterklassen $G_{allg}^{n:e} \subset G_{allg}$ mit variierendem Knoten-zu-Kanten-Verhältnis $n : e$. Lokale Maxima sind mit einem schwarzen Kreis gekennzeichnet. Die schwarze Linie dient zur Visualisierung des tendenziellen Verlaufs der Verteilung der lokalen Maxima.

- Mit einem ansteigenden Knoten-zu-Kanten-Verhältnis verschiebt sich das (anwachsende) Maximum der Verteilungen hin zu immer kleiner werdenden Werten von N.

- Die Abnahme des globalen Clustering-Koeffizienten verstärkt sich für wachsendes N und ein zunehmendes Knoten-zu-Kanten-Verhältnis.

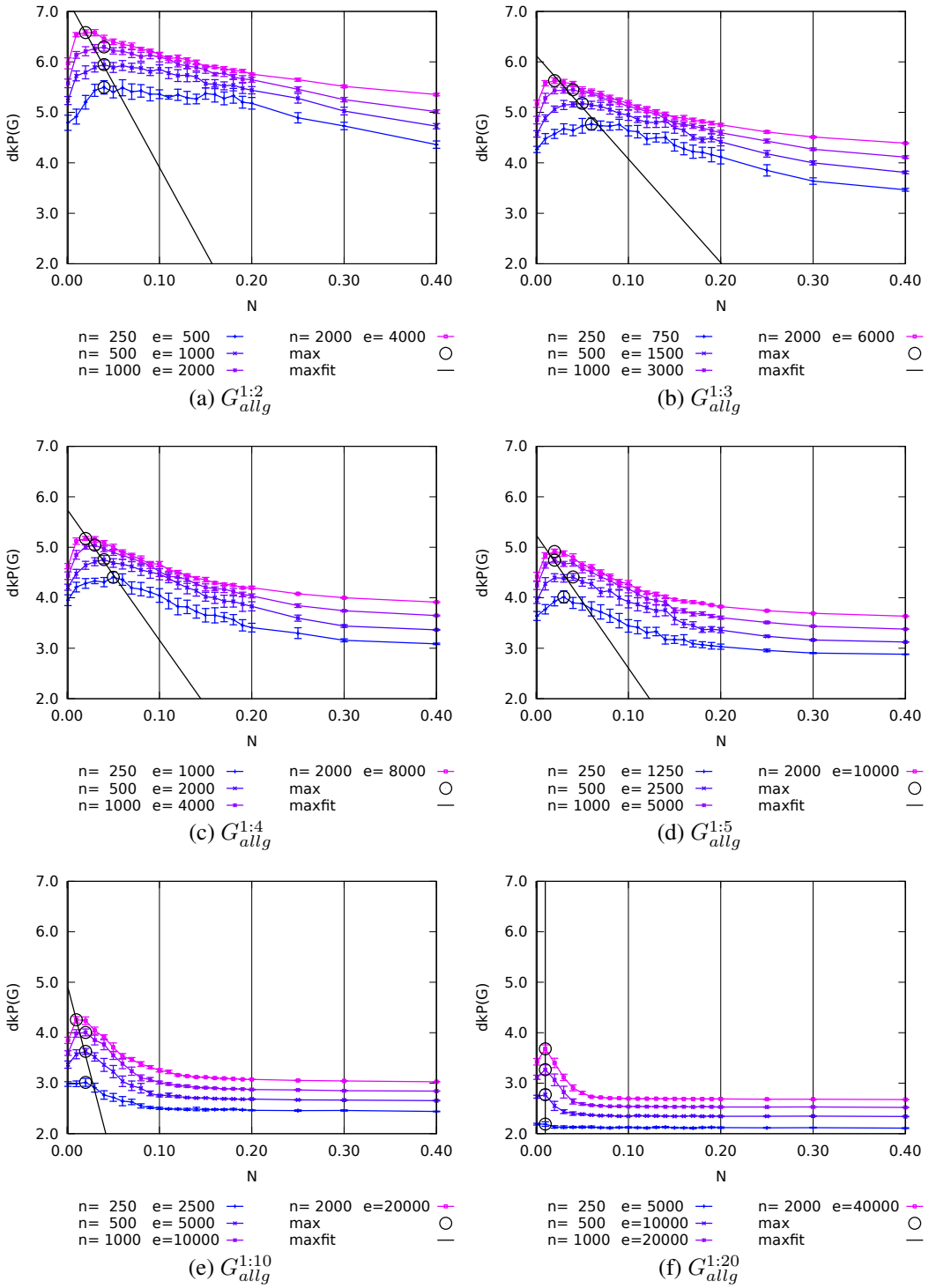
Für alle Graphen der Klasse $G_{allg}^{1:2}$ bis $G_{allg}^{1:10}$ (vgl. Abb. 5.11(a) – 5.11(e)) steigt mit zunehmender Anzahl der Knoten der globale Clustering-Koeffizient unabhängig von der Wahl von N an. Lediglich Graphen der Klasse $G_{allg}^{1:20}$ (vgl. Abb. 5.11(f)) zeigen für eine sehr kleine Wahl von N für Graphen mit einer höheren Anzahl von Knoten den gleichen bzw. einen höheren globalen Clustering-Koeffizienten als solche mit einer geringeren Anzahl von Knoten. Dieses Verhalten ist primär der Verschiebung des Maximums zu immer kleiner werdenden Werten von N bei zunehmendem Knoten-zu-Kanten-Verhältnis geschuldet.
Insgesamt kann man festhalten, dass sich zum Erreichen eines hohen globalen Clustering-Koeffizienten die Wahl von N bei Graphen mit einem kleinen Knoten-zu-Kanten-Verhältnis mit einer anwachsenden Anzahl von Knoten zu größeren Werten verschiebt. Im Falle von Graphen mit großen Knoten-zu-Kanten-Verhältnis hingegen verschiebt sich die Wahl von N mit anwachsender Anzahl von Knoten zu kleineren Werten.
Schließlich veranschaulicht Abbildung 5.12 die Verteilung des durchschnittlich kürzesten Pfades für die zuvor betrachteten Graphen. Unabhängig von der Unterklasse und der Wahl von N steigt mit zunehmender Anzahl von Knoten die Länge des durchschnittlich kürzesten Pfades an.
Als Ergebnis kann man festhalten, dass sich zum Erreichen eines langen durchschnittlich kürzesten Pfades die Wahl von N bei Graphen mit kleinem Knoten-zu-Kanten-Verhältnis mit anwachsender Anzahl von Knoten zu kleineren Werten verschiebt. Je stärker das Knoten-zu-Kanten-Verhältnis wächst, desto stärker gleichen sich die Werte von N einander an.

5.4.1.5 Approximation der Verteilungen der Grapheigenschaften

Während der Evaluation des statischen Modells wurde der Versuch unternommen, die Verteilungen der verschiedenen Grapheigenschaften mit einer Funktion zu approximieren. Mit Hilfe nichtlinearer Regression konnte die Funktion

$$f(x) = \frac{a + bx + cx^2}{1,0 + dx + ex^2} \tag{5.13}$$

als gute Annäherung für die Verteilung des globalen Clustering-Koeffizienten, der Modularität und der durchschnittlich kürzesten Pfade identifiziert werden.
Leider gelang es nicht, das nichtlineare Problem in ein lineares zu überführen, so dass bisher keine Aussage über die systematische Parametrisierung der Koeffizienten a bis e aus Gleichung 5.13 getroffen werden kann. Die Analyse eines um-



(a) $G_{allg}^{1:2}$ (b) $G_{allg}^{1:3}$

(c) $G_{allg}^{1:4}$ (d) $G_{allg}^{1:5}$

(e) $G_{allg}^{1:10}$ (f) $G_{allg}^{1:20}$

Abbildung 5.12: Verteilung des durchschnittlich kürzesten Pfades für konstante Wahl von $O = 0,99$ in Abhängigkeit von N für Graphen der Unterklassen $G_{allg}^{n:e} \subset G_{allg}$ mit variierendem Knoten-zu-Kanten-Verhältnis $n : e$. Lokale Maxima sind mit einem schwarzen Kreis gekennzeichnet. Die schwarze Linie dient zur Visualisierung des tendenziellen Verlaufs der Verteilung der lokalen Maxima.

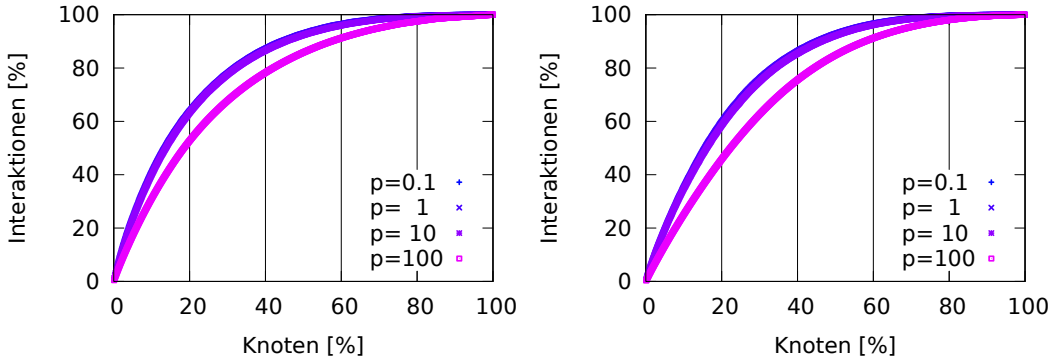
fangreicheren Datensatzes könnte in Zukunft dazu beitragen, einer Lösung dieses Problems näher zu kommen.

5.4.2 Auswirkungen der Parameter auf das interaktive Modell

Wie in Kapitel 5.3.2.3 bereits erläutert, stellt das Befreunden selbst eine Interaktion dar. Damit haben N, O und L auch direkten Einfluss auf die Verteilung der Interaktionen bei der Erzeugung der Graphen. Alle drei Parameter dienen jedoch zur Konfiguration des strukturellen Modells. Sie können nicht explizit dazu verwendet werden, das interaktive Modell zu justieren. Um die Verteilung allgemeiner Interaktionen unabhängig von den strukturellen Parametern zu beeinflussen, werden die Parameter Z und p eingesetzt.
Während Z die Auswahl der nächsten Interaktionskante regelt (vgl. Kap. 5.3.2.2), dient p dazu, den Anteil der allgemeinen Interaktionen dem Verhältnis der Anzahl existierender Freundschaften anzupassen.
Im Folgenden wird der Einfluss beider Parameter auf das Interaktionsverhalten des Modells analysiert.

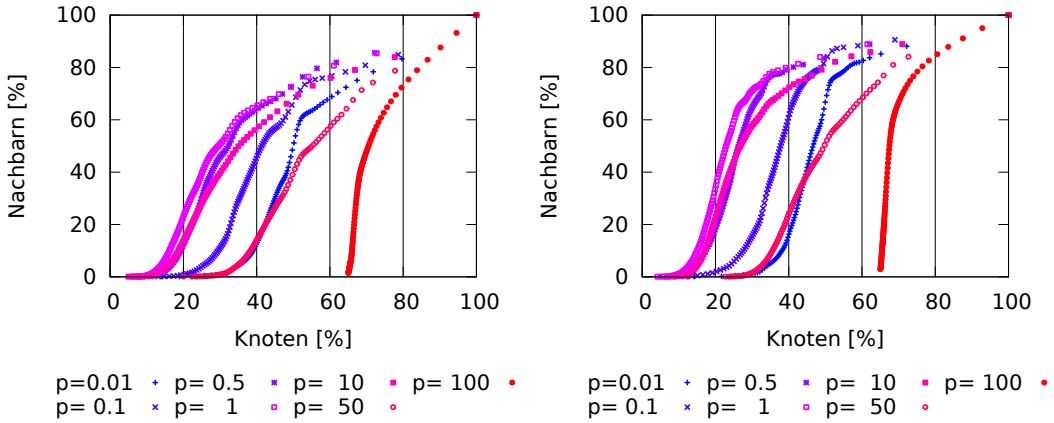
5.4.2.1 Einfluss des Verhältnisses von Befreundungs- und allgemeinen Interaktionen

Der Parameter p dient dazu, das gewünschte Verhältnis zwischen Befreundungs- und allgemeinen Interaktionen herzustellen. p beeinflusst indirekt die Verteilung der Netzwerk- und der Knoteninteraktionen (vgl. Kap. 5.1.4).
Im Folgenden referenziert der Index p_i einen Graphen, für den p den Wert i annimmt. Abbildung 5.13 veranschaulicht für jeweils vier Graphen $G^{p_i}_{1:5}$ der Klasse $G_{1:5}$ und vier Graphen $G^{p_i}_{1:10}$ der Klasse $G_{1:10}$ den Einfluss von p auf die kumulative Häufigkeitsverteilung der Interaktionen. Diese Art der Darstellung wird im Folgenden als *Verteilung der Netzwerkinteraktionen* bezeichnet. Der Parameter Z wurde konstant auf den Wert 10^2 gesetzt.
Die Wahl von $p = 100$ impliziert, dass neben Befreundungsinteraktionen keine weiteren allgemeinen Interaktionen auftreten. Für $G^{p_{100}}_{1:5}$ sind demnach 20% aller Knoten für 52% aller Interaktionen verantwortlich. Für $G^{p_{100}}_{1:10}$ ergibt sich für den gleichen Anteil von Knoten ein Anteil von 45% aller Interaktionen. Der gesteigerte Anteil für $G^{p_{100}}_{1:10}$ resultiert aus der höheren Anzahl von Kanten. Für kleiner werdende Werte von p steigt der Anteil der Interaktionen für eine gleichbleibende Anzahl Knoten an. Ab $p = 10$ verändert sich das Verhältnis nur noch minimal.
p wirkt sich besonders für Werte aus dem Intervall $[10, 100]$ auf die Verteilung der Netzwerkinteraktionen aus.
Abbildung 5.14 veranschaulicht für $G^{p_i}_{1:5}$ und $G^{p_i}_{1:10}$ den Anteil der Nachbarn, mit denen Knoten 70% ihrer Interaktionen tätigen. In Anlehnung an die Analyse von Facebook [217] erfolgt die Darstellung als kumulative Häufigkeitsverteilung der Knoten. Im Folgenden wird sie als *Verteilung der Knoteninteraktionen* bezeichnet.



(a) Netzwerkinteraktionen in $G^{p_i}_{1:5}$ (b) Netzwerkinteraktionen in $G^{p_i}_{1:10}$

Abbildung 5.13: Auswirkung von p auf die Verteilung der Netzwerkinteraktionen. Die Wahl $p = 100$ entspricht dem ausschließlichen Auftreten von Befreundungsinteraktionen.



(a) Knoteninteraktionen in $G^{p_i}_{1:5}$ (b) Knoteninteraktionen in $G^{p_i}_{1:10}$

Abbildung 5.14: Verteilungen der Knoteninteraktionen für $G^{p_i}_{1:5}$ und $G^{p_i}_{1:10}$.

Der Kurvenverlauf von $G^{p_{10}}_{1:5}$ ist wie folgt zu interpretieren: 70% aller Interaktionen beschränken sich bei 40% aller Nutzer auf 45% ihrer Freunde. Betrachtet man hingegen 60% aller Nutzer, sind es 77% ihrer Freunde mit denen sie 70% ihrer Interaktionen tätigen.

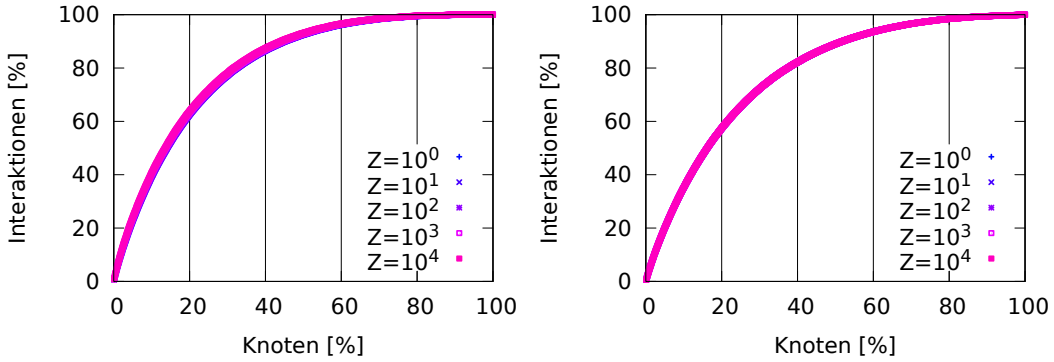
Ausgehend von $p = 100$ verschiebt sich der Verlauf der Verteilung zunächst immer weiter nach links. Der Anteil von Nutzern, die mit einem konstanten Anteil der Freunde 70% ihrer Interaktionen tätigen, nimmt kontinuierlich ab.

Diese Beobachtung lässt sich wie folgt erklären: Die wachsende Anzahl allgemeiner Interaktionen verteilt sich entsprechend der Gleichungen 5.10 und 5.11 über die Kanten des Graphen. Gleichung 5.10 bewirkt, dass hochgradigen Knoten tendenziell mehr Interaktionen zugewiesen werden als solchen niedrigen Grades. Die Verteilung auf Knotenebene hängt von der Parametrisierung der Gleichung 5.11 ab. Für den hier gewählten Wert $Z = 10^2$ ergibt sich eine kumulative Exponential-

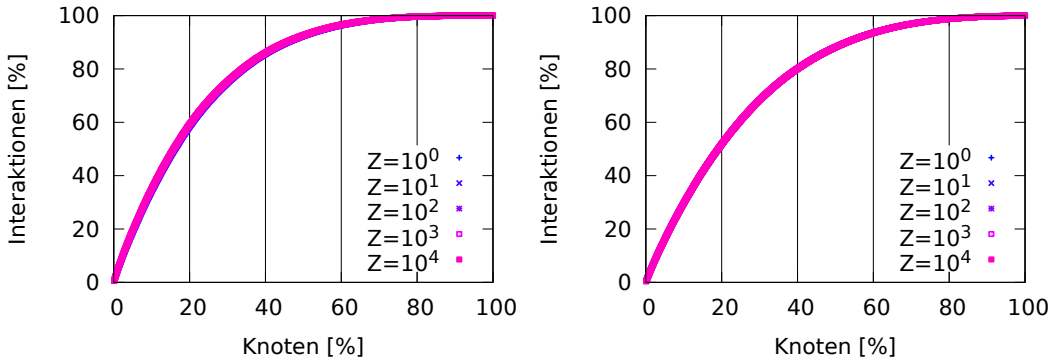
verteilung (vgl. Kap. 5.3.2.2). Dadurch, dass weitere Interaktionen bevorzugt den Kanten zwischen Knoten hohen Grades und deren aktivsten Freunden zugewiesen werden, wächst auch der Anteil derjenigen Knoten am stärksten an, welche nur mit wenigen ihrer Freunde nahezu alle Interaktionen durchführen. Dies wiederum bewirkt die Verschiebung der Verteilung nach links.
Erstaunlicherweise kommt es ab $p = 1$ wieder zu einer Verschiebung der Verteilung nach rechts.
Diese Beobachtung lässt sich wie folgt erklären: Abhängig von der Wahl von Z nimmt der Anteil der Interaktionen auf weniger aktiven Kanten mehr oder weniger schnell zu. Für große Z dauert dieser Vorgang länger, da die Anzahl bereits getätigter Interaktionen starken Einfluss auf die Auswahl einer Kante hat (vgl. Kap. 5.3.2.2 bzw. Abb. 5.2).
Insgesamt wächst die Anzahl der Interaktionen für alle, also auch weniger aktive, Kanten mit kleiner werdendem p an. Ab einem bestimmten Zeitpunkt werden auch auf weniger aktiven Kanten so viele Interaktionen getätigt, dass der Einfluss von Gleichung 5.11 bei der Auswahl einer Kante für die Zuweisung weiterer Interaktionen gänzlich verloren geht.
Um für einen Knoten weiterhin den Anteil seiner Freunde zu bestimmten, mit dem er 70% seiner Interaktionen tätigt, müssen immer mehr Freunde berücksichtigt werden. Die Verteilung verschiebt sich insgesamt wieder nach rechts.
Da die Auswahl der Knoten, die eine Interaktion durchführen, stets entsprechend der Verteilung der Knotengrade erfolgt, verläuft die Steigung der Kurve ab dem Umkehrpunkt jedoch nahezu konstant.
Über den Parameter p lässt sich also steuern, wie hoch der Anteil der Nutzer ausfallen soll, die mit einem hohen Anteil ihrer Freunde interagieren.

5.4.2.2 Einfluss der Aktivität einer Beziehung auf die Auswahl von Interaktionskanten

Nachdem ein Knoten u für die Initialisierung einer Interaktion bestimmt wurde (vgl. Kap. 5.3.2.1), muss eine Kante $e(u, v)$ zu einem der Nachbarn $v \in \Gamma(u)$ selektiert werden, auf der die neue Interaktion ausgeführt werden soll. Z beeinflusst indirekt die Verteilung der Netzwerk- und direkt die Verteilung der Knoteninteraktionen (vgl. Kap. 5.1.5).
Bei der Auswahl einer Kante $e(u, v)$ gewichtet Z den Einfluss bereits getätigter Interaktionen $ia(u, v)$. Da lediglich p die Auswahl von u bestimmt, ist zu erwarten, dass Z kaum Auswirkungen auf die Verteilung der Netzwerkinteraktionen hat. Da Interaktionen jedoch als ungerichtet betrachtet werden, könnte die Auswahl von v die Verteilung der Interaktionen doch teilweise beeinflussen.
Im Folgenden referenziert der Index Z_i einen Graphen, für den Z den Wert i annimmt. Abbildung 5.15 zeigt für $G^{Z_i}_{1:5}$ und $G^{Z_i}_{1:10}$ jeweils den Einfluss von Z auf die Verteilung der Netzwerkinteraktionen. Die Verteilung wurde jeweils für $p = 0,05$ und $p = 50$ analysiert. Die Ergebnisse zeigen deutlich, dass Z keinerlei Einfluss auf die Verteilung der Interaktionen hat. Alle Kurven sind nahezu deckungsgleich.



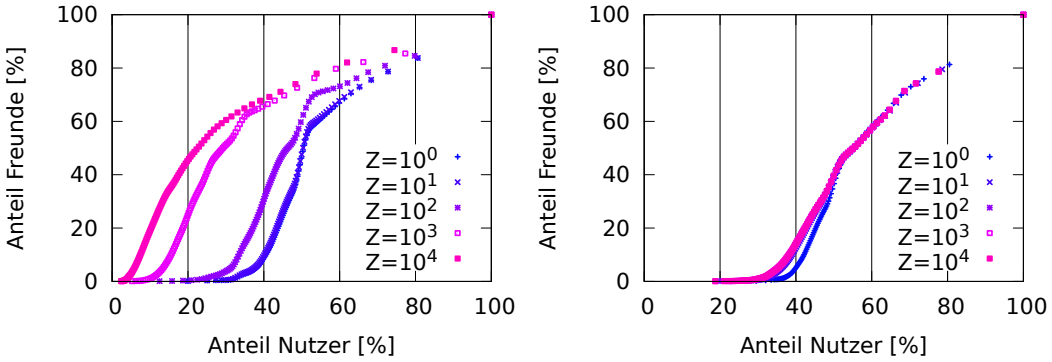
(a) Netzwerkinteraktionen in $G_{1:5}^{Z_i}$ $(p = 0,05)$ (b) Netzwerkinteraktionen in $G_{1:5}^{Z_i}$ $(p = 50)$



(c) Netzwerkinteraktionen in $G_{1:10}^{Z_i}$ $(p = 0,05)$ (d) Netzwerkinteraktionen in $G_{1:10}^{Z_i}$ $(p = 50)$

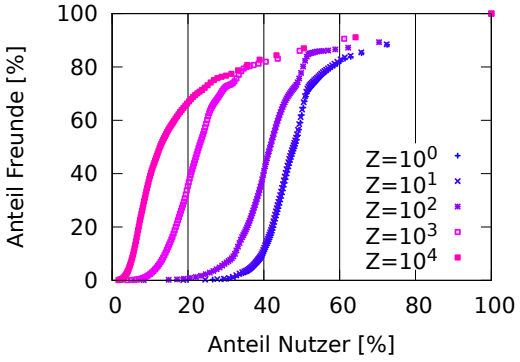
Abbildung 5.15: Auswirkung von Z auf die Verteilung der Netzwerkinteraktionen.

Abbildung 5.16 veranschaulicht die Verteilungen der Knoteninteraktionen für $G_{1:5}^{Z_i}$ und $G_{1:10}^{Z_i}$. Die Verteilungen wurden jeweils für $p = 0,05$ und $p = 50$ analysiert. In Anlehnung an Abbildung 5.14 sind wieder die Knoteninteraktionen für unterschiedliche Werte von Z dargestellt. Für die Wahl von $p = 0,05$ (vgl. Abb. 5.16(a) und 5.16(c)) lässt sich ein sehr starker Einfluss von Z auf die Verteilung beobachten. Während in $G_{1:5}^{Z_{10^0}}$ 70% aller Interaktionen von 49% aller Nutzer mit bis zu 45% ihrer Freunde tätigen, sind es bei $G_{1:5}^{Z_{10^4}}$ lediglich nur noch maximal 20% aller Nutzer, die mit dem gleichen Anteil von Freunden 70% ihrer Interaktionen ausführen.
Das Verhalten lässt sich wie folgt erklären: Die Wahrscheinlichkeit $P(e(u,v))$ für die Auswahl einer Interaktionskante $e(u,v)$ berechnet sich entsprechend Gleichung 5.11. Diese beschreibt die kumulative Verteilungsfunktion einer Exponentialverteilung (vgl. Kap. 5.3.2.2 bzw. Abb. 5.2). Mit $Z \to \infty$ hängt die Auswahl einer Kante immer weniger von der Anzahl der darauf getätigten Interaktionen ab. Es gilt daher $\lim_{Z\to\infty} P(e(u,v)) = \frac{1}{|\Gamma(u)|}$. Ein sehr kleiner Wert von Z führt dazu, dass die meisten Knoten nur mit einem sehr geringen Anteil ihrer Freunde die Mehrzahl ihrer Interaktionen durchführen. In Abbildung 5.16(a) lässt sich dieses Verhalten z.B. daran erkennen, dass für $Z = 10^0$ 40% aller Knoten mit maximal 9% ihrer Freunde 70% ihrer Interaktionen tätigen. Ein großer Wert von Z hingegen bewirkt, dass die meisten Knoten mit allen ihren Freunden ungefähr den gleichen Anteil von Interaktionen

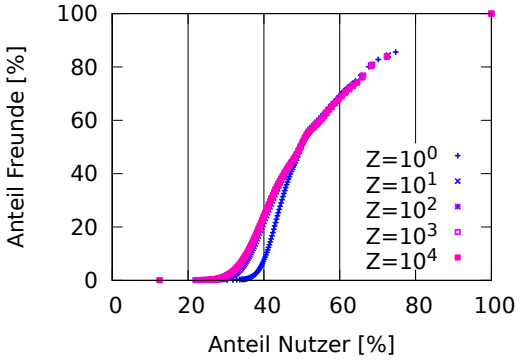


(a) Knoteninteraktionen in $G^{Z_i}_{1:5}$ ($p = 0,05$)

(b) Knoteninteraktionen in $G^{Z_i}_{1:5}$ ($p = 50$)



(c) Knoteninteraktionen in $G^{Z_i}_{1:10}$ ($p = 0,05$)



(d) Knoteninteraktionen in $G^{Z_i}_{1:10}$ ($p = 50$)

Abbildung 5.16: Verteilungen der Knoteninteraktionen für $G^{Z_i}_{1:5}$ und $G^{Z_i}_{1:10}$.

durchführen. Für $Z = 10^4$ tätigen 40% aller Knoten 70% ihrer Interaktionen mit bis zu 68% ihrer Freunde. Für $G^{Z_i}_{1:10}$ ist prinzipiell das gleiche Verhalten beobachtbar (vgl. Abb. 5.16(c)).

Die Abbildungen 5.16(b) und 5.16(d) zeigen die Verteilungen für $G^{Z_i}_{1:5}$ und $G^{Z_i}_{1:10}$ bei einer Wahl von $p = 50$. Es ist deutlich zu erkennen, dass Z nur noch einen marginalen Einfluss auf die Verteilung hat. Der Verlauf lässt sich damit erklären, dass für $p = 50$ lediglich 50% der getätigten Interaktionen allgemeine Interaktionen darstellen. Nur auf ca. 50% der Kanten kommt neben den Befreundungsinteraktionen eine weitere allgemeine Interaktion hinzu. p dominiert daher den Verlauf der Verteilungen.

In Analogie zum Parameter p lässt sich mit Z steuern, wie hoch der Anteil der Nutzer ausfallen soll, der mit einem hohen Anteil seiner Freunde interagiert. Anders als p hat Z jedoch keinen Einfluss auf die Verteilung der Netzwerkinteraktionen. Insbesondere wenn der Anteil allgemeiner Interaktionen gegenüber dem Anteil von Befreundungsinteraktionen stark überwiegt, lässt sich mit Z der Grad der Interaktionen zwischen Nutzern und ihren Freunden sehr flexibel erhöhen.

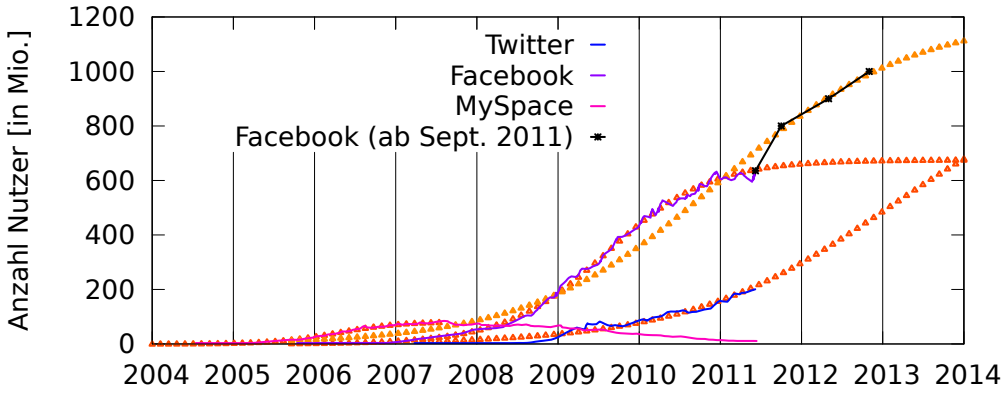


Abbildung 5.17: Ermitteltes Wachstum von Facebook, Twitter und MySpace (aufbereitet auf der Basis von [258]). Die gepunkteten Linien repräsentieren jeweils die Wachstumsprognose auf der Basis einer logistischen Approximation.

5.4.3 Evaluation des Modells auf der Basis realer Datensätze

Die beiden vorhergehenden Kapitel lieferten eine Beschreibung der Auswirkungen unterschiedlicher Konfigurationsparameter auf die strukturellen und interaktiven Eigenschaften modellierter Graphen. Im Folgenden wird die konkrete Anwendbarkeit des Modells zur Erzeugung sozialer Interaktionsgraphen evaluiert.
Zunächst wird untersucht, inwiefern sich das Netzwerkwachstum auf der Basis einer logistischen Funktion (vgl. Kap. 5.3.1.1) für die Modellierung von Netzwerkeffekten eignet. Anschließend werden das strukturelle und das interaktive Modell auf ihre Eignung zur Abbildung der Grapheigenschaften sozialer Netzwerke am Beispiel von Enron [239] und Facebook diskutiert.

5.4.3.1 Netzwerkeffekt

Wegen des hohen organisatorischen Aufwands bei der Durchführung sozialer Experimente sind frei zugängliche Datensätze von RSNs stets auf sehr kleine Populationen beschränkt [70, 227, 24]. Daher erfolgt die Beurteilung des Modells auf seine Eignung zur Simulation von Netzwerkeffekten ausschließlich am Beispiel ausgewählter OSNs.
Abbildung 5.17 veranschaulicht noch einmal die Entwicklung der Mitgliederzahlen von Facebook, Twitter und MySpace seit der Gründung des jeweiligen OSNs bis zum 30. August 2011 (vgl. Abb. 5.1). Die orangefarben gepunkteten Kurven stellen eine Näherung der Wachstumsphase auf der Basis einer logistischen Funktion entsprechend Gleichung 5.1 dar. Tabelle 5.4 zeigt die durch nichtlineare Regression [131] ermittelten Funktionsparameter für die drei OSNs.
Die rein visuelle Auswertung lässt darauf schließen, dass sich eine logistische Funktion als Approximation für das tatsächliche Wachstum der drei Netzwerke eignet.
Für Facebook prognostiziert das Modell eine Sättigung bei ca. 650 Millionen Mitgliedern im Februar 2012. Das Wachstum hat seitdem jedoch nicht abgenommen.

OSN	G	a	k
MySpace	82,35	2005,81	2,60
Twitter	999,99	2012,62	0,794
Facebook	654,26	2009,05	1,659

Tabelle 5.4: Parametrisierung der logistischen Funktion für die OSNs Facebook, Twitter und MySpace.

Facebook überschritt bereits im Oktober 2012 die Milliarden Marke [251] und erreichte Ende März 2013 eine Mitgliederzahl von 1,11 Milliarden Nutzern [247].
Als Ergänzung visualisiert die schwarze Linie im Diagramm die seitdem publizierten Zuwachsraten. Die gelb gepunktete Kurve stellt eine mögliche logistische Approximation unter Berücksichtigung dieser Daten dar. Folgt das Wachstum tatsächlich dieser Funktion, so erreicht Facebook sein Maximum mit ca. 1,2 Milliarden Nutzern Ende 2016. Bisher stimmt diese Prognose mit den heutigen Beobachtungen überein.
Glaubt man dem Modell, befindet sich Twitter derzeit am Beginn seiner stärksten Wachstumsphase. Das Maximum von einer Milliarde Nutzern wird für das Jahr 2020 prognostiziert.

5.4.3.2 Enron

Im Rahmen der Bilanzfälschungsaffäre um den Energiekonzerns Enron [145, 37] wurde im Jahr 2003 von der Energieregulierungsbehörde der USA ein E-Mail-Datensatz des Unternehmens publiziert. Der Datensatz besteht aus den E-Mail-Postfächern von 158 Mitgliedern der Managementebene des Konzerns. Die Postfächer umfassen alle E-Mails, die im Zeitraum von 1998 bis 2002 verfasst bzw. versendet wurden [115].
Da keine hinreichend umfangreichen Interaktionsgraphen von RSNs existieren (vgl. Kap. 5.4.3.1), dient der Datensatz als Hilfsmittel zur Beurteilung des Modells im Hinblick auf seine Anwendbarkeit zur Erzeugung von Graphen mit den Eigenschaften von RSNs. Die Ähnlichkeit des Kommunikationsverhaltens innerhalb eines E-Mail-Netzwerks mit dem eines RSNs wird damit begründet, dass hinter jeder Interaktion (E-Mail) ein tatsächlicher Anreiz zur Kommunikation besteht. In diesem Punkt unterscheidet sich das Interaktionsaufkommen des Enron-Netzwerks stark von demjenigen eines gewöhnlichen OSNs. Beispielsweise wird in einem OSN schon das Markieren von Inhalten als Interaktion interpretiert.
Für die Auswertung wird auf eine aufbereitete Version [272] des ursprünglichen Datensatzes zurückgegriffen. Um die Existenz einer gewissen sozialen Bindung der kommunizierende Mitarbeiter sicherzustellen, wurde der Datensatz abermals reduziert und in einen sozialen Interaktionsgraphen G_E konvertiert [295].
In G_E existiert eine Kante zwischen Knoten nur noch dann, wenn sich die entsprechenden Mitarbeiter gegenseitig jeweils mindestens drei Nachrichten zugeschickt haben. Der soziale Graph reduziert sich damit auf 129 der ursprünglich 158 Knoten.

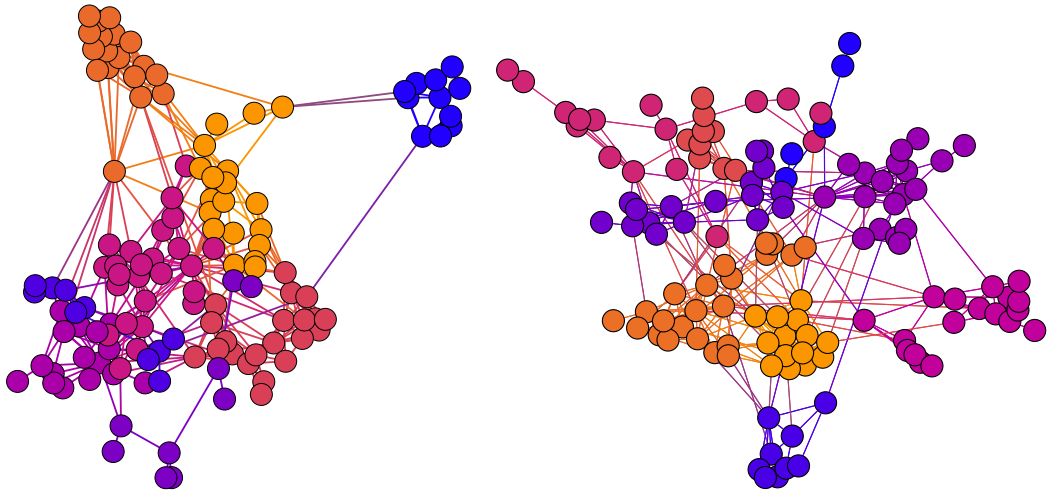
$Parameter$	t	n	e	N	O	L	E	Z	p
$Wert$	10	129	402	0,09	0,98	8	0	50	0,01

Tabelle 5.5: Konfiguration der Parameter zur Erzeugung von G_E.

$Parameter$	G_E	$\emptyset(G_E^{Syn})$	$KI(G_E^{Syn})$	$G_E^{Syn^*}$
CK	0,429	0,429	0,010	0,426
dkP	3,420	3,675	0,052	3,394
Rad	4,000	4,900	0,172	4,000
Dia	8,000	8,900	0,317	7,000
Mod	0,620	0,717	0,009	0,666

Tabelle 5.6: Auflistung der Grapheigenschaften für den Enron-Datensatz G_E und für den Durchschnitt $\emptyset(G_E^{Syn})$ inklusive der Konfidenzintervalle $KI(G_E^{Syn})$ der künstlich erzeugten Graphen G_E^{Syn}. $G_E^{Syn^*}$ repräsentiert den künstlich erzeugten Graphen mit der besten Übereinstimmung in den Grapheigenschaften von G_E.

Diese sind über 402 Kanten miteinander assoziiert. Von den ursprünglich 619.446 verfassten bzw. ausgetauschten E-Mails werden nur noch die verbleibenden 37.129 als Interaktionen interpretiert.
Dieses Vorgehen stellt nicht zwangsläufig die Existenz einer echten Beziehung zwischen den Mitarbeitern sicher. Ein unverzerrtes Abbild der Struktur des sozialen Graphen ist somit nicht garantiert. Es werden durch den Filtervorgang jedoch alle Interaktionen eliminiert, die sicher keine echte Kommunikationsbeziehung repräsentieren. Dazu zählen beispielsweise E-Mails, die auf der Basis einer „Allen Antworten“-Funktion generiert wurden.
Tabelle 5.5 veranschaulicht die Parametrisierung für die Erzeugung der modellierten Graphen G_E^{Syn}. Mit t wird die Anzahl der Wachstumsphasen des Graphen auf 10 festgelegt. Für die Auswertung spielt t an dieser Stelle keine weitere Rolle. Die Parameter n und e ergeben sich aus der Anforderung zur Modellierung eines Graphen mit 129 Knoten und 402 Kanten. Das Knoten-zu-Kanten-Verhältnis liegt somit bei 1:3,116.
Tabelle 5.6 beinhaltet in der Spalte G_E die Auflistung der strukturellen Grapheigenschaften für den Enron-Datensatz. Der globale Clustering-Koeffizient liegt mit 0,429 extrem hoch. Die Modularität von 0,620 entspricht einem eher moderaten Wert.
Abbildung 5.6 zeigte bereits den Einfluss von N und O (bei $L = 10$ konstant) auf den globalen Clustering-Koeffizienten und die Modularität von $G_{1:5}$. $G_{1:5}$ hat ein größeres Knoten-zu-Kanten-Verhältnis (1:5) als G_E (1:3,116) und dient daher nur zur Orientierung für die Auswahl passender Parameter für die Modellierung von G_E. Offensichtlich erreicht man für $G_{1:5}$ einen maximalen globalen Clustering-Koeffizienten von ca. 0,35 für $N = 0,10$ und $O = 0,99$. Dieser ist deutlich kleiner als der von G_E (0,429). Die entsprechende Modularität liegt für $N = 0,10$ und



(a) G_E (b) $G_E^{Syn^*}$

Abbildung 5.18: Visualisierung von G_E und $G_E^{Syn^*}$. Die Einfärbung der Knoten repräsentiert die Zugehörigkeit der über die Modularität ermittelten Gemeinschaften.

$O = 0,99$ bei ca. 0,8 und somit deutlich höher als die von G_E. Um den globalen Clustering-Koeffizienten zu erhöhen und gleichzeitig die Modularität zu vermindern, muss man N erhöhen (vgl. Kap. 5.4.1.2) und L erniedrigen (vgl. Kap. 5.4.1.3). Für G_E ergeben sich $N = 0,09$, $O = 0,98$ und $L = 8$ als die am besten passenden Werte.

Tabelle 5.6 veranschaulicht den Durchschnitt $\text{ø}(G_E^{Syn})$ der Grapheigenschaften von 30 erzeugten Graphen, die zugehörigen Konfidenzintervalle $KI(G_E^{Syn})$ und die Kennzahlen des Graphen $G_E^{Syn^*}$, dessen Grapheigenschaften die größte Übereinstimmung mit denen von G_E besitzen.

Offensichtlich erzeugt das Modell Graphen G_E^{Syn}, deren Eigenschaften denen von G_E sehr nahe kommen. Lediglich die Modularität liegt im Durchschnitt mit einer Differenz von ca. $0,1$ im Falle von G_E^{Syn} etwas höher.

In Abbildung 5.18 sind die Graphen von G_E und $G_E^{Syn^*}$ dargestellt. Die Einfärbung der Knoten repräsentiert die Zugehörigkeit der über die Modularität ermittelten Gemeinschaften.

Abbildung 5.19(a) zeigt für G_E und $\text{ø}(G_E^{Syn^*})$ die Verteilung der Knotengrade. Offensichtlich liefert das Modell eine sehr hohe Übereinstimmung mit der Verteilung von G_E. Die Verteilung lokaler Clustering-Koeffizienten von $G_E^{Syn^*}$ kommt der von G_E ebenfalls sehr nahe (vgl. Abb. 5.19(b)).

Abbildung 5.19(c) veranschaulicht schließlich die Verteilung lokaler Clustering-Koeffizienten als Funktion der Knotengrade für G_E, $\text{ø}(G_E^{Syn})$ und $G_E^{Syn^*}$. Im Mittel zeigt sich für die meisten Knoten in G_E und $\text{ø}(G_E^{Syn})$ für ähnliche Grade auch eine ähnliche Zuordnung der Clustering-Koeffizienten. Insgesamt liefert das Modell eine sehr gute Approximation des Kontaktgraphen des Enron-Datensatz.

Die Anzahl aller getätigten Interaktionen innerhalb von G_E liegt bei 37.129. Damit ergibt sich für p der Wert $402/37.129 = 0,01$. Z wird auf den Wert 50 festgelegt, so

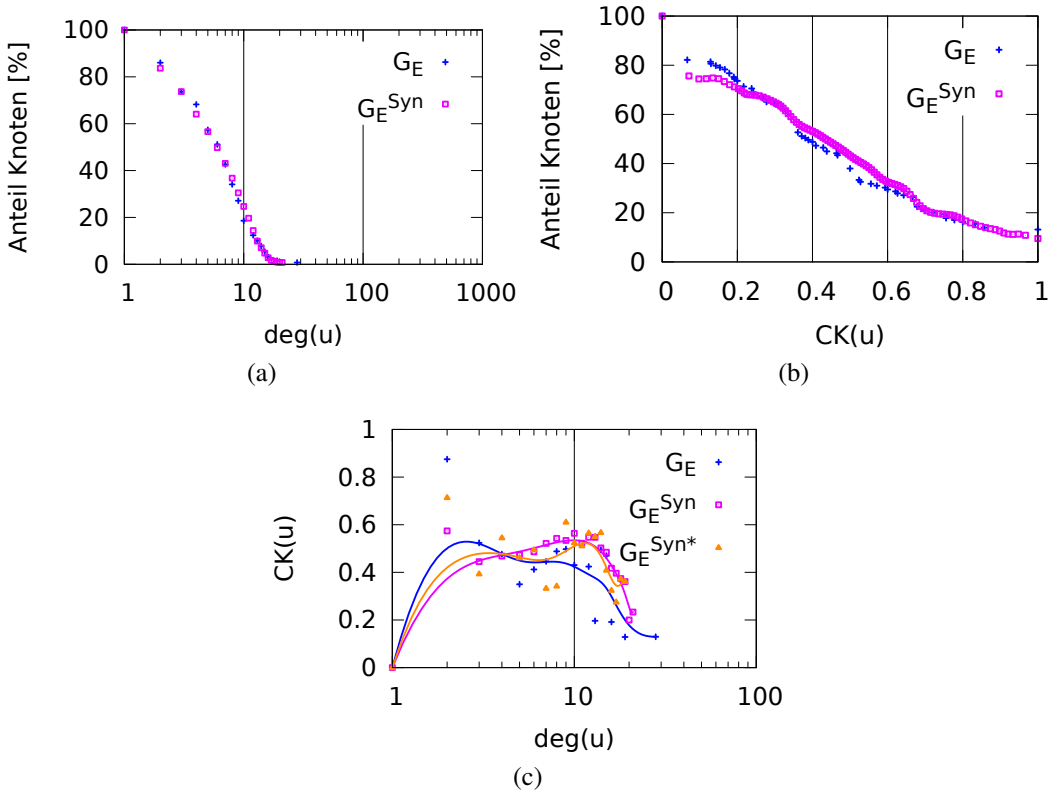


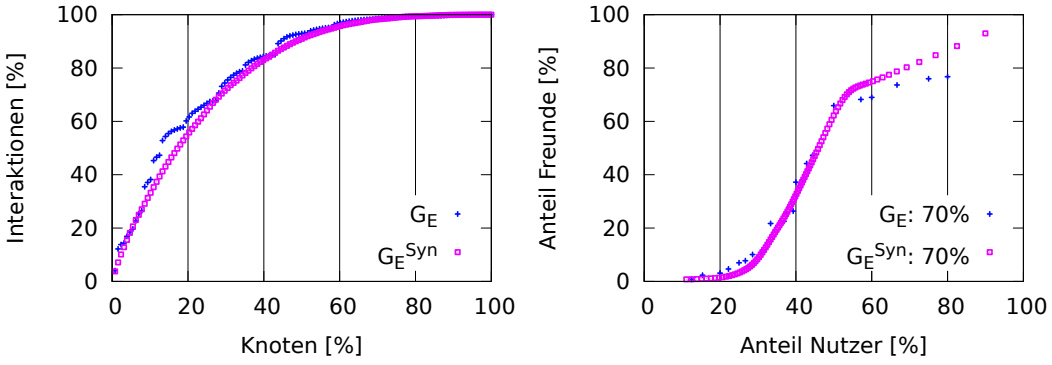
Abbildung 5.19: (a) Komplementär kumulative Häufigkeitsverteilungen der Knotengrade, (b) der lokalen Clustering-Koeffizienten und (c) Verteilung lokaler Clustering-Koeffizienten als Funktion der Knotengrade für G_E und $G_E^{Syn^*}$.

dass das Auftreten von Interaktionen etwas stärker von der Anzahl bereits getätigter Interaktionen abhängt (vgl. Kap. 5.4.2.2).

Abbildung 5.20(a) zeigt für G_E und $\text{ø}(G_E^{Syn})$ die Verteilung der Netzwerkinteraktionen. Insgesamt liefert das Modell hier ebenfalls ein sehr zufriedenstellendes Ergebnis. Für 0 bis 8% sowie 30 bis 100% der Knoten deckt sich die Verteilung von $\text{ø}(G_E^{Syn})$ nahezu mit der von G_E. Lediglich im dazwischen liegenden Bereich zeigt das Modell im Durchschnitt eine leichte Abweichung vom Enron-Datensatz.

Die Verteilungen der Knoteninteraktionen sind in Abbildung 5.20(b) visualisiert. Für einen Anteil von Nutzern zwischen 0 und 50% erzielt das Modell eine sehr gute Übereinstimmung mit G_E. Für einen Anteil von mehr als 50% liegt die Verteilung von $\text{ø}(G_E^{Syn})$ um 5 bis 10% höher als die von G_E.

Insgesamt kann man festhalten, dass das Modell sich sehr gut eignet, sowohl die strukturellen als auch die interaktiven Eigenschaften des Enron-Netzwerks abzubilden.



(a) Netzwerkinteraktionen (b) Knoteninteraktionen

Abbildung 5.20: (a) Verteilung der Netzwerkinteraktionen und (b) der Knoteninteraktionen für G_E und $\emptyset(G_E^{Syn})$.

5.4.3.3 Facebook

Da derzeit keine Crawling-Datensätze dezentraler OSNs existieren (vgl. Kap. 5.2.2), lässt sich keine Aussage darüber treffen, welche strukturellen und interaktiven Gemeinsamkeiten Systeme wie Vegas und gewöhnliche OSNs besitzen. Daher wird das Modell auch auf seine Eignung zur künstlichen Erzeugung gewöhnlicher OSNs hin untersucht.
Im Rahmen dieser Arbeit wird auf einen aufbereiteten Datensatz des OSNs Facebook [217] zurückgegriffen. Der Datensatz zeichnet sich dadurch aus, dass er neben den rein strukturellen Eigenschaften des Kontaktgraphen auch Informationen über das Interaktionsverhalten der Mitglieder enthält.
Um auf einfache Art und Weise interessante Personen in der näheren Umgebung zu finden und sich mit diesen zu verbinden, konnte man sich bei Facebook bis ins Jahr 2009 [266] sogenannten *regionalen Netzwerken* (engl: *regional networks*) anschließen. Der oben genannte Datensatz setzt sich aus 22 solcher regionalen Netzwerke zusammen. Diese wurden von März bis einschließlich Mai 2008 gecrawlt. Das regionale Netzwerk von San Francisco wurde zudem täglich komplett gecrawlt. Die strukturellen Grapheigenschaften des Facebook-Datensatzes beziehen sich daher auf den gesamten Datenbestand, wohingegen sich die Erkenntnisse über das Interaktionsverhalten fast ausschließlich auf das regionale Netzwerk von San Francisco beschränken.
Insgesamt wurden ca. 10 Millionen Nutzer und 408 Millionen Verbindungen identifiziert. Das Knoten-zu-Kanten-Verhältnis liegt somit bei ca. 1:40. Tabelle 5.7 zeigt in der Spalte G_{FB} alle weiteren strukturellen Grapheigenschaften für den Durchschnitt von 10 der 22 regionalen Netzwerke.
Mit 0,164 liegt der globale Clustering-Koeffizient von G_{FB} deutlich unter dem von G_E (vgl. Kap. 5.4.3.2). Leider liefert der Facebook-Datensatz keine Informationen über die Modularität der gecrawlten Graphen. Die Vermutung liegt jedoch nahe, dass die Modularität von G_{FB} im Vergleich zu G_E ebenfalls einen relativ kleinen Wert aufweist. Die 10 Datensätze liefern für den durchschnittlich kürzesten Pfad als

$Parameter$	G_{FB}	$\emptyset(G_{FB}^{Syn})$	$KI(G_{FB}^{Syn})$	$G_{FB}^{Syn^*}$
CK	0,164	0,068	0,001	0,069
dkP	4,800	2,563	0,001	2,562
Rad	9,800	3,067	0,143	4,000
Dia	13,400	5,600	0,281	6,000
Mod	?	0,103	0,001	0,106

Tabelle 5.7: Auflistung der Grapheigenschaften für den Facebook-Datensatz G_{FB} und für den Durchschnitt $\emptyset(G_{FB}^{Syn})$ inklusive der Konfidenzintervalle $KI(G_{FB}^{Syn})$ der künstlich erzeugten Graphen G_{FB}^{Syn}. $G_{FB}^{Syn^*}$ repräsentiert den künstlich erzeugten Graphen mit der besten Übereinstimmung in den Grapheigenschaften von G_{FB}.

$Parameter$	t	n	e	N	O	L	E	Z	p
$Wert$	1.000	5.000	200.000	0,00	1,00	10	0	10.000	0,4

Tabelle 5.8: Konfiguration der Parameter zur Erzeugung von G_{FB}.

Mittelwert eine Länge von 4,80. Insgesamt stellt Facebook ein skalenfreies Kleine-Welt-Netzwerk dar, bei dem die Verteilung der Knotengrade einem Potenzgesetz mit dem Koeffizienten 1,5 folgt [217].
Tabelle 5.8 veranschaulicht die verwendete Parametrisierung für die Erzeugung der Graphen G_{FB}^{Syn}. Bedingt durch den enormen Umfang der gecrawlten regionalen Netzwerke wurde die Größe auf 5.000 Knoten und 200.000 Kanten beschränkt. Als Parametrisierung für die Modellierung wurden die Werte $N = 0,00$, $O = 1,00$ und $L = 10$ gewählt. Es wird folglich kein Zwang zur Bildung gesonderter Gemeinschaften ausgeübt.
Entsprechend der beobachteten Skaleninvarianz von G_{FB} wird die Struktur der erzeugten Graphen primär durch das Preferential-Attachment determiniert. Lediglich L erhöht die Wahrscheinlichkeit zur verstärkten Bildung von Kanten zwischen Knoten mit einem hohen Anteil gemeinsamer Nachbarn. Für den Facebook-Datensatz wurden auf den insgesamt 940 Millionen sozialen Verbindungen ca. 24 Millionen Interaktionen gezählt. Bezogen auf die 10 ausgewählten regionalen Netzwerke ergeben sich im Durchschnitt ca. eine Millionen Interaktionen. Bei 400.000 Nutzern des regionalen Netzwerks von San Francisco ergibt sich für p daher ein Wert von ca. 0,4. Z wird auf den Wert 10.000 festgelegt, so dass die Verteilung der Knoteninteraktionen kaum von der Anzahl bereits getätigter Interaktionen abhängt (vgl. Kap. 5.4.2.2).
Tabelle 5.7 veranschaulicht den Durchschnitt $\emptyset(G_{FB}^{Syn})$ der Grapheigenschaften von 30 erzeugten Graphen, die zugehörigen Konfidenzintervalle $KI(G_{FB}^{Syn})$ und die Kennzahlen des Graphen $G_{FB}^{Syn^*}$, dessen Grapheigenschaften die größte Übereinstimmung mit denen von G_{FB} besitzen.
Ein Vergleich der Clustering-Charakteristik zweier unterschiedlich großer Graphen liefert leider keine brauchbaren Erkenntnisse über die Ähnlichkeit ihrer strukturel-

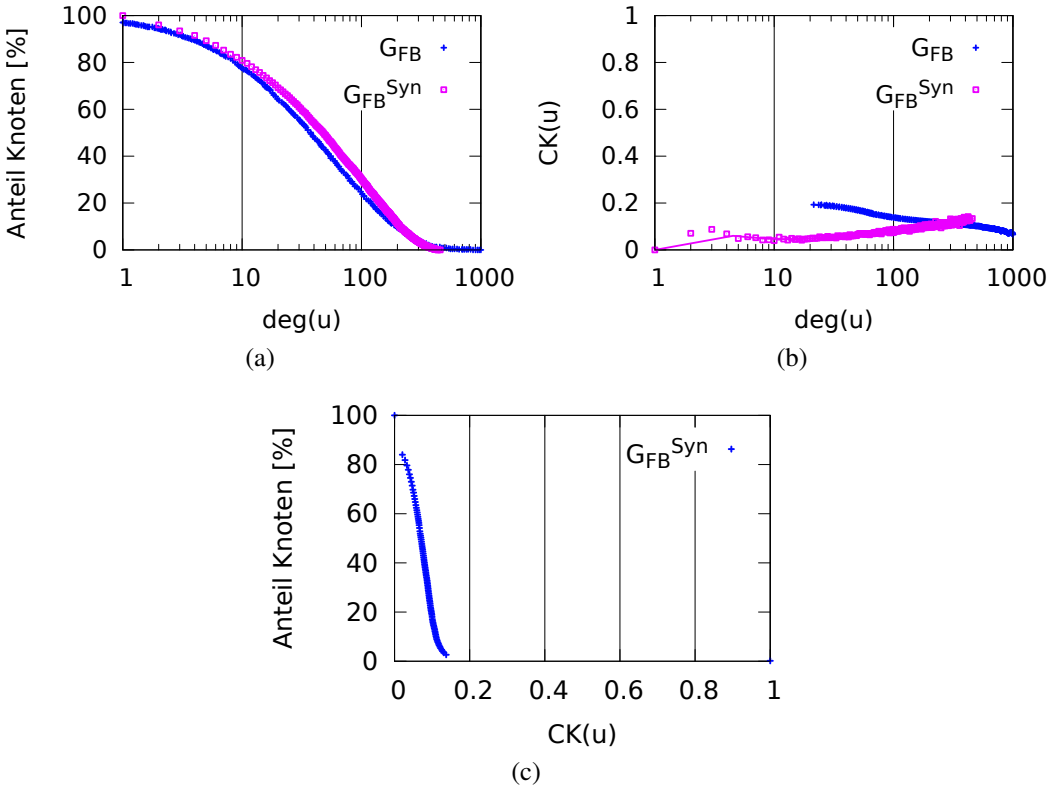
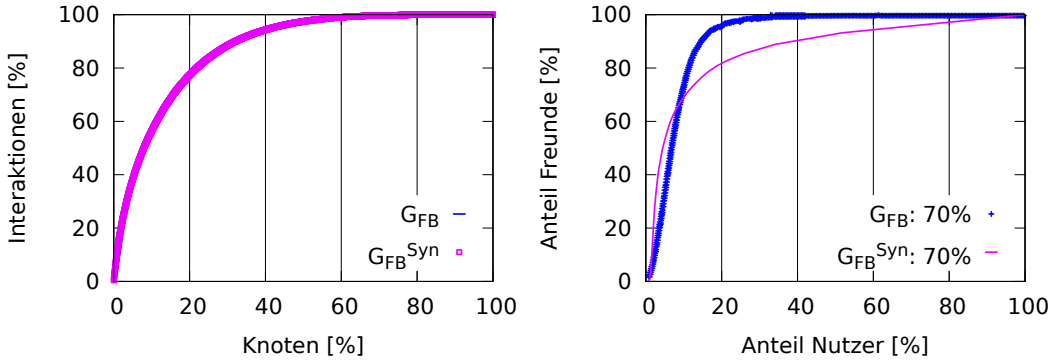


Abbildung 5.21: (a) Komplementär kumulative Häufigkeitsverteilungen der Knotengrade und (b) Verteilung lokaler Clustering-Koeffizienten als Funktion der Knotengrade für G_{FB} und $G_{FB}^{Syn^*}$. (c) Die komplementär kumulative Häufigkeitsverteilung lokaler Clustering-Koeffizienten ist nur für $G_{FB}^{Syn^*}$ dargestellt.

len Zusammensetzung. Aus Gründen der Vollständigkeit wird dennoch kurz auf diese Eigenschaften eingegangen.

Abbildung 5.21(a) zeigt für G_{FB} und $\text{ø}(G_{FB}^{Syn})$ die Verteilung der Knotengrade. Trotz der reduzierten Größe des generierten Graphen zeigt die Abbildung sehr deutlich, dass das Modell ohne den Einfluss von N und O ebenfalls eine Verteilung der Knotengrade entsprechend eines Potenzgesetzes erlaubt.

Offensichtlich weicht die Verteilung lokaler Clustering-Koeffizienten als Funktion der Knotengrade von $\text{ø}(G_{FB}^{Syn})$ deutlich von der von G_{FB} ab (vgl. Abb. 5.21(b)). Dass sich die Verteilung für $\text{ø}(G_{FB}^{Syn})$ eher links und die für G_{FB} eher rechts ansiedelt, ist auf die unterschiedliche Größe der beiden Graphen zurückzuführen. Der Grund für ein Abfallen der Clustering-Koeffizienten für G_{FB} und ein Anstieg für $\text{ø}(G_{FB}^{Syn})$ bei wachsendem Knotengrad kann an dieser Stelle nicht interpretiert werden. Unter Umständen sind die hohen Clustering-Koeffizienten für niedriggradige Knoten in G_{FB} auf die Art der Interpretation der gecrawlten Daten zurückzuführen. Dies wäre der Fall, wenn der Grad eines Knotens nicht über die tatsächlich gecrawlten Verbindungen, sondern über das Profilattribut „Anzahl der Freunde“ bestimmt



(a) Netzwerkinteraktionen (b) Knoteninteraktionen

Abbildung 5.22: (a) Verteilung der Netzwerkinteraktionen und (b) der Knoteninteraktionen für G_{FB} und $\text{ø}(G_{FB}^{Syn})$.

wurde. Nähere Informationen darüber konnten während der Evaluation leider nicht in Erfahrung gebracht werden.
Während der Analyse von G_{FB} wurde auch versucht, die Betrachtung der Clustering-Charakteristik lediglich auf das regionale Netzwerk von San Francisco zu beschränken. Leider erwies sich dieser Ansatz nicht als praktikabel, da nicht klar war, wie stark sich die strukturellen Eigenschaften des Graphen verändern, wenn man z.B. alle Kanten eliminiert, die sich nicht in der größten Zusammenhangskomponente dieses Netzwerkes befinden.
Für G_{FB} wurden zudem keine Ergebnisse zur Verteilung lokaler Clustering-Koeffizienten publiziert. Eine Interpretation der Verteilung lokaler Clustering-Koeffizienten als Funktion der Knotengrade in Korrelation mit der Häufigkeit ihres Auftretens ist daher nicht möglich. Abbildung 5.21(c) beschränkt sich auf die Darstellung der Verteilung lokaler Clustering-Koeffizienten für $\text{ø}(G_{FB}^{Syn})$.
Abbildung 5.22(a) zeigt die Verteilung der Netzwerkinteraktionen für G_{FB} und $\text{ø}(G_{FB}^{Syn})$. Offensichtlich liefert das Modell hier eine sehr gute Approximation des realen Datensatz. Die Verteilungen sind deckungsgleich.
Die zugehörigen Knoteninteraktionen sind in Abbildung 5.22(b) visualisiert. Für einen Anteil von Nutzern zwischen 0 und 8% erzielt das Modell eine sehr gute Übereinstimmung mit der Verteilung von G_{FB}. Darüber hinaus liegt die Verteilung von $\text{ø}(G_{FB}^{Syn})$ stets unterhalb der von G_{FB}. Dieser Sachverhalt lässt sich primär auf die unterschiedliche Größe der Graphen zurückführen. Je größer die modellierten Graphen ausfallen, desto stärker gleichen sich die beiden Verteilungen auch in diesem Segment einander an.

5.5 Zusammenfassung

Um frühzeitig ein fehlerhaftes oder ineffizientes Verhalten einer Netzwerkanwendung zu erkennen, bietet sich die Analyse innerhalb einer Simulationsumgebung an. Im Kontext von OSNs benötigt man dafür als Grundlage einen Datensatz, der

den sozialen Interaktionsgraphen des Netzwerks repräsentiert. Die bevorzugte Vorgehensweise zur Herleitung des Interaktionsgraphen besteht im Crawling des entsprechenden OSNs.

Eine Evaluation auf der Grundlage gecrawlter Datensätze ist mit Problemen verbunden. Zum einen besteht die Gefahr der Deanonymisierung sozialer Interaktionsgraphen, zum anderen mangelt es den gewonnen Ergebnissen an statistischer Konfidenz. Außerdem liefert das zeitaufwendige Crawling lediglich ein stark verzerrtes Abbild des tatsächlichen Zustands eines OSNs.

Als Alternative zum Crawling bietet sich die Modellierung sozialer Interaktionsgraphen an. Derzeit existieren bereits zahlreiche Ansätze zur Erzeugung von Netzwerkgraphen. Einige davon produzieren Graphen mit den strukturellen Eigenschaften von RSNs bzw. OSNs.

Für dezentrale OSNs wie z.B. Vegas ergeben sich jedoch besondere Anforderungen an ein geeignetes Modell. Dazu zählen die Berücksichtigung wiederkehrender Kosten sowie die Möglichkeit zur direkten Einflussnahme auf die Ausbildung von Gemeinschaften. Zudem unterstützt kein Ansatz die Modellierung von Netzwerkeffekten und Nutzerinteraktionen.

Um diese Lücke zu füllen, wurde ein generisches Modell zur Erzeugung von Interaktionsgraphen konzipiert. Das Modell kombiniert die Theorie des Preferential-Attachments mit der des Newman-Clusterings. Zusätzlich wird das Hinzufügen neuer Knoten und Kanten an die Bedingung zur Ausbildung von Gemeinschaften geknüpft. Diese Bedingung dient dazu, soziale Graphen mit den strukturellen Eigenschaften von RSNs sowie zentralisierter und dezentraler OSNs zu generieren. Des Weiteren berücksichtigt das Modell den Einfluss von Netzwerkeffekten auf das Wachstum eines OSNs sowie das Auftreten von Interaktionen zwischen seinen Nutzern.

Die Evaluation des Modells hat gezeigt, dass der Zwang zur Bildung von Gemeinschaften eine isolierte Einflussnahme auf verschiedene Grapheigenschaften erlaubt. Es können soziale Graphen mit derselben Anzahl von Knoten und Kanten, jedoch unterschiedlicher Verteilungen der Knotengrade z.B. entsprechend eines Potenzgesetzes mit oder auch ohne Cut-Off erzeugt werden. Die Modellierung von Netzwerkeffekten in Kombination mit Nutzerinteraktionen ermöglicht es, neue Algorithmen und Protokolle auch im Hinblick auf spezifische Evolutionsphasen eines OSNs zu untersuchen. Die logische Unterscheidung von Netzwerk- und Knoteninteraktionen erlaubt eine flexible Einflussnahme auf die Verteilung allgemeiner Interaktionen.

Mit Hilfe nichtlinearer Regression wurde eine Funktion zur Approximation der verschiedenen Grapheigenschaften identifiziert. Bisher ließ sich jedoch keine systematische Parametrisierung der Koeffizienten identifizieren. Die Bestimmung der Modellparameter auf der Basis eines Gleichungssystems ist derzeit nicht möglich. Die Lösung dieser Aufgabe stellt eine interessante Herausforderung für zukünftige Arbeiten dar.

Abschließend wurde das Modell auf der Grundlage des E-Mail-Netzwerks Enron und eines OSN-Datensatzes von Facebook evaluiert. Während ersterer dazu dient,

den Kontaktgraphen und das Kommunikationsverhalten eines RSNs zu approximieren, stellt letzterer den Interaktionsgraphen des derzeit dominierenden OSNs bereit. Als Ergebnis der Untersuchung auf die Eignung zur Erzeugung sozialer Interaktionsgraphen eines RSNs lieferte das Modell eine extrem hohe Übereinstimmung mit den Grapheigenschaften und dem Interaktionsaufkommen von Enron. Im Hinblick auf die Fähigkeit, das Interaktionsverhalten von OSNs zu simulieren, approximierte das Modell den Facebook-Datensatz ebenfalls erstaunlich genau. Bedingt durch den Informationsgehalt des verwendeten Datensatzes konnten über die Ähnlichkeit des modellierten und des realen Kontaktgraphen nur eingeschränkt Aussagen getroffen werden. Lediglich für die Verteilung der Knotengrade lässt sich festhalten, dass das Modell die reale Verteilung in Facebook sehr gut abbildet.

6 Zusammenfassung und Ausblick

6.1 Zusammenfassung

Die positiven Effekte sozialer Medien und die Vorteile der Vernetzung über OSNs sind unumstritten. Unglücklicherweise ist die damit einhergehende Bereitstellung privater Informationen jedoch mit erheblichen Risiken für die Sicherheit und den Schutz der Privatsphäre der Nutzer verbunden. Besonders deutlich wird das Gefahrenpotential an den jüngsten Beispielen des uneingeschränkten Zugriffs auf die Infrastruktur von OSN-Betreibern durch Nachrichtendienste und Ermittlungsbehörden in den USA.
Leider bieten die Integration von Sicherheits- und Privatsphäreneinstellungen keinen ausreichenden Schutz. Es ist das fehlende Vertrauen in die Betreiber, in ihre Angestellten und in die Qualität der verwendeten Software, das den Schutz der Privatsphäre und die Sicherheit der eigenen Identität limitiert.
Das wesentliche Problem etablierter OSNs wie Facebook besteht in ihrer zentralisierten Architektur. Der Zugriff auf den gesamten sozialen Graphen ermöglicht es den Betreibern, alle strukturellen und semantischen Verknüpfungen zwischen den Nutzern nachzuvollziehen.
Um diesem Problem entgegenzutreten, wurde im ersten Teil dieser Arbeit das sichere und die Privatsphäre schützende dezentrale OSN Vegas entwickelt. Vegas forciert die Abbildung der Semantik einer realen Beziehung auf den virtuellen Kontakt innerhalb eines OSNs. Der Schwerpunkt liegt auf dem Schutz des Rechts auf informationelle Selbstbestimmung und auf der Gewährleistung einer maximalen Anonymität seiner Nutzer. Die Umsetzung dieser Anforderungen beruht auf dem Konzept starker Vertrauensbeziehungen. Die Ausbildung einer Freundschaft in Vegas setzt die Existenz einer realen Beziehung voraus. Eine direkte Kommunikation ist auf die Mitglieder des eigenen Egonetzwerks beschränkt.
Vegas differenziert zwischen der Client-, Exchanger- und Datastore-Domäne. Während die Client-Domäne die Integration mehrerer Endgeräte forciert, sorgt die Exchanger-Domäne für den sicheren, anonymen und protokollunabhängigen Austausch von Nachrichten. Die Datastore-Domäne gewährleistet den permanenten Zugriff auf persönliche Inhalte und erfüllt die Anforderung an die Unterstützung mobiler Teilnehmer.
Technisch wird der Schutz des Rechts auf informationelle Selbstbestimmung der Nutzer auf der Grundlage verbindungsspezifischer Schlüsselpaare realisiert. Der flexible Einsatz von Exchangern und Datastores in Verbindung mit verbindungs-

spezifischen Schlüsselpaaren ermöglicht es Nutzern, den Grad ihrer Anonymität individuell zu beeinflussen.

Mit seiner restriktiven Architektur erfüllt Vegas die hohen Anforderungen an die Sicherheit und den Schutz der Privatsphäre. Damit verbunden sind jedoch starke Einschränkungen im Hinblick auf den gewohnten Funktionsumfang etablierter OSNs. Beispielsweise ist das Browsen anderer Nutzer außerhalb des eigenen Egonetzwerks in Vegas generell nicht möglich.

In OSNs wird ungemein viel Wissen akkumuliert, so dass sich die Suche nach Inhalten in deren sozialen Graphen geradezu aufdrängt. Außerdem erlaubt die Berücksichtigung sozialer Informationen eine individualisierte Filterung bei der Suche in anderen Quellen des WWW.

Um auch in restriktiven Systemen wie Vegas vom Wissen um die Semantik und die Struktur des sozialen Graphen zu profitieren, beschäftigte sich der zweite Teil dieser Arbeit mit den Möglichkeiten zur Suche und Verbreitung von Informationen in dezentralen OSNs. Dazu wurden zahlreiche Priorisierungsstrategien zur Weiterleitung von Suchanfragen im Hinblick auf die Messgrößen Erfolgsrate, durchschnittlich kürzester Suchpfad und Verteilung der Netzlast untersucht. Im Vordergrund der Analyse stand der Einfluss sozialer Kontextinformationen auf das Routing von Suchanfragen.

Neben Priorisierungsstrategien, die sich ausschließlich auf die Kontextinformationen des eigenen Egonetzwerks beziehen, wurden auch solche untersucht, die auf dem globalen Wissen des sozialen Graphen beruhen. Dabei galt es herauszufinden, ob und wie stark ein Aufweichen der Anforderungen an den Schutz der Privatsphäre zu einer Verbesserung der sozialen Suche führen kann.

Die Mehrzahl der Priorisierungsstrategien wurde auf der Basis von Grapheigenschaften wie der Degree-, der Closeness- oder der Betweenness-Zentralität formuliert. Zahlreiche Simulationen haben ergeben, dass diejenigen Priorisierungsstrategien zu besseren Ergebnisse führen, bei denen die Auswahl des nächsten Hops auf globalen Informationen wie der soziozentrischen Closeness- oder Betweenness-Zentralität beruht. Mit der egozentrischen Betweenness wurde eine Strategie identifiziert, die den Ergebnissen globaler Ansätze sehr nahe kommt. Deren Einsatz für die Weiterleitung von Suchanfragen ist ohne größere Anpassungen auch in dezentralen OSNs wie Vegas möglich.

Insgesamt hängt die Auswahl einer passenden Priorisierungsstrategie stark von der zu optimierenden Messgröße ab. Priorisierungsstrategien, die auf einer globalen Grapheigenschaft beruhen, verursachen eine stark asymmetrische Netzlast. Ansätze, die sich auf lokale Informationen wie das Egonetzwerk beschränken, liefern hingegen schlechtere Erfolgsraten. Steht die Vermeidung einer asymmetrischen Netzlast im Vordergrund, stellen solche Ansätze dennoch eine sinnvolle Alternative dar.

Im Kontext von OSNs benötigt man als Grundlage für die Simulation neuer Algorithmen oder Protokolle einen Datensatz, der den sozialen Graphen repräsentiert. Die Priorisierungsstrategien für das Weiterleiten sozialer Suchanfragen wurden ebenfalls auf der Basis realer OSN-Datensätze evaluiert. Die herkömmliche

Vorgehensweise für das Erzeugen solcher Datensätze besteht im Crawling des entsprechenden OSNs.
Die Simulation auf der Basis gecrawlter Datensätze ist jedoch mit zahlreichen Problemen verbunden. Es besteht die Gefahr der Deanonymisierung sozialer Interaktionsgraphen. Den Ergebnissen mangelt es an statistischer Konfidenz und das zeitaufwendige Crawling resultiert meist in einem stark verzerrten Abbild des tatsächlichen Zustands eines OSNs. Eine Alternative zum Crawling besteht in der Verwendung künstlich generierter sozialer Graphen.
Obwohl bereits einige Modelle existieren, die Graphen mit den Eigenschaften von OSNs produzieren, sind sie nur bedingt für die Erzeugung sozialer Graphen geeignet. Bei den meisten Ansätzen handelt es sich um reine Wachstumsmodelle, die sich auf die Erzeugung eines Kontaktgraphen beschränken.
Gerade bei der Analyse eines OSNs spielt aber auch das Interaktionsverhalten seiner Nutzer eine tragende Rolle. Hinzu kommt die Tatsache, dass das Wachstum eines OSNs einem Netzwerkeffekt folgt. Beide Faktoren werden in den derzeitigen Modellen nicht ausreichend berücksichtigt.
Für die Modellierung sicherer und die Privatsphäre schützender OSNs wie Vegas ergeben sich zudem sehr spezielle Anforderungen. Dazu zählen die Berücksichtigung wiederkehrender Kosten sowie die Möglichkeit zur direkten Einflussnahme auf die Ausbildung von Gemeinschaften.
Der dritte Teil dieser Arbeit befasste sich daher mit der Konzeption und der Evaluation eines generischen Modells zur Erzeugung sozialer Interaktionsgraphen. Um Graphen mit den strukturellen Eigenschaften von RSNs sowie von zentralisierten und dezentralen OSNs zu generieren, wird das Hinzufügen neuer Knoten und Kanten durch eine Kombination aus Preferential-Attachment und Newman-Clustering determiniert. Der gesamte Wachstumsprozess wird außerdem durch den Zwang zur Ausbildung von Gemeinschaften gesteuert. Schließlich simuliert das Modell einen Netzwerkeffekt und erlaubt eine direkte Einflussnahme auf die Verteilung von Interaktionen. Damit können Simulationen nicht nur für den finalen Zustand, sondern auch für spezifische Evolutionsphasen eines OSNs durchgeführt werden.
Die Evaluation des Modells hat gezeigt, dass der Zwang zur Bildung von Gemeinschaften eine isolierte Einflussnahme auf verschiedene Grapheigenschaften erlaubt. Diese Möglichkeit wird bisher von keinem anderen Ansatz unterstützt. Auch die Verteilung der Interaktionen lässt sich durch eine entsprechende Parametrisierung flexibel steuern.
Die Eignung des Modells zur Erzeugung sozialer Graphen wurde am Beispiel des E-Mail-Netzwerks Enron und eines OSN-Datensatzes von Facebook evaluiert. Da keine ausreichend umfangreichen Datensätze von RSNs existieren, diente der Enron-Datensatz als Approximation für einen Interaktionsgraphen eines RSNs.
Die Modellierung der beiden Datensätze hat gezeigt, dass der generische Ansatz Graphen mit einer extrem genauen Übereinstimmung in den strukturellen Eigenschaften und im Interaktionsaufkommen von Enron produziert.
Im Falle des Facebook-Datensatzes approximiert das Modell das Interaktionsverhalten ebenfalls erstaunlich genau. Über die Ähnlichkeit des modellierten und des

realen Kontaktgraphen konnten hingegen keine fundierten Erkenntnisse gesammelt werden. Für die Verteilung der Knotengrade lässt sich jedoch festhalten, dass das Modell die reale Verteilung in Facebook sehr gut abzubilden vermag.

6.2 Ausblick

Die hier behandelten Fragestellungen und die gesammelten Erkenntnisse bieten das Potential für weitere Forschungsarbeiten.
Ein bisher ungelöstes Problem von Vegas stellt der Parallelbetrieb mehrerer Endgeräte innerhalb der Client-Domäne dar. Auch die Entwicklung einer Synchronisationslösung steht noch aus. Im Hinblick auf die vorgeschlagenen Priorisierungsstrategien stellt sich die Frage, wie sich diese konkret in einen Prototyp integrieren lassen. Beispielsweise existiert bisher noch kein Ähnlichkeitsmaß, das man für den Vergleich von Suchanfragen und Profilattributen verwenden könnte, um die entsprechende Priorisierungsstrategie zu implementieren.
Bei der Untersuchung des Modells zur Erzeugung von Interaktionsgraphen konnte bereits eine Funktion identifiziert werden, die die Verteilung der unterschiedlichen globalen Grapheigenschaften beschreibt. Bisher konnte jedoch keine systematische Parametrisierung der Funktion gefunden werden. Die Analyse eines umfangreicheren Datensatzes künstlich erzeugter Interaktionsgraphen könnte in Zukunft dabei helfen, eine funktionale Abhängigkeit in der Verteilung der Koeffizienten zu identifizieren.
Um einen Einblick in die Charakteristik sicherer und die Privatsphäre schützender OSNs zu erlangen, wurde der Prototyp Vegas Mobile implementiert. Dieser realisiert das Konzept von Vegas und steht zum öffentlich Download zur Verfügung. Ausgehend von einer hinreichend starken Verbreitung der Anwendung könnten in naher Zukunft erste Rückschlüsse auf die strukturellen Gegebenheiten und das Interaktionsverhalten innerhalb dezentraler OSNs gezogen werden. In diesem Zusammenhang ergeben sich die interessantesten Aufgabenfelder für zukünftige Forschungsarbeiten.
Sobald Informationen über die strukturellen und interaktiven Gegebenheiten von Vegas zur Verfügung stehen, kann das generische Modell auf seine Eignung zur Erzeugung von Interaktionsgraphen mit diesen spezifischen Eigenschaften hin untersucht werden. Die mit dem Modell erzeugten Graphen dienen wiederum als Basis für die Auswertung der entwickelten Priorisierungsstrategien für die Weiterleitung von Suchanfragen in Vegas. Für die Zukunft ist geplant, eine der Priorisierungsstrategien in Vegas Mobile zu integrieren, um darauf basierend Erkenntnisse über die Unterstützung einer sozialen Suche durch Vegas Mobile und seine Nutzer zu sammeln. Schließlich ließe sich mit diesen Informationen überprüfen, inwiefern sich Priorisierungsstrategien, die auf der Ähnlichkeit von Suchanfragen und Profilattributen basieren, für den Einsatz in dezentralen OSNs wie Vegas eignen.

Literaturverzeichnis

[1] L. A. Adamic, R. M. Lukose, A. R. Puniyani, and B. A. Huberman, "Search in power-law networks," *Physical Review E*, vol. 64, pp. 046135+, Sept. 2001.

[2] A. Ahmed and E. P. Xing, "Recovering time-varying networks of dependencies in social and biological studies," *Proceedings of the National Academy of Sciences*, vol. 106, no. 29, pp. 11878–11883, 2009.

[3] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong, "Analysis of topological characteristics of huge online social networking services," in *Proceedings of the 16th international conference on World Wide Web*, WWW '07, (New York, NY, USA), pp. 835–844, ACM, 2007.

[4] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Rev. Mod. Phys.*, vol. 74, pp. 47–97, Jan 2002.

[5] R. Albert, H. Jeong, and A.-L. Barabási, "Internet: Diameter of the world-wide web," in *Nature*, vol. 401, pp. 130–131, Macmillan Magazines Ltd., sept. 1999.

[6] L. A. N. Amaral, A. Scala, M. Barthélémy, and H. E. Stanley, "Classes of small-world networks," *Proceedings of the National Academy of Sciences*, vol. 97, no. 21, pp. 11149–11152, 2000.

[7] E. Amitay, D. Carmel, N. Har'El, S. Ofek-Koifman, A. Soffer, S. Yogev, and N. Golbandi, "Social search and discovery using a unified approach," in *Proceedings of the 20th ACM conference on Hypertext and hypermedia*, HT '09, (New York, NY, USA), pp. 199–208, ACM, 2009.

[8] S. Androutsellis-Theotokis and D. Spinellis, "A survey of peer-to-peer content distribution technologies," *ACM Comput. Surv.*, vol. 36, pp. 335–371, Dec. 2004.

[9] Z. Anwar, W. Yurcik, V. Pandey, A. Shankar, I. Gupta, and R. H. Campbell, "Leveraging Social-Network infrastructure to improve Peer-to-Peer overlay performance: Results from orkut," *CoRR*, vol. abs/cs/0509095, 2005.

[10] M. Artigas, P. Lopez, J. Ahullo, and A. Skarmeta, "Cyclone: a novel design schema for hierarchical dhts," *Peer-to-Peer Computing, 2005. P2P 2005. Fifth IEEE International Conference on*, pp. 49–56, Aug.-2 Sept. 2005.

[11] L. Backstrom, P. Boldi, M. Rosa, J. Ugander, and S. Vigna, "Four degrees of separation," *arXiv preprint arXiv:1111.4570*, 2012.

[12] L. Backstrom, C. Dwork, and J. Kleinberg, "Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography," *Commun. ACM*, vol. 54, pp. 133–141, Dec. 2011.

[13] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan, "Group formation in large social networks: membership, growth, and evolution," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, (New York, NY, USA), pp. 44–54, ACM, 2006.

[14] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin, "Persona: an online social network with user-defined privacy," *SIGCOMM Comput. Commun. Rev.*, vol. 39, pp. 135–146, Aug. 2009.

[15] X. Bai, M. Bertier, R. Guerraoui, A.-M. Kermarrec, and V. Leroy, "Gossiping personalized queries," in *Proceedings of the 13th International Conference on Extending Database Technology*, EDBT '10, (New York, NY, USA), pp. 87–98, ACM, 2010.

[16] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic, "The role of social networks in information diffusion," in *Proceedings of the 21st international conference on World Wide Web*, WWW '12, (New York, NY, USA), pp. 519–528, ACM, 2012.

[17] M. Balduzzi, C. Platzer, T. Holz, E. Kirda, D. Balzarotti, and C. Kruegel, "Abusing social networks for automated user profiling," in *Proceedings of the 13th international conference on Recent advances in intrusion detection*, RAID'10, (Berlin, Heidelberg), pp. 422–441, Springer-Verlag, 2010.

[18] S. Bao, G. Xue, X. Wu, Y. Yu, B. Fei, and Z. Su, "Optimizing web search using social annotations," in *Proceedings of the 16th international conference on World Wide Web*, WWW '07, (New York, NY, USA), pp. 501–510, ACM, 2007.

[19] A. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[20] A. Barabási, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, and T. Vicsek, "Evolution of the social network of scientific collaborations," *Physica A: Statistical Mechanics and its Applications*, vol. 311, no. 3–4, pp. 590 – 614, 2002.

[21] A.-L. Barabási, R. Albert, and H. Jeong, "Mean-field theory for scale-free random networks," *Physica A: Statistical Mechanics and its Applications*, vol. 272, no. 1–2, pp. 173 – 187, 1999.

[22] M. Bender, T. Crecelius, M. Kacimi, S. Michel, J. X. Parreira, and G. Weikum, "Peer-to-peer information search: Semantic, social, or spiritual?," *IEEE Data Eng. Bull.*, vol. 30, no. 2, pp. 51–60, 2007.

[23] F. Benevenuto, T. Rodrigues, M. Cha, and V. Almeida, "Characterizing user behavior in online social networks," in *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, IMC '09, (New York, NY, USA), pp. 49–62, ACM, 2009.

[24] H. R. Bernard, P. D. Killworth, M. J. Evans, C. McCarty, and G. A. Shelley, "Studying social relations cross-culturally," *Ethnology*, vol. 27, no. 2, pp. pp. 155–179, 1988.

[25] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of the 2007 IEEE Symposium on Security and Privacy*, SP '07, (Washington, DC, USA), pp. 321–334, IEEE Computer Society, 2007.

[26] G. Bianconi and A.-L. Barabási, "Competition and multiscaling in evolving networks," *EPL (Europhysics Letters)*, vol. 54, no. 4, p. 436, 2001.

[27] L. Bilge, T. Strufe, D. Balzarotti, and E. Kirda, "All your contacts are belong to us: automated identity theft attacks on social networks," in *Proceedings of the 18th international conference on World wide web*, WWW '09, (New York, NY, USA), pp. 551–560, ACM, 2009.

[28] N. Bisnik and A. Abouzeid, "Modeling and analysis of random walk search algorithms in p2p networks," in *HOT-P2P '05: Proceedings of the Second International Workshop on Hot Topics in Peer-to-Peer Systems*, (Washington, DC, USA), pp. 95–103, IEEE Computer Society, 2005.

[29] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 10, pp. 8–+, Oct. 2008.

[30] J. Bonneau, J. Anderson, R. Anderson, and F. Stajano, "Eight friends are enough: social graph approximation via public listings," in *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*, SNS '09, (New York, NY, USA), pp. 13–18, ACM, 2009.

[31] N. Borenstein and N. Freed, "MIME (Multipurpose Internet Mail Extensions): Mechanisms for Specifying and Describing the Format of Internet Message Bodies." RFC 1341 (Proposed Standard), June 1992. Obsoleted by RFC 1521.

[32] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu, "Design and analysis of a social botnet," *Comput. Netw.*, vol. 57, pp. 556–578, Feb. 2013.

[33] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu, "The socialbot network: when bots socialize for fame and money," in *Proceedings of the 27th Annual Computer Security Applications Conference*, ACSAC '11, (New York, NY, USA), pp. 93–102, ACM, 2011.

[34] D. Boyd and N. B. Ellison, "Social network sites: Definition, history, and scholarship," *Journal of Computer-Mediated Communication*, vol. 13, pp. 210–230, Oct. 2007.

[35] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener, "Graph structure in the web," *Computer Networks*, vol. 33, no. 1–6, pp. 309 – 320, 2000.

[36] G. Brown, T. Howe, M. Ihbe, A. Prakash, and K. Borders, "Social networks and context-aware spam," in *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, CSCW '08, (New York, NY, USA), pp. 403–412, ACM, 2008.

[37] R. Bryce, *Pipe dreams: Greed, ego, and the death of Enron*. PublicAffairs, 2004.

[38] S. Buchegger and A. Datta, "A case for P2P infrastructure for social networks - opportunities & challenges," in *WONS'09*, pp. 161–168, IEEE, Feb 2009.

[39] S. Buchegger, D. Schiöberg, L.-H. Vu, and A. Datta, "Peerson: P2p social networking: early experiences and insights," in *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*, SNS '09, (New York, NY, USA), pp. 46–52, ACM, 2009.

[40] E. Bulut and B. K. Szymanski, "Exploiting friendship relations for efficient routing in mobile social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 12, pp. 2254–2265, 2012.

[41] B. Carlsson and R. Gustavsson, "The rise and fall of napster - an evolutionary approach," in *Proceedings of the 6th International Computer Science Conference on Active Media Technology*, AMT '01, (London, UK, UK), pp. 347–354, Springer-Verlag, 2001.

[42] D. Carmel, N. Zwerdling, I. Guy, S. Ofek-Koifman, N. Har'el, I. Ronen, E. Uziel, S. Yogev, and S. Chernov, "Personalized social search based on the user's social network," in *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, (New York, NY, USA), pp. 1227–1236, ACM, 2009.

[43] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach, "Secure routing for structured peer-to-peer overlay networks," *SIGOPS Oper. Syst. Rev.*, vol. 36, pp. 299–314, Dec. 2002.

[44] P. Cauwels and D. Sornette, "Quis pendit ipsa pretia: facebook valuation and diagnostic of a bubble based on nonlinear demographic dynamics," *Journal of Portfolio Management*, vol. 38, no. 2, 2011.

[45] M. Cha, A. Mislove, and K. P. Gummadi, "A Measurement-driven Analysis of Information Propagation in the Flickr Social Network," in *WWW'09*, April 2009.

[46] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker, "Making gnutella-like p2p systems scalable," in *SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, (New York, NY, USA), pp. 407–418, ACM, 2003.

[47] S. Chib and E. Greenberg, "Understanding the Metropolis-Hastings algorithm," *The American Statistician*, vol. 49, no. 4, pp. 327–335, 1995.

[48] V. Cholvi, P. Felber, and E. Biersack, "Efficient search in unstructured peer-to-peer networks," in *Proceedings of the sixteenth annual ACM symposium on Parallelism in algorithms and architectures*, SPAA '04, pp. 271–272, ACM, 2004.

[49] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Phys. Rev. E*, vol. 70, p. 066111, Dec 2004.

[50] J. S. Coleman, *Foundations of Social Theory*. Cambridge, MA: Harvard University Press, 1994.

[51] A. Crespo and H. Garcia-Molina, "Routing indices for peer-to-peer systems," in *Proceedings of the 22 nd International Conference on Distributed Computing Systems (ICDCS'02)*, ICDCS '02, (Washington, DC, USA), pp. 23–, IEEE Computer Society, 2002.

[52] A. Crespo and H. G. Molina, "Semantic overlay networks for P2P systems," tech. rep., Computer Science Department, Stanford University, 2002.

[53] M. Crispin, "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1." RFC 3501 (Proposed Standard), Mar. 2003. Updated by RFCs 4466, 4469, 4551, 5032, 5182, 5738, 6186.

[54] J. Croll and S. Weber, "Deine daten im netz . . . ," in *Die Alten und das Netz* (B. Kampmann, B. Keller, M. Knippelmeyer, and F. Wagner, eds.), pp. 157–170, Gabler Verlag, 2012.

[55] L. A. Cutillo, R. Molva, and T. Strufe, "Privacy preserving social networking through decentralization," in *2009 Sixth International Conference on Wireless On-Demand Network Systems and Services (WONS)*, pp. 145–152, IEEE, Feb. 2009.

[56] L. A. Cutillo, R. Molva, and T. Strufe, "Safebook: A privacy-preserving online social network leveraging on real-life trust," *Communications Magazine, IEEE*, vol. 47, pp. 94–101, Dec. 2009.

[57] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 09, p. P09008, 2005.

[58] N. Daswani, H. Garcia-Molina, and B. Yang, "Open problems in data-sharing peer-to-peer systems," in *Database Theory — ICDT 2003* (D. Calvanese, M. Lenzerini, and R. Motwani, eds.), vol. 2572 of *Lecture Notes in Computer Science*, pp. 1–15, Springer Berlin Heidelberg, 2002.

[59] J. Davidsen, H. Ebel, and S. Bornholdt, "Emergence of a small world from local interactions: Modeling acquaintance networks," *Phys. Rev. Lett.*, vol. 88, p. 128701, Mar 2002.

[60] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE transactions on Information Theory*, vol. 22, 1976.

[61] P. S. Dodds, R. Muhamad, and D. J. Watts, "An experimental study of search in global social networks," *Science*, vol. 301, no. 5634, pp. 827–829, 2003.

[62] P. Domingos and M. Richardson, "Mining the network value of customers," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '01, (New York, NY, USA), pp. 57–66, ACM, 2001.

[63] S. N. Dorogovtsev, J. F. F. Mendes, and A. N. Samukhin, "Structure of growing networks with preferential linking," *Phys. Rev. Lett.*, vol. 85, pp. 4633–4636, Nov 2000.

[64] J. R. Douceur, "The sybil attack," in *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, (London, UK), pp. 251–260, Springer-Verlag, 2002.

[65] N. Du, B. Wu, X. Pei, B. Wang, and L. Xu, "Community detection in large-scale social networks," in *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, WebKDD/SNA-KDD '07, (New York, NY, USA), pp. 16–25, ACM, 2007.

[66] R. Dunbar and M. Spoors, "Social networks, support cliques, and kinship," *Human Nature*, vol. 6, no. 3, pp. 273–290, 1995.

[67] L. Dusseault, "HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV)." RFC 4918 (Proposed Standard), June 2007. Updated by RFC 5689.

[68] P. Erdös and A. Rényi, "On random graphs, I," *Publicationes Mathematicae (Debrecen)*, vol. 6, pp. 290–297, 1959.

[69] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," in *Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*, SIGCOMM '99, (New York, NY, USA), pp. 251–262, ACM, 1999.

[70] T. Fararo and M. Sunshine, *A study of a biased friendship net*. Youth Development Center, Syracuse University, 1964.

[71] A. Fast, D. Jensen, and B. N. Levine, "Creating social networks to improve peer-to-peer networking," in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, KDD '05, (New York, NY, USA), pp. 568–573, ACM, 2005.

[72] S. Fortunato and M. Barthélemy, "Resolution limit in community detection," *Proceedings of the National Academy of Sciences*, vol. 104, no. 1, pp. 36–41, 2007.

[73] A. Franzen and D. Hangartner, "Social networks and labour market outcomes: The non-monetary benefits of social capital," *European Sociological Review*, vol. 22, no. 4, pp. 353–368, 2006.

[74] M. Freedman and D. Maziéres, *Peer-to-Peer Systems II*, vol. 2735 of *Lecture Notes in Computer Science*, ch. Sloppy Hashing and Self-Organizing Clusters, pp. 45–55. Springer Berlin / Heidelberg, October 2003.

[75] L. C. Freeman, "Centrality in social networks conceptual clarification," *Social Networks*, vol. 1, no. 3, pp. 215–239, 1979.

[76] L. C. Freeman, "A set of measures of centrality based upon betweenness," *Sociometry*, vol. 40, no. 1, pp. 35–41, 1977.

[77] P. Ganesan, K. Gummadi, and H. Garcia-Molina, "Canon in g major: Designing dhts with hierarchical structure," in *Distributed Computing Systems, 2004. Proceedings. 24th International Conference on*, (Los Alamitos, CA, USA), pp. 263–272, IEEE Computer Society, 2004.

[78] W. Gao, G. Cao, T. L. Porta, and J. Han, "On exploiting transient social contact patterns for data forwarding in delay-tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 1, pp. 151–165, 2013.

[79] L. Garcés-Erice, E. W. Biersack, P. A. Felber, K. W. Ross, and G. Urvoy-keller, "Hierarchical peer-to-peer systems," in *Proceedings of ACM/IFIP International Conference on Parallel and Distributed Computing (Euro-Par)*, pp. 643–657, 2003.

[80] S. Garfinkel, "Pretty good privacy (pgp)," in *Encyclopedia of Computer Science*, pp. 1421–1422, Chichester, UK: John Wiley and Sons Ltd., 2003.

[81] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[82] M. Gjoka, M. Kurant, C. Butts, and A. Markopoulou, "Walking in facebook: A case study of unbiased sampling of osns," in *INFOCOM, 2010 Proceedings IEEE*, pp. 1–9, 2010.

[83] C. Gkantsidis, M. Mihail, and A. Saberi, "Random walks in peer-to-peer networks: algorithms and evaluation," *Perform. Eval.*, vol. 63, no. 3, pp. 241–263, 2006.

[84] S. Goel, R. Muhamad, and D. Watts, "Social search in ßmall-worldëxperiments," in *Proceedings of the 18th international conference on World wide web*, WWW '09, (New York, NY, USA), pp. 701–710, ACM, 2009.

[85] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "Learning influence probabilities in social networks," in *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, (New York, NY, USA), pp. 241–250, ACM, 2010.

[86] K. Graffi, C. Gross, D. Stingl, D. Hartung, A. Kovacevic, and R. Steinmetz, "Lifesocial.kom: A secure and p2p-based solution for online social networks," in *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*, pp. 554–558, 2011.

[87] M. Granovetter, "The Strength of Weak Ties," *The American Journal of Sociology*, vol. 78, no. 6, pp. 1360–1380, 1973.

[88] J. Guare, *Six degrees of separation*. New York: Vintage Books, 1 ed., 1990.

[89] S. Guha, K. Tang, and P. Francis, "NOYB: privacy in online social networks," in *WOSN '08*, pp. 49–54, ACM, 2008.

[90] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan, "Measurement, modeling, and analysis of a peer-to-peer file-sharing workload," in *SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles*, pp. 314–329, ACM, 2003.

[91] M. S. Handcock, A. E. Raftery, and J. M. Tantrum, "Model-based clustering for social networks," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, vol. 170, no. 2, pp. 301–354, 2007.

[92] C.-W. Hang, Y. Wang, and M. P. Singh, "Operators for propagating trust and their evaluation in social networks," in *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '09, (Richland, SC), pp. 1025–1032, International Foundation for Autonomous Agents and Multiagent Systems, 2009.

[93] M. Hasan and M. Zaki, "A survey of link prediction in social networks," in *Social Network Data Analytics* (C. C. Aggarwal, ed.), pp. 243–275, Springer US, 2011.

[94] A. Haugen, "Abstract: The open graph protocol design decisions," in *The Semantic Web – ISWC 2010* (P. Patel-Schneider, Y. Pan, P. Hitzler, P. Mika, L. Zhang, J. Pan, I. Horrocks, and B. Glimm, eds.), vol. 6497 of *Lecture Notes in Computer Science*, pp. 338–338, Springer Berlin Heidelberg, 2010.

[95] P. Heymann, G. Koutrika, and H. Garcia-Molina, "Can social bookmarking improve web search?," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, (New York, NY, USA), pp. 195–206, ACM, 2008.

[96] P. Holme and B. J. Kim, "Growing scale-free networks with tunable clustering," *Phys. Rev. E*, vol. 65, p. 026107, Jan 2002.

[97] H. Hu and X. Wang, "Evolution of a large online social network," *Physics Letters A*, vol. 373, no. 12–13, pp. 1105–1110, 2009.

[98] M. Huber, S. Kowalski, M. Nohlberg, and S. Tjoa, "Towards automating social engineering using social networking sites," in *Proceedings of the 2009 International Conference on Computational Science and Engineering - Volume 03*, CSE '09, (Washington, DC, USA), pp. 117–124, IEEE Computer Society, 2009.

[99] M. Huber, M. Mulazzani, E. Weippl, G. Kitzler, and S. Goluch, "Friend-in-the-middle attacks: Exploiting social networking sites for spam," *IEEE Internet Computing*, vol. 15, pp. 28–34, May 2011.

[100] M. Huber, M. Mulazzani, E. Weippl, G. Kitzler, and S. Goluch, "Exploiting social networking sites for spam," in *Proceedings of the 17th ACM conference on Computer and communications security*, CCS '10, (New York, NY, USA), pp. 693–695, ACM, 2010.

[101] S. Ioannidis and A. Chaintreau, "On the strength of weak ties in mobile social networks," in *SNS '09*, pp. 19–25, ACM, Mar. 2009.

[102] D. Irani, M. Balduzzi, D. Balzarotti, E. Kirda, and C. Pu, "Reverse social engineering attacks in online social networks," in *Proceedings of the 8th international conference on Detection of intrusions and malware, and vulnerability assessment*, DIMVA'11, (Berlin, Heidelberg), pp. 55–74, Springer-Verlag, 2011.

[103] D. Irani, S. Webb, K. Li, and C. Pu, "Large online social footprints–an emerging threat," in *2009 International Conference on Computational Science and Engineering*, pp. 271–276, IEEE, 2009.

[104] I. Jawhar and J. Wu, "A two-level random walk search protocol for peer-to-peer networks," in *Proc. of the 8th World Multi-Conference on Systemics, Cybernetics and Informatics*, 2004.

[105] H. Jeong, Z. Néda, and A. L. Barabási, "Measuring preferential attachment in evolving networks," *EPL (Europhysics Letters)*, vol. 61, no. 4, p. 567, 2003.

[106] E. M. Jin, M. Girvan, and M. E. J. Newman, "Structure of growing social networks," *Phys. Rev. E*, vol. 64, p. 046132, Sep 2001.

[107] L. Jin, J. B. Joshi, and M. Anwar, "Mutual-friend based attacks in social network systems," *Computers & Security*, vol. 37, no. 0, pp. 15 – 30, 2013.

[108] M. A. Jovanovic, F. S. Annexstein, and K. A. Berman, "Scalability issues in large peer to peer networks - a case study of gnutella," tech. rep., Univ. of Cincinati, 2001.

[109] V. Kalogeraki, D. Gunopulos, and Z. D. Yazti, "A local search mechanism for peer-to-peer networks," in *Proceedings of the eleventh international conference on Information and knowledge management*, pp. 300–307, ACM Press, 2002.

[110] A. M. Kaplan and M. Haenlein, "Users of the world, unite! the challenges and opportunities of social media," *Business Horizons*, vol. 53, no. 1, pp. 59 – 68, 2010.

[111] W. Ke and J. Mostafa, "Strong ties vs. weak ties: Studying the clustering paradox for decentralized search," in *Proc. LSDS-IR*, 2009.

[112] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '03, (New York, NY, USA), pp. 137–146, ACM, 2003.

[113] B. J. Kim, C. N. Yoon, S. K. Han, and H. Jeong, "Path finding strategies in scale-free networks.," *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 65, Feb. 2002.

[114] J. Kleinberg, "The small-world phenomenon: an algorithm perspective," in *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, STOC '00, (New York, NY, USA), pp. 163–170, ACM, 2000.

[115] B. Klimt and Y. Yang, "Introducing the enron corpus," in *First conference on email and anti-spam (CEAS)*, 2004.

[116] I. Konstas, V. Stathopoulos, and J. M. Jose, "On social networks and collaborative recommendation," in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, (New York, NY, USA), pp. 195–202, ACM, 2009.

[117] P. L. Krapivsky and S. Redner, "Network growth by copying.," *Physical Review E - Statistical, Nonlinear and Soft Matter Physics*, vol. 71, no. 3 Pt 2A, p. 036118, 2005.

[118] P. L. Krapivsky, S. Redner, and F. Leyvraz, "Connectivity of growing random networks," *Phys. Rev. Lett.*, vol. 85, pp. 4629–4632, Nov 2000.

[119] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication." RFC 2104 (Informational), Feb. 1997. Updated by RFC 6151.

[120] B. Krishnamurthy and C. E. Wills, "On the leakage of personally identifiable information via online social networks," *SIGCOMM Comput. Commun. Rev.*, vol. 40, pp. 112–117, January 2010.

[121] S. O. Krumke and H. Noltemeier, *Graphentheoretische Konzepte und Algorithmen*. Vieweg+Teubner Verlag, Jun 2012.

[122] R. Kumar, J. Novak, and A. Tomkins, "Structure and evolution of online social networks," in *Link Mining: Models, Algorithms, and Applications* (P. S. Yu, J. Han, and C. Faloutsos, eds.), pp. 337–357, Springer New York, 2010.

[123] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?," in *Proceedings of the 19th international conference on World wide web*, WWW '10, (New York, NY, USA), pp. 591–600, ACM, 2010.

[124] A. N. Langville and C. D. Meyer, *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton, NJ, USA: Princeton University Press, 2012.

[125] P. Lazarsfeld, R. Merton, *et al.*, "Friendship as a social process: A substantive and methodological analysis," *Freedom and control in modern society*, vol. 18, no. 1, pp. 18–66, 1954.

[126] P. Leach, M. Mealling, and R. Salz, "A Universally Unique IDentifier (UUID) URN Namespace." RFC 4122 (Proposed Standard), July 2005.

[127] A. Lenhart, K. Purcell, A. Smith, and K. Zickuhr, *Social media & mobile internet use among teens and young adults*. Pew Internet & American Life Project Washington, DC, 2010.

[128] J. Leskovec and C. Faloutsos, "Scalable modeling of real graphs using kronecker multiplication," in *ICML '07*, pp. 497–504, ACM, 2007.

[129] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, "Statistical properties of community structure in large social and information networks," in *Proceedings of the 17th international conference on World Wide Web*, WWW '08, (New York, NY, USA), pp. 695–704, ACM, 2008.

[130] J. Leskovec and E. Horvitz, "Planetary-Scale Views on an Instant-Messaging Network," *ArXiv e-prints*, Mar. 2008.

[131] K. Levenberg, "A method for the solution of certain problems in least squares," *Quarterly of Applied Mathematics*, vol. 2, pp. 164–168, 1944.

[132] J. Li, B. Thau, L. Joseph, M. Hellerstein, and M. F. Kaashoek, "On the feasibility of peer-to-peer web indexing and search," in *Peer-to-Peer Systems II*, vol. 2735 of *Lecture Notes in Computer Science*, pp. 207–215, Springer Berlin/Heidelberg, 2003.

[133] Z. Li and H. Shen, "Sedum: Exploiting social networks in utility–based distributed routing for dtns," *IEEE Transactions on Computers*, vol. 62, no. 1, pp. 83–97, 2013.

[134] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proceedings of the twelfth international conference on Information and knowledge management*, CIKM '03, (New York, NY, USA), pp. 556–559, 2003.

[135] L. Liu and Y. Jing, "A survey on social-based routing and forwarding protocols in opportunistic networks," *Computer and Information Technology, International Conference on*, vol. 0, pp. 635–639, 2012.

[136] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, "A survey and comparison of peer-to-peer overlay network schemes," *IEEE Communications Surveys and Tutorials*, vol. 7, pp. 72–93, 2005.

[137] M. M. Lucas and N. Borisov, "Flybynight: mitigating the privacy risks of social networking," in *Proceedings of the 7th ACM workshop on Privacy in the electronic society*, WPES '08, (New York, NY, USA), pp. 1–8, ACM, 2008.

[138] W. Luo, J. Liu, J. Liu, and C. Fan, "An analysis of security in social networks," in *Proceedings of the 2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing*, DASC '09, (Washington, DC, USA), pp. 648–651, IEEE Computer Society, 2009.

[139] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker, "Search and replication in unstructured peer-to-peer networks," in *ICS '02: Proceedings of the 16th international conference on Supercomputing*, (New York, NY, USA), pp. 84–95, ACM, 2002.

[140] Q. Lv, S. Ratnasamy, and S. Shenker, "Can heterogeneity make gnutella scalable?," in *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, (London, UK), pp. 94–103, Springer-Verlag, 2002.

[141] G. Lüpken-Räder, *Datenschutz von A-Z.* Haufe-Lexware GmbH, 2012.

[142] P. Mahadevan, D. Krioukov, K. Fall, and A. Vahdat, "Systematic topology analysis and generation using degree correlations," in *SIGCOMM '06*, pp. 135–146, ACM, 2006.

[143] P. V. Marsden, "Egocentric and sociocentric measures of network centrality," *Social Networks*, vol. 24, no. 4, pp. 407 – 422, 2002.

[144] P. Maymounkov and D. Mazières, "Kademlia: A peer-to-peer information system based on the xor metric," in *Peer-to-Peer Systems* (P. Druschel, F. Kaashoek, and A. Rowstron, eds.), vol. 2429 of *Lecture Notes in Computer Science*, pp. 53–65, Springer Berlin Heidelberg, 2002.

[145] B. McLean and P. Elkind, "The smartest guys in the room: The amazing rise and scandalous fall of enron," *New York*, 2003.

[146] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual Review of Sociology*, vol. 27, pp. pp. 415–444, 2001.

[147] B. Metcalfe, "Metcalfe's law: A network becomes more valuable as it reaches more users," *InfoWorld*, vol. 17, no. 40, pp. 53–54, 1995.

[148] S. Milgram, "The small world problem," *Psychology Today*, vol. 2, no. 1, pp. 60–67, 1967.

[149] A. Mislove and P. Druschel, "Providing administrative control and autonomy in peer-to-peer overlays," in *Proceedings of the 3rd International Workshop on Peer-to-Peer Systems (IPTPS'04)*, February 2004.

[150] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *IMC '07*, pp. 29–42, ACM, 2007.

[151] A. Mizrak, Y. Cheng, V. Kumar, and S. Savage, "Structured superpeers: leveraging heterogeneity to provide constant-time lookup," *Internet Applications. WIAPP 2003. Proceedings. The Third IEEE Workshop on*, pp. 104–111, June 2003.

[152] S. Mossa, M. Barthélémy, H. Eugene Stanley, and L. A. Nunes Amaral, "Truncation of power law behavior in "scale-free" network models due to information filtering," *Phys. Rev. Lett.*, vol. 88, p. 138701, Mar 2002.

[153] J. Myers and M. Rose, "Post Office Protocol - Version 3." RFC 1939 (INTERNET STANDARD), May 1996. Updated by RFCs 1957, 2449, 6186.

[154] F. Nagle and L. Singh, "Can friends be trusted? exploring privacy in online social networks," *Social Network Analysis and Mining, International Conference on Advances in*, vol. 0, pp. 312–315, 2009.

[155] D. Naor, M. Naor, and J. Lotspiech, "Revocation and tracing schemes for stateless receivers," in *Advances in Cryptology — CRYPTO 2001* (J. Kilian, ed.), vol. 2139 of *Lecture Notes in Computer Science*, pp. 41–62, Springer Berlin Heidelberg, 2001.

[156] A. Narayanan, E. Shi, and B. Rubinstein, "Link prediction by de-anonymization: How we won the kaggle social network challenge," in *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pp. 1825–1834, 2011.

[157] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in *Proceedings of the 2009 30th IEEE Symposium on Security and Privacy*, SP '09, (Washington, DC, USA), pp. 173–187, IEEE Computer Society, 2009.

[158] R. Narendula, T. Papaioannou, and K. Aberer, "My3: A highly-available p2p-based online social network," in *The 2011 IEEE International Conference on Peer-to-Peer Computing (P2P)*, pp. 166–167, 2011.

[159] R. Narendula, T. Papaioannou, and K. Aberer, "Privacy-aware and highly-available osn profiles," in *Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE), 2010 19th IEEE International Workshop on*, pp. 211–216, 2010.

[160] M. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.

[161] M. Newman, "Detecting community structure in networks," *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 38, no. 2, pp. 321–330, 2004.

[162] M. Newman, "Models of the small world," *Journal of Statistical Physics*, vol. 101, pp. 819–841, 2000.

[163] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E*, vol. 74, p. 036104, Sep 2006.

[164] M. E. J. Newman, "Clustering and preferential attachment in growing networks," *Phys. Rev. E*, vol. 64, p. 025102, Jul 2001.

[165] M. E. J. Newman, "The structure of scientific collaboration networks," *Proceedings of the National Academy of Sciences*, vol. 98, no. 2, pp. 404–409, 2001.

[166] M. E. J. Newman, "Scientific collaboration networks. I. Network construction and fundamental results," *Phys. Rev. E*, vol. 64, p. 016131, Jun 2001.

[167] M. E. J. Newman, "Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality," *Phys. Rev. E*, vol. 64, p. 016132, Jun 2001.

[168] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E*, vol. 69, p. 026113, Feb 2004.

[169] M. E. J. Newman, D. J. Watts, and S. H. Strogatz, "Random graph models of social networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. Suppl 1, pp. 2566–2572, 2002.

[170] M. E. J. Newman, "The Structure and Function of Complex Networks," *SIAM Review*, vol. 45, pp. 167–256, Jan. 2003.

[171] J. Oikarinen and D. Reed, "Internet Relay Chat Protocol." RFC 1459 (Experimental), May 1993. Updated by RFCs 2810, 2811, 2812, 2813.

[172] T. Opsahl, F. Agneessens, and J. Skvoretz, "Node centrality in weighted networks: Generalizing degree and shortest paths," *Social Networks*, vol. 32, no. 3, pp. 245 – 251, 2010.

[173] W. Peng, F. Li, X. Zou, and J. Wu, "A two-stage deanonymization attack against anonymized social networks," *IEEE Transactions on Computers*, vol. 99, no. PrePrints, p. 1, 2012.

[174] I. Polakis, M. Lancini, G. Kontaxis, F. Maggi, S. Ioannidis, A. D. Keromytis, and S. Zanero, "All your face are belong to us: breaking facebook's social authentication," in *Proceedings of the 28th Annual Computer Security Applications Conference*, ACSAC '12, (New York, NY, USA), pp. 399–408, ACM, 2012.

[175] J. Postel, "Simple Mail Transfer Protocol." RFC 821 (INTERNET STANDARD), Aug. 1982. Obsoleted by RFC 2821.

[176] J. Postel and J. Reynolds, "File Transfer Protocol." RFC 959 (INTERNET STANDARD), Oct. 1985. Updated by RFCs 2228, 2640, 2773, 3659, 5797.

[177] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, "Defining and identifying communities in networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 9, pp. 2658–2663, 2004.

[178] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A scalable content-addressable network," in *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '01, (New York, NY, USA), pp. 161–172, ACM, 2001.

[179] P. Reynolds and A. Vahdat, "Efficient peer-to-peer keyword searching," in *IFIP International Federation for Information Processing*, vol. 2672 of *Lecture Notes of Computer Science*, pp. 21 – 40, Springer Berlin/Heidelberg, 2003.

[180] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, pp. 120–126, Feb. 1978.

[181] D. Rosenblum, "What anyone can know: The privacy risks of social networking sites," *IEEE Security and Privacy*, vol. 5, pp. 40–49, May 2007.

[182] M. Roth, A. Ben-David, D. Deutscher, G. Flysher, I. Horn, A. Leichtberg, N. Leiser, Y. Matias, and R. Merom, "Suggesting friends using the implicit social graph," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, (New York, NY, USA), pp. 233–242, ACM, 2010.

[183] A. I. T. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," in *Middleware '01: Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms Heidelberg*, (London, UK), pp. 329–350, Springer-Verlag, 2001.

[184] P. Saint-Andre, "Extensible Messaging and Presence Protocol (XMPP): Core." RFC 6120 (Proposed Standard), Mar. 2011.

[185] A. Sala, L. Cao, C. Wilson, R. Zablit, H. Zheng, and B. Y. Zhao, "Measurement-calibrated graph models for social network experiments," in *WWW '10*, pp. 861–870, ACM, 2010.

[186] R. Schenkel, T. Crecelius, M. Kacimi, S. Michel, T. Neumann, J. X. Parreira, and G. Weikum, "Efficient top-k querying over social-tagging networks," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, (New York, NY, USA), pp. 523–530, ACM, 2008.

[187] D. Schiöberg, S. Schmid, F. Schneider, S. Uhlig, H. Schiöberg, and A. Feldmann, "Tracing the birth of an osn: social graph and profile analysis in google+," in *Proceedings of the 3rd Annual ACM Web Science Conference*, WebSci '12, (New York, NY, USA), pp. 265–274, ACM, 2012.

[188] S. Schrittwieser, P. Fruehwirt, P. Kieseberg, M. Leithner, M. Mulazzani, M. Huber, and E. Weippl, "Guess who is texting you? evaluating the security of smartphone messaging applications," in *Network and Distributed System Security Symposium (NDSS 2012)*, 2 2012.

[189] A. Shakimov, H. Lim, L. P. Cox, and R. Caceres, "Vis-à-vis:online social networking via virtual individual servers," tech. rep., Duke University, May 2008.

[190] H. A. Simon, "On a class of skew distribution functions," *Biometrika*, vol. 42, no. 3/4, pp. pp. 425–440, 1955.

[191] A. Singh and L. Liu, "A hybrid topology architecture for p2p systems," in *Computer Communications and Networks, 2004. ICCCN 2004. Proceedings. 13th International Conference on*, pp. 475–480, 2004.

[192] K. Sripanidkulchai, B. Maggs, and H. Zhang, "Efficient content location using interest-based locality in peer-to-peer systems," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, vol. 3, pp. 2166–2176 vol.3, 2003.

[193] W. Stallings, *Cryptography and Network Security: Principles and Practice*. Prentice Hall, 5 ed., January 2010.

[194] R. Steinmetz and K. Wehrle, *Peer-to-Peer Systems and Applications (Lecture Notes in Computer Science)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005.

[195] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," in *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '01, (New York, NY, USA), pp. 149–160, ACM, 2001.

[196] D. Stutzbach and R. Rejaie, "Understanding churn in peer-to-peer networks," in *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, IMC '06, (New York, NY, USA), pp. 189–202, ACM, 2006.

[197] C. Tang, Z. Xu, and M. Mahalingam, "psearch: information retrieval in structured overlays," *SIGCOMM Comput. Commun. Rev.*, vol. 33, no. 1, pp. 89–94, 2003.

[198] J. Teng, B. Zhang, X. Li, X. Bai, and D. Xuan, "E-shadow: Lubricating social interaction using mobile phones," *IEEE Transactions on Computers*, vol. 99, no. PrePrints, p. 1, 2012.

[199] A. Tootoonchian, S. Saroiu, Y. Ganjali, and A. Wolman, "Lockr: better privacy for social networks," in *CoNEXT '09*, pp. 169–180, ACM, 2009.

[200] A. L. Traud, P. J. Mucha, and M. A. Porter, "Social structure of facebook networks," *Physica A: Statistical Mechanics and its Applications*, vol. 391, no. 16, pp. 4165 – 4180, 2012.

[201] J. Travers and S. Milgram, "An experimental study of the small world problem," *Sociometry*, vol. 32, no. 4, pp. pp. 425–443, 1969.

[202] D. Tsoumakos and N. Roussopoulos, "A comparison of Peer-to-Peer search methods," in *Proceedings of the Sixth International Workshop on the Web and Databases*, 2003.

[203] D. Tsoumakos and N. Roussopoulos, "Adaptive probabilistic search for peer-to-peer networks," in *Proceedings of the 3rd International Conference on Peer-to-Peer Computing*, P2P '03, (Washington, DC, USA), pp. 102–, IEEE Computer Society, 2003.

[204] T. Tylenda, R. Angelova, and S. Bedathur, "Towards time-aware link prediction in evolving social networks," in *Proceedings of the 3rd Workshop on Social Network Mining and Analysis*, SNA-KDD '09, (New York, NY, USA), pp. 9:1–9:10, ACM, 2009.

[205] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow, "The anatomy of the facebook social graph," *arXiv preprint arXiv:1111.4503*, 2011.

[206] A. Vázquez, "Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations," *Phys. Rev. E*, vol. 67, p. 056104, May 2003.

[207] T. Valente, "Network models of the diffusion of innovations," *Computational & Mathematical Organization Theory*, vol. 2, pp. 163–164, 1996.

[208] P.-F. Verhulst, "Notice sur la loi que la population poursuit dans son accroissement," *Correspondance mathématique et physique*, vol. 10, pp. 113–121, 1838.

[209] S. Voulgaris, A. M. Kermarrec, and L. Massoulie, "Exploiting semantic proximity in peer-to-peer content searching," in *Distributed Computing Systems, 2004. FTDCS 2004. Proceedings. 10th IEEE International Workshop on Future Trends of*, pp. 238–243, 2004.

[210] D. Wallach, "A survey of peer-to-peer security issues," in *Software Security — Theories and Systems* (M. Okada, B. Pierce, A. Scedrov, H. Tokuda, and A. Yonezawa, eds.), vol. 2609 of *Lecture Notes in Computer Science*, pp. 42–57, Springer Berlin Heidelberg, 2003.

[211] S. Wasserman and K. Faust, *Social network analysis: Methods and applications*, vol. 8. Cambridge University Press, 1994.

[212] D. Watts, *Small worlds: the dynamics of networks between order and randomness*. Princeton University Press, 1999.

[213] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," in *Letters to Nature*, vol. 393, pp. 440–442, nov 1998.

[214] M. Werner, "A Privacy-Enabled Architecture for Location-Based Services," in *MobiSec '10*, 2010.

[215] H. C. White, "Search parameters for the small world problem," *Social Forces*, vol. 49, no. 2, pp. 259–264, 1970.

[216] H. C. White, S. A. Boorman, and R. L. Breiger, "Social structure from multiple networks. i. blockmodels of roles and positions," *American Journal of Sociology*, vol. 81, no. 4, pp. pp. 730–780, 1976.

[217] C. Wilson, B. Boe, A. Sala, K. P. Puttaswamy, and B. Y. Zhao, "User interactions in social networks and their implications," in *EuroSys '09*, pp. 205–218, ACM, 2009.

[218] G. Wondracek, T. Holz, E. Kirda, and C. Kruegel, "A practical attack to de-anonymize social network users," in *Proceedings of the 2010 IEEE Symposium on Security and Privacy*, SP '10, (Washington, DC, USA), pp. 223–238, IEEE Computer Society, 2010.

[219] C. K. Wong, M. Gouda, and S. S. Lam, "Secure group communications using key graphs," in *Proceedings of the ACM SIGCOMM '98 conference on Applications, technologies, architectures, and protocols for computer communication*, SIGCOMM '98, (New York, NY, USA), pp. 68–79, ACM, 1998.

[220] F. Wu and B. Huberman, "Finding communities in linear time: a physics approach," *The European Physical Journal B - Condensed Matter and Complex Systems*, vol. 38, pp. 331–338, 2004.

[221] B. Yan and S. Gregory, "Detecting community structure in networks using edge prediction methods," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2012, no. 09, p. P09008, 2012.

[222] B. Yang and H. Garcia-Molina, "Improving search in peer-to-peer networks," in *Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on*, pp. 5–14, 2002.

[223] W.-S. Yang, J.-B. Dia, H.-C. Cheng, and H.-T. Lin, "Mining social networks for targeted advertising," in *System Sciences, 2006. HICSS '06. Proceedings of the 39th Annual Hawaii International Conference on*, vol. 6, pp. 137a–137a, 2006.

[224] S. Ye, J. Lang, and F. Wu, "Crawling online social graphs," in *Web Conference (APWEB), 2010 12th International Asia-Pacific*, pp. 236–242, 2010.

[225] Z. Yu, L. Feng, X. Bin, G. Kening, and Y. Ge, "Using non-topological node attributes to improve results of link prediction in social networks," *Web Information Systems and Applications Conference*, vol. 0, pp. 141–146, 2012.

[226] F. Yuan, J. Liu, C. Yin, S. Liang, and N. Shen, "A novel search algorithm utilizing high degree nodes," in *Communications and Networking in China, 2007. CHINACOM '07*, pp. 44–48, 2007.

[227] W. W. Zachary, "An information flow model for conflict and fission in small groups," *J. Anthrop. Res.*, vol. 33, pp. 452–473, 1977.

[228] J. Zhang and M. S. Ackerman, "Searching for expertise in social networks: a simulation of potential strategies," in *GROUP '05*, pp. 71–80, ACM, 2005.

[229] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph, "Tapestry: a fault-tolerant wide-area application infrastructure," *SIGCOMM Comput. Commun. Rev.*, vol. 32, pp. 81–81, Jan. 2002.

[230] S. Zhao, D. Stutzbach, and R. Rejaie, "Characterizing files in the modern gnutella network: A measurement study," in *In Proceedings of SPIE/ACM Multimedia Computing and Networking*, vol. 6071, 2006.

[231] E. Zheleva and L. Getoor, "To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles," in *Proceedings of the 18th international conference on World wide web*, WWW '09, (New York, NY, USA), pp. 531–540, ACM, 2009.

[232] Y. Zhu, H. Wang, and Y. Hu, "A super-peer based lookup in structured peer-to-peer systems," *Proceedings of the 16th International Conference on Parallel and Distributed Computing Systems (PDCS)*, 2003.

Internetquellen

[233] Alexa, "Alexa the web informaion company," 2013. `http://www.alexa.com/topsites/`, letzter Abruf: 30.04.2013.

[234] Android, "Android," 2013. `http://www.android.com/`, letzter Abruf: 30.07.2013.

[235] M. Aspan, "How Sticky Is Membership on Facebook? Just Try Breaking Free." New York Times, Feb 2008. `http://www.nytimes.com/2008/02/11/technology/11facebook.html`, letzter Abruf: 07.03.2013.

[236] Care2, "Care2," 2013. `http://www.care2.com`, letzter Abruf: 08.03.2013.

[237] B. Cohen, "The bittorrent protocol specification," 2008. `http://www.bittorrent.org/beps/bep_0003.html`, letzter Abruf: 17.05.2013.

[238] N. Cohen, "The breakfast meeting: Grilling for james murdoch, and facebook tops 900 million users." Online (New York Times), 2012. `http://online.wsj.com/article/SB10000872396390443635404578036164027386112.html`, letzter Abruf: 23.01.2013.

[239] W. W. Cohen, "Enron email dataset." Online, 2004. `http://www-2.cs.cmu.edu/~enron/`, letzter Abruf: 21.01.2013.

[240] Couchsurfing, "Couchsurfing," 2013. https://www.couchsurfing.org/, letzter Abruf: 08.03.2013.

[241] Delicious, "Discover, remember, and showcase your passions from around the web.," 2013. https://delicious.com/, letzter Abruf: 30.04.2013.

[242] Diaspora, "Share what you want, with whom you want.," September 2010. https://joindiaspora.com/, letzter Abruf: 10.06.2013.

[243] DPA, "Nach facebook-panne: Tausend gäste kommen uneingeladen zu geburtstagsparty." Online, 2012. http://www.spiegel.de/panorama/gesellschaft/nach-facebook-panne-tausend-gaeste-kommen-uneingeladen-zu-geburtstagsparty-a-766556.html, letzter Abruf: 24.05.2013.

[244] Dropbox, "Dropbox," 2013. https://www.dropbox.com/, letzter Abruf: 03.04.2013.

[245] EC2, "Amazon web services," 2013. http://aws.amazon.com/ec2/, letzter Abruf: 27.03.2013.

[246] Facebook, "Facebook," 2013. https://www.facebook.com/, letzter Abruf: 15.04.2013.

[247] Facebook, "Facebook reports first quarter 2013 results," 2013. http://investor.fb.com/releasedetail.cfm?ReleaseID=761090, letzter Abruf: 10.06.2013.

[248] Facebook, "Open graph protocol," Mai 2010. http://developers.facebook.com/docs/opengraph, letzter Abruf: 28.03.2013.

[249] flickr, "flickr," 2013. http://www.flickr.com/, letzter Abruf: 04.04.2013.

[250] T. G. D. Forum, "The annotated gnutella protocol specification v0.4," 2003. http://rfc-gnutella.sourceforge.net/developer/stable/index.html, letzter Abruf: 17.05.2013.

[251] G. A. Fowler, "Facebook: One billion and counting." Online (The Wall Street Journal), 2012. http://online.wsj.com/article/SB10000872396390443635404578036164027386112.html, letzter Abruf: 23.01.2013.

[252] B. Gellman and L. Poitras, "Documents: U.s. mining data from 9 leading internet firms; companies deny knowledge," 2013. http://www.washingtonpost.com/investigations/us-intelligence-mining-data-from-nine-us-internet-companies-in-broad-secret-program/2013/06/06/3a0c0da8-cebf-11e2-8845-d970ccb04497_story.html, letzter Abruf: 17.05.2013.

[253] GMX, "Gmx," 2013. http://www.gmx.net/, letzter Abruf: 17.05.2013.

[254] Gnutella2, "Gnutella2," 2007. http://g2.trillinux.org/, letzter Abruf: 30.04.2013.

[255] Google, "Google+," 2013. https://plus.google.com, letzter Abruf: 08.03.2013.

[256] Google, "Google," 2013. https://www.google.com/, letzter Abruf: 30.04.2013.

[257] Handelsblatt, "Aigner fordert mehr datenschutz von google und facebook," 2012. http://www.handelsblatt.com/politik/deutschland/datenschutzverordnung-aigner-fordert-mehr-datenschutz-von-google-und-facebook/7157094.html, letzter Abruf: 17.05.2013.

[258] J. U. Henrikson, "The growth of social media: An infographic." Online, 2011. http://www.searchenginejournal.com/the-growth-of-social-media-an-infographic/32788/, letzter Abruf: 04.01.2013.

[259] ility.de, "Vegas mobile," 2013. https://play.google.com/store/apps/details?id=de.lmu.ifi.mobile.vegas, letzter Abruf: 30.07.2013.

[260] IMDb, "Imdb," 2013. http://www.imdb.com/, letzter Abruf: 02.06.2013.

[261] T. Klingberg and R. Manfredi, "RFC Draft of Gnutella v0. 6," 2002. http://rfcgnutella.sourceforge.net/src/rfc-0_6-draft.html, letzter Abruf: 30.04.2013.

[262] Last.fm, "Last.fm," 2013. http://www.last.fm, letzter Abruf: 08.03.2013.

[263] LinkedIn, "Linkedin," 2013. https://www.linkedin.com, letzter Abruf: 08.03.2013.

[264] F. Lüpke-Narberhaus, "Polizei und facebook-partys: Angst vor dem klick." Online, 2012. http://www.spiegel.de/schulspiegel/leben/facebook-partys-polizei-geht-gegen-veranstalter-und-teilnehmer-vor-a-849393.html, letzter Abruf: 24.05.2013.

[265] C. Mackenzie, "Heartbreak for family at funeral of 15-year-old who killed herself because of bullying - and are still bombarded with hate messages on facebook tribute page." Online, 2012. http://www.dailymail.co.uk/news/article-2083504/Amanda-Cummings-suicide-Hate-messages-Facebook-tribute-page.html, letzter Abruf: 24.05.2013.

[266] P. McDonald, "Growing beyond regional networks." Online (The Facebook Blog), 2009. http://blog.facebook.com/blog.php?post=91242982130, letzter Abruf: 04.03.2013.

[267] Napster, "Napster." http://opennap.sourceforge.net/napster.txt, Apr. 2000. Online Access: 2011-11-01 21:54.

[268] S. Networks, "The FastTrack Protocol (Version 1.19)." Online, July 2007. http://cvs.berlios.de/cgi-bin/viewcvs.cgi/gift-fasttrack/giFT-FastTrack/PROTOCOL?rev=HEAD.

[269] process one, "ejabberd 3," 2013. http://www.process-one.net/en/ejabberd/, letzter Abruf: 30.07.2013.

[270] S3, "Amazon simple storage service (amazon s3)," 2013. http://aws.amazon.com/s3/, letzter Abruf: 03.04.2013.

[271] M. Schorn, "Datenschützer: Facebook muss mehr privatsphäre garantieren," 2012. http://www.hna.de/nachrichten/netzwelt/datenschuetzer-fordert-facebook-muss-mehr-privatsphaere-garantieren-2356628.html, letzter Abruf: 07.06.2013.

[272] J. Shetty and J. Adibi, "The enron email dataset database schema and brief statistical report," 2004. http://www.isi.edu/~adibi/Enron/Enron.htm, letzter Abruf: 21.10.2013.

[273] Skype, "Skype," 2013. http://skype.com/, letzter Abruf: 17.05.2013.

[274] C. Stoecker, "Aufruf in konstanz: Facebook-party könnte 200.000 euro kosten." Online, 2012. http://www.spiegel.de/netzwelt/netzpolitik/facebook-party-in-konstanz-200-000-euro-kosten-fuer-den-organisator-a-844288.html, letzter Abruf: 24.05.2013.

[275] B. Taylor, "The next evolution of facebook platform," April 2010. http://developers.facebook.com/blog/post/377, letzter Abruf: 08.03.2013.

[276] Twitter, "Twitter," 2013. https://www.twitter.com, letzter Abruf: 22.05.2013.

[277] VegasMobile, "Vegasmobile," 2013. http://vegasmobile.de/, letzter Abruf: 30.06.2013.

[278] vsftpd, "vsftpd," 2013. https://security.appspot.com/vsftpd.html, letzter Abruf: 30.04.2013.

[279] C. Wade, "Wpp," 2013. http://www.wpp.com/wpp/press/2013/jun/06/twitter-and-wpp-announce-global-strategic-partnership/, letzter Abruf:07.06.2013.

[280] WEB.DE, "Web.de," 2013. http://web.de/, letzter Abruf: 17.05.2013.

[281] Wikipedia, "List of social networking websites," 2013. http://en.wikipedia.org/wiki/List_of_social_networking_websites, letzter Abruf: 08.03.2013.

[282] Xing, "Xing," 2013. https://www.xing.com, letzter Abruf: 08.03.2013.

[283] Yahoo, "Yahoo," 2013. http://www.yahoo.com//, letzter Abruf: 30.04.2013.

[284] Youtube, "Youtube," 2013. http://www.youtube.com/, letzter Abruf: 08.03.2013.

[285] Zattoo, "Zattoo," 2013. http://zattoo.com/, letzter Abruf: 17.05.2013.

Eigene Publikationen

[286] M. Duchon, M. Dürr, and K. Wiesner, "Kollaboratives Parkplatzmanagement: Ein Community basierter Ansatz," in *8. GI/KuVS-Fachgespräch "Ortsbezogene Anwendungen und Dienste"*, Logos Berlin, 2011.

[287] M. Duchon, J. Köpke, M. Dürr, C. Schindhelm, and F. Gschwandtner, "Pervasive Ad hoc Location Sharing To Enhance Dynamic Group Tours," in *The First International Conference on Advances in Information Mining and Management (IMMM)*, pp. 85–90, 2011.

[288] M. Duchon, D. Sommer, and M. Dürr, "Evaluation of Dynamic Transfer Nodes for Distributed Cooperative On-Demand Transportation," in *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, pp. 1–5, Sept. 2011.

[289] M. Dürr, M. Duchon, K. Wiesner, and A. Sedlmeier, "Distributed Group and Rights Management for Mobile Ad Hoc Networks," in *Wireless and Mobile Networking Conference (WMNC), 2011 4th Joint IFIP*, pp. 1 –8, Oct. 2011.

[290] M. Dürr, F. Gschwandtner, C.-K. Schindhelm, and M. Duchon, "Secure and Privacy-Preserving Cross-Layer Advertising of Location-Based Social Network Services," in *Mobile Computing, Applications, and Services* (Zhang, JoyYing and Wilkiewicz, Jarek and Nahapetian, Ani, ed.), vol. 95 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 331–337, Springer Berlin Heidelberg, 2012.

[291] M. Dürr and R. Hunt, "An Analysis of Security Threats to Mobile IPv6," *Int. J. Internet Protoc. Technol.*, vol. 3, pp. 107–118, Sept. 2008.

[292] M. Dürr, M. Maier, and F. Dorfmeister, "Vegas - A Secure and Privacy-Preserving Peer-to-Peer Online Social Network," in *2012 IIEEE Fourth International Conference on Social Computing (SocialCom)*, (Amsterdam, The Netherlands), Sept. 2012.

[293] M. Dürr, M. Maier, and K. Wiesner, "An Analysis of Query Forwarding Strategies for Secure and Privacy-Preserving Social Networks," in *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, ASONAM '12, (Washington, DC, USA), pp. 535–542, IEEE Computer Society, 2012.

[294] M. Dürr, P. Marcus, and K. Wiesner, "Secure, Privacy-Preserving, and Context-Restricted Information Sharing for Location-based Social Networks," in *The Seventh*

International Conference on Wireless and Mobile Communications (ICWMC), (Luxembourg City, Luxembourg), Jun. 2011.

[295] M. Dürr, V. Protschky, and C. Linnhoff-Popien, "Modeling Social Network Interaction Graphs," in *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, ASONAM '12, (Washington, DC, USA), pp. 660–667, IEEE Computer Society, 2012.

[296] M. Dürr, M. Werner, and M. Maier, "Re-Socializing Online Social Networks," in *Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*, pp. 786 –791, Dec. 2010.

[297] M. Dürr and K. Wiesner, "A Privacy-Preserving Social P2P Infrastructure for People-Centric Sensing," in *KiVS*, vol. 17 of *OASICS*, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2011.

[298] P. Marcus, M. Kessel, and M. Dürr, "Regelgesteuerte Auswertungsrichtlinien für LBAC-Systeme," in *8. GI/KuVS-Fachgespräch "Ortsbezogene Anwendungen und Dienste"*, Logos Berlin, 2011.

[299] M. Werner, M. Kessel, F. Gschwandtner, M. Dürr, K. Wiesner, and T. Mair, "Technologische Herausforderungen für kontextsensitive Geschäftsanwendungen," in *Smart Mobile Apps: Mit Business-Apps ins Zeitalter mobiler Geschäftsprozesse*, Springer, 2012.

[300] K. Wiesner, M. Duchon, and M. Michael Dürr, "Distributed Multi-Head Clustering for People-Centric Sensor Networks," in *SENSORCOMM 2012, The Sixth International Conference on Sensor Technologies and Applications*, pp. 53–58, 2012.

[301] K. Wiesner, M. Dürr, and M. Duchon, "Private Pooling: A Privacy-Preserving Approach for Mobile Collaborative Sensing," in *Security and Privacy in Mobile Information and Communication Systems* (Prasad, Ramjee and Farkas, Károly and Schmidt, Andreas U. and Lioy, Antonio and Russello, Giovanni and Luccio, Flaminia L., ed.), vol. 94 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 52–63, Springer Berlin Heidelberg, 2012.