

BENUTZUNG STATISTISCHER
EIGENSCHAFTEN VON RTT-MESSUNGEN ZUR
EINGLIEDERUNG VON NETZWERKEN IN
VIRTUELLE RÄUME

Bachelorarbeit
der Philosophisch-naturwissenschaftlichen Fakultät
der Universität Bern

vorgelegt von

Roger Strähl
2009

Leiter der Arbeit:
Professor Dr. Torsten Braun
Institut für Informatik und angewandte Mathematik

Inhaltsverzeichnis

1	Einleitung	3
2	GNP - Global Network Positioning	5
2.1	Die Idee	5
2.2	Der konkrete Ansatz	6
2.3	Verwandte Verfahren	9
2.3.1	Vivaldi	9
2.3.2	Lighthouses	9
2.3.3	ICS - Internet Coordinate System	10
3	Statistische Eigenschaften von RTT-Messungen	11
3.1	Idee und theoretische Herleitung	11
3.2	Anwendung auf Multilateration	12
3.3	Vergleich mit GNP	13
3.4	Optimierungsvorschlag für GNP	14
4	Evaluation	15
4.1	Auswertungsmethode	15
4.2	Versuchsaufbau	16
4.3	Ergebnisse	16
5	Fazit	19

Dank

Der Dank gebührt an dieser Stelle Dragan Milic, der diese Arbeit während ihrer Entstehung betreut hat. Insbesondere mit seiner Fachkompetenz und seiner Geduld war er mir persönlich eine grosse Hilfe. Danke!

Abstract

Für die Vorhersage von Netzwerkdistanzen existieren verschiedene Ansätze, die auf der Repräsentation von Netzwerken in virtuellen Räumen basieren. Oftmals werden zur Eingliederung gemessene Paketumlaufzeiten (Round Trip Times, RTT) benutzt, ohne dabei jedoch die wichtigsten statistischen Eigenschaften mehrerer Messungen in Betracht zu ziehen.

Diese Arbeit greift bereits existierende Überlegungen zur Berücksichtigung von Mittelwert und Varianz einer Serie von RTT-Messungen auf. Mit dem Ziel, vorhandene Ansätze wie Global Network Positioning (GNP) in ihrer Vorhersagegenauigkeit verbessern zu können, wird eine neue statistische Fehlerfunktion hergeleitet. Anhand einer empirischen Evaluation wird diese den aus GNP bekannten Fehlerfunktionen gegenüber gestellt.

Die Auswertung führt schliesslich zum Ergebnis, dass der Einbezug von statistischen Eigenschaften mittels der in der Arbeit vorgestellten Variante nicht zu einer Verbesserung der Vorhersagegenauigkeit von Netzwerkdistanzen führt.

Kapitel 1

Einleitung

Hat man die Wahl zwischen verschiedenen Anbietern eines identischen Dienstes, wählt man idealerweise denjenigen, der die Anfrage in der kürzesten Zeitspanne zu erfüllen vermag. Im Falle eines Netzwerkes aus Computern, welche wir im Folgenden auch Hosts nennen, trägt die Paketumlaufzeit (Round Trip Time, RTT) ihren wesentlichen Teil zu dieser Zeitspanne bei. Weiter wird auch die Kapazität von TCP-Verbindungen durch die RTT beschränkt, wie Floyd und Fall gezeigt haben[2]. Daher ist es sinnvoll, eine Anfrage an denjenigen Host zu senden, zu dem man den kürzesten Distanz aufweist, wobei die Distanz hier anhand der RTT definiert wird.

Die Messung von Distanzen jeweils dann auszuführen, wenn gerade eine Entscheidung getroffen werden muss, resultiert in unverhältnismässig grossem Aufwand. Effizienter ist es, die benötigten Daten bereits im Vorfeld zu beschaffen und sich während einer gewissen Zeitspanne darauf zu verlassen. Wird dies jedoch mit dem Anspruch erledigt, sämtliche Distanzen zu kennen, treten nach wie vor zwei schwerwiegende Nachteile auf: In einem Netzwerk mit n Rechnern resultiert die Messung der Distanzen sämtlicher Rechnerpaare im Bereich von $O(n^2)$. In selbem Rahmen bewegt sich die Speicherung der gemessenen Daten. In grösseren Netzwerken ist es deshalb nicht vernünftig, sämtliche Distanzen zu messen und zu speichern.

Um diese Probleme in den Griff zu bekommen und wenigstens eine ungefähre Vorhersage von Distanzen zu ermöglichen, wurden in den vergangenen Jahren verschiedenste Ansätze präsentiert. Die vielversprechendsten Vorschläge basieren darauf, Netzwerke in virtuelle, meist euklidische Räume einzugliedern. Dadurch, dass Distanzen nur noch zu ausgewählten Rechnern gemessen werden, können die benötigten Daten in vertretbarem linearem Aufwand beschafft werden. Die Speicherung der Daten geschieht durch die Zuweisung von Koordinaten an jeden Host des Netzwerkes, woraus schliesslich durch die Berechnung der (euklidischen) Distanz die realen RTT-Distanzen vorhergesagt werden können. Dadurch reduziert sich auch das benötigte Volumen zur Speicherung sämtlicher Daten auf ein lineares Mass.

Die vorliegende Arbeit beschäftigt sich mit der Auswertung potentieller Verbesserungen, welche aus Überlegungen zu den statistischen Eigenschaften einer RTT-Messung resultieren. Unter zu treffenden Annahmen kann davon ausgegangen werden, dass mehrmaliges Messen einer RTT eine Normalverteilung approximiert. Damit lässt sich eine multivariate Wahrscheinlichkeitsfunktion für die Position eines Rechners im virtuellen Raum herleiten. Der Vergleich mit vorhandenen Ansätzen weist auf eine mögliche Optimierung der bestehenden Multilaterationsverfahren hin, welche es schliesslich zu evaluieren gilt.

In Kapitel 2 wird der konkrete Ansatz Global Network Positioning (GNP)[1] vorgestellt. Eine Zusammenfassung der wesentlichsten mit GNP verwandten Verfahren folgt in Kapitel 2.3. Kapitel 3 legt die Annahmen für diese Arbeit fest und zeigt die statistischen Eigenschaften von RTT-Messungen auf. Unter den getroffenen Annahmen wird der Ansatz entwickelt, wie die besagten Eigenschaften für die Eingliederung von Hosts in virtuelle Räume benutzt werden können. Aus dem Vergleich mit den von GNP benutzten Mitteln wird ein konkreter Vorschlag zur Verbesserung vorhandener Verfahren abgeleitet. In Kapitel 4 werden die ausgearbeiteten Vorschläge schliesslich den bestehenden Ansätzen in einer empirischen Evaluation gegenüber gestellt.

Kapitel 2

GNP - Global Network Positioning

In Jahr 2002 veröffentlichten T.S. Eugene Ng und Hui Zhang von der Carnegie Mellon University in Pittsburgh in einer Arbeit erstmals ihre Idee von Global Network Positioning (GNP)[1]. Der Vergleich mit bereits vorhandenen Ansätzen versprach eine wesentliche Verbesserung bezüglich der Verlässlichkeit der Vorhersagen von Netzwerkdistanzen. Im Folgenden wird zusammenfassend erklärt, wie GNP funktioniert.

2.1 Die Idee

Um sämtliche Distanzen in einem Netzwerk sowohl zu messen als auch zu speichern, ist wie erwähnt ein grosser Aufwand nötig. Diesen Aufwand versucht GNP mit seinen Ansätzen zu reduzieren. Die Reduktion an benötigtem Speicherplatz gelingt GNP dadurch, dass das Netzwerk in einen euklidischen Raum eingegliedert wird. Jeder Host im Netzwerk erhält dadurch seine eigenen Koordinaten. Diese werden so gewählt, dass die euklidische Distanz zweier Hosts im virtuellen Raum als möglichst genaue Vorhersage der tatsächlichen Distanz, also der RTT im Netzwerk dient. Zu speichern gilt es mit diesem Ansatz lediglich die Koordinaten jedes Hosts. Und weil die Dimension des euklidischen Raumes unabhängig von der Anzahl Hosts und üblicherweise zugleich bedeutend kleiner ist, wird der Bedarf an Speicherkapazität auf ein lineares Mass reduziert.

Weil die Dimension des euklidischen Raumes nicht der Anzahl Hosts im Netzwerk entspricht, ist es für die Eingliederung eines einzelnen Hosts nicht nötig, die Distanz zu sämtlichen anderen Hosts zu kennen. Die durchgeführten Messungen beschränken sich bei GNP deshalb auf die Messung der RTT zu einer fixen Menge an bereits im Raum eingegliederten Hosts. Diese werden entsprechend ihrer Aufgabe Landmarken genannt. Die Anzahl der Landmarken ist wiederum unabhängig von der Anzahl im Netzwerk vorhandenen Hosts. Damit gelingt auch für die Beschaffung der Messdaten die Reduktion auf ein lineares Mass.

Eine Schwierigkeit, die nach wie vor besteht, ist die Eingliederung an sich. Ein metrischer Raum S mit der Distanzfunktion $d : S \times S \rightarrow \mathbb{R}$ setzt nämlich die Erfüllung folgender Bedingungen voraus:

$$\begin{aligned}
 \text{Positivität :} & & d(x, y) & \geq 0 \\
 \text{Identität :} & d(x, x) = 0 & \& \quad d(x, y) = 0 \Rightarrow x = y \\
 \text{Symmetrie :} & & d(x, y) & = d(y, x) \\
 \text{Dreiecksungleichung :} & & d(x, y) & \leq d(x, z) + d(z, y)
 \end{aligned}$$

Erstens kann eine Distanz nicht negativ sein. Zweitens ist die Distanz eines Punktes zu sich selber immer gleich Null. Weiter sind zwei Punkte als identisch zu betrachten, falls ihre Distanz gleich Null ist. Drittens gilt die Symmetrie der Distanz und schliesslich muss viertens die Dreiecksungleichung erfüllt sein. Gerade die Dreiecksungleichung, also dass die Distanz zweier Punkte immer kleiner gleich der Distanz mittels eines dritten Punktes ist, kann in Computernetzwerken nicht als gegeben angesehen werden. Tatsächlich können gesendete Pakete aufgrund der Bandbreite einer Verbindung schneller als Ziel gelangen, wenn sie einen Umweg über zusätzliche Hosts benutzen. Will man in einem solchen Fall das Netzwerk in einen metrischen Raum eingliedern, müssen die Distanzen so abgeändert werden, dass sie der Dreiecksungleichung gerecht werden. Damit entsteht zwangsmässig eine Ungenauigkeit bei der Vorhersage der tatsächlichen Netzwerkdistanz. Genau diese Ungenauigkeit macht sich GNP nun für die Eingliederung zu Nutzen.

2.2 Der konkrete Ansatz

Global Network Positioning basiert auf einer hierarchischen Struktur. Aus den verfügbaren Hosts wird zuerst eine Auswahl an Landmarken getroffen. Anhand der RTT-Messungen zwischen den Landmarken werden letztere in den euklidischen Raum eingegliedert, womit jede Landmarke seine Koordinaten erhält. Mit diesen Koordinaten dienen die Landmarken nun den übrigen Hosts als Referenzpunkte für deren Eingliederung mittels Multilateration.

Dadurch, dass die Landmarken die Grundlage für die Eingliederung sämtlicher Hosts bilden, hat bereits die Auswahl der Landmarken einen grossen Einfluss auf die Qualität der schlussendlichen Vorhersagen. Ng und Zhang machen Vorschläge, wie diese Auswahl getroffen werden kann. Beispielsweise werden bei der *N-medians* genannten Methode N Landmarken so gewählt, dass der aufsummierte Abstand zu jedem nicht gewählten Host minimiert wird. Weiter ist von Bedeutung, dass in einem euklidischen Raum der Dimension D mindestens $D + 1$ Landmarken bestimmt werden, um anschliessend die Koordinaten der restlichen Hosts eindeutig berechnen zu können.

Nach der mehrmaligen Messung der Distanzen zwischen den Landmarken wird zur Eingliederung von den erhaltenen Messungen die kleinste verwendet. Um die Positionierung der Landmarken und später auch der restlichen Hosts mathematisch erklären zu können, werden nun einige Notationen eingeführt. Der benutzte metrische Raum wird S genannt. Die Koordinaten eines Hosts H in S werden mit c_H^S dargestellt. Die gemessene Distanz zwischen Host H_1

und H_2 wird notiert als $d_{H_1H_2}$, die aus den Koordinaten in S berechnete Distanz als $\hat{d}_{H_1H_2}^S$. Entsprechend stehen $d_{L_1L_2}$ und $\hat{d}_{L_1L_2}^S$ für die Distanzen zwischen den Landmarken L_1 und L_2 .

Für die Einbettung versucht man nun, die Koordinaten der Landmarken so zu wählen, dass für alle Paare von Landmarken der Fehler zwischen gemessener Distanz $d_{L_iL_j}$ und berechneter Distanz $\hat{d}_{L_iL_j}^S$ möglichst klein wird. Mathematisch gesprochen soll die Funktion

$$f(c_{L_1}^S, \dots, c_{L_N}^S) := \sum_{L_i, L_j \in \{L_1, \dots, L_N\} \mid i > j} \varepsilon(d_{L_iL_j}, \hat{d}_{L_iL_j}^S)$$

minimiert werden. Dabei ist $\varepsilon(\dots)$ eine Fehlerfunktion. Ng und Zhang benutzen in ihrer Arbeit einerseits den einfachen quadratischen Fehler

$$\varepsilon_{Sq}(d_{L_1L_2}, \hat{d}_{L_1L_2}^S) := (d_{L_1L_2} - \hat{d}_{L_1L_2}^S)^2 \quad (2.1)$$

sowie den Normierten Fehler

$$\varepsilon_{Norm}(d_{L_1L_2}, \hat{d}_{L_1L_2}^S) := \left(\frac{d_{L_1L_2} - \hat{d}_{L_1L_2}^S}{d_{L_1L_2}} \right)^2, \quad (2.2)$$

wobei letzterer die besseren Vorhersagen lieferte und schliesslich zur Evaluation von GNP verwendet wurde.

Die Minimierung von $f(c_{L_1}^S, \dots, c_{L_N}^S)$ wird üblicherweise mit Hilfe approximativer Verfahren angegangen, beispielsweise mit der Simplex Downhill Methode [3]. Mit diesem Vorgehen erhält man ein lokales Minimum der obigen Funktion, jedoch kann nicht sicher gestellt werden, dass die Lösung gleichzeitig das globale Minimum ist. Unabhängig von der gewählten Methode ist, dass für die Koordinaten der Landmarken keine eindeutige Lösung existiert: die Menge der Landmarken kann ohne Änderung der Qualität des Resultates verschoben und rotiert werden. Genauere Untersuchungen zu diesem Thema wurden bereits angestellt [4].

Sind die Landmarken im virtuellen Raum eingegliedert, kann selbiges auch mit den übrigen Hosts getan werden. Dazu müssen die Hosts Nachrichten an sämtliche Landmarken verschicken um so die jeweiligen RTT in Erfahrung zu bringen. Wiederum werden jeweils mehrere Messungen getätigt, wovon lediglich die kürzeste gemessene Distanz verwendet wird. Anhand dieser und mit den Koordinaten der Landmarken ist es dem Host möglich, mit Hilfe der Funktionsminimierung seine eigenen Koordinaten in S herzuleiten. Unter der Voraussetzung, dass erneut die zuvor gewählte Fehlerfunktion verwendet wird, gilt es nun folgende Funktion zu minimieren:

$$f(c_H^S) := \sum_{L_i \in \{L_1, \dots, L_N\}} \varepsilon(d_{HL_i}, \hat{d}_{HL_i}^S)$$

Damit unterhalten nun alle Hosts des Netzwerkes ihre eigenen Koordinaten, mit welchen ohne grossen Aufwand Vorhersagen für die RTT im Netzwerk berechnet werden können. Wie genau diese Vorhersage jedoch ist, hängt von verschiedenen Parametern ab.

Als erstens übt die Wahl des virtuellen Raumes S einen grossen Einfluss aus. Es wurden bereits von Ng und Zhang Versuche mit den Oberflächen von Sphären und Zylindern als zur

Verfügung stehende Räume experimentiert. Die Ergebnisse fielen jedoch zu Gunsten eines euklidischen Raumes aus. Für diesen wiederum ist die Anzahl der Dimensionen zu wählen. Dabei gilt es zu beachten, dass zusätzliche Dimensionen zwar zu besseren Vorhersageergebnissen führen können, jedoch auch zusätzlichen Rechenaufwand mit sich ziehen.

Weiter spielt es eine Rolle, wie viele und welche Hosts als Landmarken ausgewählt werden. An die Auswahl wird neben ihrer hohen Verfügbarkeit der Anspruch gestellt, dass die Landmarken möglichst gut im Netzwerk verteilt sind. Jedoch wurde bereits gezeigt, dass es beispielsweise nicht optimal ist, die Landmarken so zu wählen, dass die Abstände untereinander maximiert werden.

Die Funktionsminimierung ist bezüglich Rechenarbeit der aufwändigste Teil von GNP, weiter bestimmt sie auch die schlussendlichen Koordinaten, weswegen das gewählte Verfahren eine wesentliche Rolle spielt.

Schliesslich ist die Fehlerfunktion zu erwähnen. Sie bestimmt das Gewicht der Abweichung zwischen $d_{H_1H_2}$ und $\hat{d}_{H_1H_2}^S$ und hat somit einen direkten Einfluss auf die Bestimmung der Koordinaten der Hosts. Ng und Zhang schlagen in ihrer Arbeit die Benutzung der Normalisierten Fehlerfunktion vor. Wir werden in Kapitel 3.4 erläutern, wieso wir denken, dass dies keine optimale Wahl ist.

2.3 Verwandte Verfahren

t Neben GNP gibt es zahlreiche andere Ansätze zur Vorhersage von Netzwerkdistanzen. Nachfolgend werden die populärsten Methoden vorgestellt.

2.3.1 Vivaldi

Mit Vivaldi [5] schlagen die Autoren Frank Dabek et al. ein auf Koordinaten basiertes, jedoch völlig dezentrales System zur Vorhersage von Netzwerkdistanzen vor. Motiviert wird Vivaldi durch die Modellierung eines aus physikalischen Federn bestehenden Netzes. Jedem Paar von Hosts wird eine Feder zugeordnet, deren Länge die vorhergesagte Distanz im Koordinatenraum repräsentiert. Die Federlänge im Ruhezustand entspricht dabei genau der gemessenen Distanz zwischen den beiden Rechnern. Die Abweichung zwischen vorhergesagter und realer Distanz wird durch den bereits in GNP erwähnten einfachen quadratischen Fehler quantifiziert. Gleichzeitig steht dieser Fehler für die physikalische Energie, die für die Verformung der entsprechenden Feder aufgebracht werden muss.

Gesucht wird bei Vivaldi nun der Zustand des Gesamtsystems mit der kleinsten gespeicherten Energie, also mit dem kleinsten globalen Fehler. Dabei passt jeder Host seine Position selbstständig an neu verfügbare Daten an, die bei der Kommunikation mit anderen Hosts anfallen. Im Modell bewegt sich der Host ein Stück in die Richtung der von der deformierten Feder ausgehenden Kraft. Wie stark diese Bewegung ausfällt, hängt unter anderem davon ab, wie sicher sich die anderen Hosts bezüglich ihrer Koordinaten sind. Optimalerweise pendelt sich damit das gesamte System im gewünschten Zustand ein. Die Evaluation von Vivaldi hat allerdings gezeigt, dass die Koordinaten der Hosts nicht gegen einen stabilen Wert konvergieren. Diesem Nachteil tragen Cédric de Launois et al. mit ihren Vorschlägen für SVivaldi [6] Rechnung, gleichzeitig wird der Algorithmus von Vivaldi so angepasst, dass er genauere Vorhersagen liefert.

Im Vergleich zu GNP liefert Vivaldi zwar weniger zuverlässige Vorhersagen, ist hingegen wie erwähnt völlig dezentral organisiert und kann durch die ständige Anpassung der Hostkoordinaten selbstständig auf Änderungen der Netzwerkstruktur reagieren.

2.3.2 Lighthouses

In der fixen Zusammensetzung der Landmarken sehen die Autoren Marcelo Pias et al. GNPs grösste Schwäche. Diese wollen sie mit ihrem Ansatz der Lighthouses (Leuchttürme) [7] umgehen. Das Problem fest vorgegebener Landmarken ist die Abhängigkeit des ganzen Systems von der Verfügbarkeit letzterer. Und selbst wenn eine vollständige Verfügbarkeit gewährleistet werden könnte, kann sich die Wahl ungünstiger Landmarken durch ungenaue Vorhersagen netzwerkweit bemerkbar machen. Um also Flaschenhälse und einzelne Stellen des Scheiterns zu vermeiden, basiert das Konzept von Lighthouses darauf, statt einem globalen mehrere lokale Koordinatensysteme zu verwenden, wobei jeder Host als potentieller Pivotknoten (so werden hier die Landmarken genannt) dienen kann.

Die ersten Hosts, die sich im Netzwerk anmelden, bilden untereinander lediglich lokale Koordinatenbasen. Erst wenn genügend Rechner vorhanden sind, wird eine definitive Basis kon-

struiert, mit der dann die Koordinaten aller anderen Hosts hergeleitet werden. Ein einzelner Host kontaktiert im konkreten Fall einen anderen ihm bekannten Host, um eine Liste der verfügbaren Pivotknoten zu bekommen. Zu einer Auswahl dieser Pivotknoten misst der Host anschliessend die Distanz und platziert sich mittels Gram-Schmidt-Verfahren[8] in einem lokalen Koordinatensystem relativ zu den ausgewählten Pivotknoten. Für diese Pivotknoten existiert eine vorher errechnete Transitionsmatrix, mit welcher es dem Host wiederum möglich ist, von seiner lokalen Position global gültige Koordinaten abzuleiten.

Die Genauigkeit von Lighthouses bezüglich Vorhersagen ist vergleichbar mit GNP, wenn auch nicht ganz so hoch. Einen Vorteil bietet hingegen die höhere Flexibilität bei der Auswahl der für die Herleitung der Koordinaten referenzierten Pivotknoten.

2.3.3 ICS - Internet Coordinate System

Internet Coordinate System [9], kurz ICS und im Jahre 2003 von Hyuk Lim et al. entwickelt, arbeitet wie GNP ebenfalls mit Landmarken, die von den Autoren als Leuchtsignale (Beacon Nodes) bezeichnet werden. Gegenüber GNP verspricht dieser Ansatz eindeutige Koordinaten, die nicht wie beispielsweise in GNP vom benutzten Minimierungsverfahren abhängig sind. Wiederum messen Hosts die Distanz zu einer Auswahl an n Leuchtsignalen und erhalten damit einen n -dimensionalen Distanzvektor. Mittels PCA (Principal Component Analysis, zu deutsch Hauptkomponentenanalyse)[10] wird dieser Vektor in seiner Dimension reduziert. Kurz gefasst wird bei PCA der Merkmalsraum durch Hauptachsentransformation auf einen Merkmalsraum niederer Dimension gebrochen, indem die Eigenvektoren mit dem grössten Eigenwert, also dem grössten Informationsgehalt, als Basis des neuen Raumes dienen.

Zu Beginn werden die Distanzen zwischen den Leuchtsignalen gemessen und daraus mit dem oben beschriebenen Verfahren eine Transformationsmatrix berechnet. Kennt ein Host nun die Distanzen zu einer Auswahl an Leuchtsignalen, kann er anhand dieser Matrix auf einfache Weise seine definitiven Koordinaten berechnen.

Somit besteht die Berechnung lediglich aus Operationen der linearen Algebra und ist daher weniger aufwändig als das von GNP benutzte Simplex-Downhill-Verfahren. Weiter sind die Koordinaten eines Hosts keinen zufälligen Einflüssen ausgesetzt, sondern eindeutig festgelegt. Da nicht zu allen Leuchtsignalen die Distanz gemessen wird, kann das bereits erwähnte Flaschenhals-Problem umgangen werden. Vorhersagetechnisch liefert ICS ähnliche Genauigkeit wie GNP. Allerdings resultieren unter gewissen Voraussetzungen, beispielsweise bei einer hierarchisch aufgebauten Netzwerkstruktur, deutlich bessere Ergebnisse.

Kapitel 3

Statistische Eigenschaften von RTT-Messungen

Die in diesem Kapitel vorgestellten Überlegungen und mathematischen Formeln stammen aus der Hand von Dragan Milic, Mitglied der Forschungsgruppe *Rechnernetze und verteilte Systeme* der Universität Bern.

3.1 Idee und theoretische Herleitung

Den folgenden Überlegungen zu Grunde liegt die Betrachtung des Weges, den ein Datenpaket auf seiner Reise vom Sender zum Empfänger zurücklegt. Üblicherweise passiert das Paket dabei mehrere Router, wobei jeder Router als Zwischenstation die Auslieferung um eine kurze Dauer verzögert. Diese Verzögerung hängt von verschiedenen Faktoren ab, beispielsweise der Auslastung des Routers, seiner Konfiguration oder seinem Scheduling-Algorithmus.

Selbst wenn gesendete Pakete also die gleichen Router passieren, treffen sie mit unterschiedlichen Verzögerungen beim Ziel an. Dabei trägt jedes Paket die Summe der von den einzelnen Routern verursachten Verzögerungen mit sich. Unter der Annahme, dass diese Verzögerungen voneinander unabhängig sind, können wir letztere nun als unabhängige Zufallsvariable X_i betrachten. Die Limitierung der Warteschlangengrösse eines Routers führt dazu, dass X_i eine endliche Varianz besitzt. Somit ist die resultierende Verzögerung eines Paketes eine Summe von unabhängigen und endlichen Zufallsvariablen X_i . Unter der zusätzlichen Annahme, dass die X_i identisch verteilt sind, kann schliesslich der zentrale Grenzwertsatz angewendet werden. Dieser besagt, dass die Summe von n unabhängigen identisch verteilten Zufallsvariablen mit wachsendem n eine Zufallsvariable mit Normalverteilung approximiert. In diesem Fall ist n die Anzahl involvierter Router.

Die Wahrscheinlichkeitsdichte einer Normalverteilung mit Mittelwert μ und Varianz σ ist mit folgender Funktion definiert:

$$f(x) := \frac{1}{\sigma\sqrt{2\pi}} e^{-\left(\frac{(x-\mu)^2}{2\sigma^2}\right)}$$

Mit der Annahme von Unkorreliertheit zwischen den Dimensionen kann diese Formel auch für mehrdimensionale Räume benutzt werden, x , μ und σ sind dabei als Vektoren zu verstehen.

3.2 Anwendung auf Multilateration

Die Betrachtung der Verzögerung als Zufallsvariable wollen wir nun für die beispielsweise in GNP benutzte Multilateration zur Eingliederung in virtuelle Räume verwenden. Von einem Host H_1 messe man mehrmals die RTT zu einem zweiten Host H_2 . Mit Mittelwert μ und Varianz σ der gemessenen Daten bekommt man durch die oben genannte Dichte eine Funktion für die Wahrscheinlichkeit der tatsächlichen Distanz zwischen H_1 und H_2 . Bei der Messung der RTT zu m Landmarken L_i entsteht für die Distanz zu jeder Landmarke eine Dichtefunktion. Sind die Koordinaten der Landmarken im Raum S bekannt, ergibt sich für die gesuchte Position des Hosts H_1 eine Wahrscheinlichkeitsfunktion, indem die einzelnen Funktionen multipliziert werden. Folgende Funktion resultiert, ein Beispiel dazu ist in Abbildung 3.1 ersichtlich:

$$F(c_H^S) := \prod_{j=1}^m \frac{1}{\sigma_j \sqrt{2\pi}} e^{-\left(\frac{(\hat{d}_{H_1 L_j} - \mu_j)^2}{2\sigma_j^2}\right)}$$

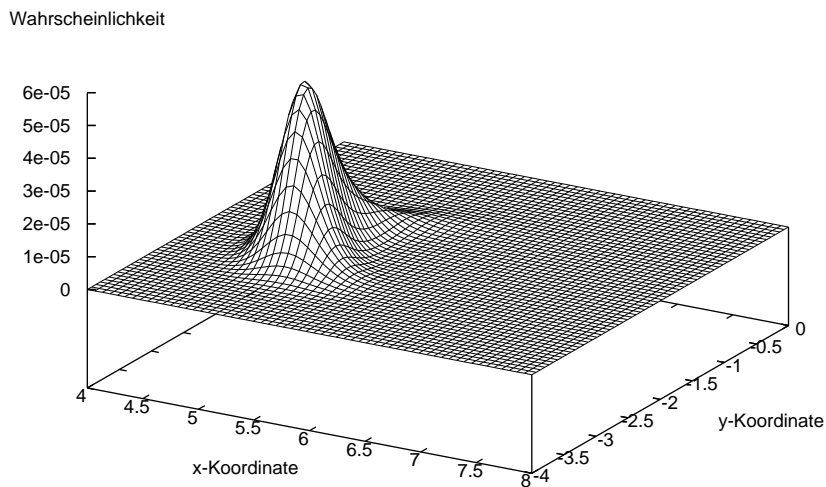


Abbildung 3.1: Beispiel einer zweidimensionalen Wahrscheinlichkeitsfunktion für die Koordinaten eines Hosts anhand von Messungen zu drei Landmarken

Um die wahrscheinlichste Position von H_1 zu finden, brauchen wir lediglich das Maximum der obigen Funktion zu bestimmen. Dies mit vorhandenen approximativen Verfahren erledigen zu wollen, führt uns jedoch zugleich zu folgendem Problem: Ausserhalb des Maximums fallen die Funktionswerte schnell ab und werden, gespeichert als Gleitkommazahl, auf den Wert 0

gerundet. Eine numerische Lösung des Problems wird so verunmöglicht. Um dies zu umgehen, kann die Funktion logarithmiert werden, die Position des Maximums bleibt dabei erhalten. Gemäss den Logarithmusgesetzen nehmen wir anschliessend weitere Vereinfachungen vor:

$$\begin{aligned}
F_{\ln}(c_H^S) &:= \ln \left(\prod_{j=1}^m \frac{e^{-\frac{(\hat{d}_{H_1 L_j} - \mu_j)^2}{2\sigma_j^2}}}{\sigma_j \sqrt{2\pi}} \right) \\
&= \sum_{j=1}^m \ln \left(\frac{e^{-\frac{(\hat{d}_{H_1 L_j} - \mu_j)^2}{2\sigma_j^2}}}{\sigma_j \sqrt{2\pi}} \right) \\
&= -\frac{1}{2} \sum_{j=1}^m \left(\frac{\ln(\sigma_j \sqrt{2\pi})}{2} \right) - \frac{1}{2} \sum_{j=1}^m \left(\frac{\hat{d}_{H_1 L_j} - \mu_j}{\sigma_j} \right)^2
\end{aligned}$$

Anstatt das Maximum dieser Funktion kann man auch nach dem Minimum der negierten Funktion suchen:

$$F_{\ln}^{neg}(c_H^S) := \frac{1}{2} \sum_{j=1}^m \left(\frac{\ln(\sigma_j \sqrt{2\pi})}{2} \right) + \frac{1}{2} \sum_{j=1}^m \left(\frac{\hat{d}_{H_1 L_j} - \mu_j}{\sigma_j} \right)^2$$

Da wir nun lediglich am Minimum interessiert sind, kann der konstante erste Term ersatzlos entfernt werden. Weiter wird das Resultat durch die Multiplikation mit einer positiven Konstanten nicht verfremdet. Wir erhalten nach diesen letzten Umformungen nun die endgültige Formel:

$$F^{final}(c_H^S) := \sum_{j=1}^m \left(\frac{\hat{d}_{H_1 L_j} - \mu_j}{\sigma_j} \right)^2 \tag{3.1}$$

3.3 Vergleich mit GNP

Die soeben hergeleitete Formel lässt sich nun für die Eingliederung der Hosts benutzen. Die abgeleitete statistische Fehlerfunktion lautet wie folgt:

$$\varepsilon_{Stat}(d_{H_i H_j}, \hat{d}_{H_i H_j}^S) := \left(\frac{\hat{d}_{H_i H_j}^S - \mu_{ij}}{\sigma_{ij}} \right)^2 \tag{3.2}$$

Betrachten wir die Fehlerfunktionen (2.1) und (2.2) aus Kapitel 2.2, fällt die Ähnlichkeit auf. Wir wollen die Unterschiede nun genauer betrachten.

$$\varepsilon_{Sq}(d_{H_i H_j}, \hat{d}_{H_i H_j}^S) := (\hat{d}_{H_i H_j}^S - d_{H_i H_j})^2 \tag{3.3}$$

Beim einfachen quadratischen Fehler (3.3) wird an Stelle des Mittelwertes μ die minimale gemessene RTT verwendet. Dies führt dazu, dass die tatsächliche Netzwerkdistanz systematisch unterschätzt wird. Selbes gilt für die Varianz σ : Diese wird vollends ignoriert und damit ebenfalls unterschätzt.

$$\varepsilon_{Norm}(d_{H_i H_j}, \hat{d}_{H_i H_j}^S) := \left(\frac{\hat{d}_{H_i H_j}^S - d_{H_i H_j}}{d_{H_i H_j}} \right)^2, \quad (3.4)$$

Im Gegensatz dazu wird beim normierten Fehler (3.4) für die Varianz wiederum die kleinste RTT verwendet. Da die Varianz mit der Steigerung der RTT grundsätzlich ebenfalls zunimmt, stellt dies zwar eine Verbesserung gegenüber dem einfachen quadratischen Fehler dar, führt aber gleichzeitig zu einer Überschätzung der tatsächlichen Varianz.

3.4 Optimierungsvorschlag für GNP

Der in vorigem Kapitel gemachte Vergleich der drei bekannten Fehlerfunktionen lässt darauf schließen, dass bei GNP wesentliche Messdaten vernachlässigt werden. Wir erwarten deshalb, dass die Verwendung der neuen statistischen Fehlerfunktion zu einer Verbesserung der Vorhersage von Netzwerkdistanzen führen wird. Dies wollen wir anhand einer empirischen Evaluation überprüfen.

Kapitel 4

Evaluation

Das Ziel der Evaluation ist es, herauszufinden, ob die Anwendung der in Kapitel 3.1 hergeleiteten theoretischen Überlegungen tatsächlich zu einer verbesserten Vorhersagemöglichkeit von Netzwerkdistancen führen. Insbesondere stellt sich bei GNP die Frage, wie sich die Anpassung der Fehlerfunktion für die Eingliederung der Hosts anhand der Koordinaten der Landmarken auf die Vorhersagen im gesamten Netzwerk auswirkt.

4.1 Auswertungsmethode

In einem ersten Schritt betrachten wir die Abweichung der Vorhersagen von den tatsächlichen Netzwerkdistancen. Dazu werden nach der Berechnung der Koordinaten von Landmarken und Hosts sämtliche Distanzen im virtuellen Raum mit den dazugehörigen gemessenen Distanzen verglichen. Dies tun wir unter der Annahme, dass die in Kapitel 3.1 getroffenen statistischen Annahmen über Netzwerkverzögerungen zutreffen. Für die Quantifizierung der Abweichungen wenden wir deshalb folgende Funktion an:

$$\varepsilon_{H_i H_j} = \left| \frac{d_{ij} - \mu_{ij}}{\sigma_{ij}} \right| \quad (4.1)$$

Gemessen wird also die mittels Varianz normierte Differenz zwischen Vorhersage und Mittelwert der Messungen. Je kleiner die Differenz, desto genauer ist die Vorhersage. Die aus den drei Fehlerfunktionen resultierenden Ergebnisse können damit direkt verglichen werden.

Weiter wollen wir eine von allen Annahmen unabhängige Auswertung durchführen. Dazu erinnern wir uns an die Motivation, aus einer Auswahl von Servern denjenigen zu wählen, der mit kürzester Verzögerung auf die gewünschte Anforderung reagieren kann. Folglich spielt die Genauigkeit der vorhergesagten Distanzen keine wesentliche Rolle, solange die der Realität entsprechende Reihenfolge eingehalten wird. Dies wollen wir mit der so genannten Ranggenauigkeit überprüfen.

Dabei wird für jeden Host die Reihenfolge der nach Distanz sortierten restlichen Hosts betrachtet. Für die n nächsten vorhergesagten Hosts wird überprüft, wie weit diese mit den n gemessenen nächsten Hosts übereinstimmen. Dabei gilt: Je höher die Übereinstimmung, desto besser sind die Vorhersageergebnisse. Dies ermöglicht wiederum den direkten Vergleich der drei Fehlerfunktionen.

4.2 Versuchsaufbau

Der Versuchsaufbau wurde so konstruiert, dass die oben gestellten Fragen möglichst ohne Einflüsse übriger Parameter beantwortet werden können. Die Implementierung wurde mittels Java[11] und Octave[12] vorgenommen.

Als Messdaten dienten insgesamt 102 zufällig ausgewählte Auszüge aus dem PlanetLab all-sites-pings Experiment[13] mit jeweils 73 - 210 Hosts. Bei diesem Experiment wurden zu verschiedenen Zeitpunkten zwischen jedem beteiligten Hostpaar jeweils zehn RTT-Messungen durchgeführt und festgehalten. Zur Verfügung gestellt wurden daraus der kleinste und der grösste Messwert sowie der Mittelwert. Damit lässt sich aus den beiden ersten Werten ein Schätzwert für die Varianz berechnen. Benutzt haben wir schliesslich den kleinsten Messwert, den Mittelwert und die geschätzte Varianz.

Bei der Festlegung des virtuellen Raumes beschränkten wir uns auf die Benutzung eines 7-dimensionalen Euklidischen Raumes, was sich in bisherigen Versuchen als vernünftige Wahl herausgestellt hatte. Variabel war die Anzahl der Landmarken: Für jeden Auszug wurden in verschiedenen Durchläufen 9, 11, 13 und 15 Landmarken benutzt. Die Auswahl selbiger geschah mittels pseudozufälliger Auswahl. Um auch diesen Zufallsfaktor abzuschwächen, wurde jeder Durchlauf wiederum zehn mal ausgeführt, wobei der Zufallsgenerator jeweils mit einem unterschiedlichen Seed initialisiert wurde. Die Kombination dieser Variationen führte für jede der drei Fehlerfunktionen zu insgesamt über 4000 Durchläufen, wovon jeder Vorhersagen generierte, die es anschliessend anhand der oben erläuterten Kriterien auszuwerten galt.

Um die absoluten Abweichungen zu quantifizieren, wurden alle Abweichungen sämtlicher Durchgänge zusammengefasst, um sie dann als eine grosse Datenmenge zu betrachten. Bei der Ranggenauigkeit berechnen wir die Werte für jeden Host in jedem Durchgang und betrachten anschliessend deren Durchschnittswerte.

4.3 Ergebnisse

Als Erstes wurden die aus dem Versuchsaufbau gewonnenen Daten mit Hilfe der in Kapitel 4.1 definierten Fehlerfunktion quantifiziert. Für jedes Hostpaar aus jedem Durchlauf erhalten wir eine Abweichung der Vorhersagen. Diese Masse an Daten lässt sich grafisch am besten mittels einer kumulativen Verteilungsfunktion darstellen. Das Resultat ist in Abbildung 4.1 sichtbar.

Die Linien stehen für die Resultate der einzelnen Fehlerfunktionen, die es zu evaluieren galt. Zu jeder Abweichung auf der horizontalen Achse zeigt die Linie, wie viele Prozent der gemessenen Fehler kleiner oder gleich dem entsprechenden Wert sind. Je höher also die Kurve, desto besser sind die Vorhersagen der Fehlerfunktion zu werten. Durch die Normierung ist die Abweichung gemäss Formel (4.1) ohne Einheit.

Wie bereits bei der ursprünglichen Version von GNP schneidet auch unter unseren Annahmen die normierte Fehlerfunktion besser ab als die einfache quadratische Fehlerfunktion. Unerwartet deutlich schlechtere Vorhersagen liefert hingegen die neue statistische Fehlerfunktion: Die gewonnenen Vorhersagen sind gar ungenauer als die der einfachen quadratischen Fehlerfunktion.

Dies bestätigt auch ein Blick auf die in Tabelle 4.1 aufgeführten Quantile. Betrachten wir

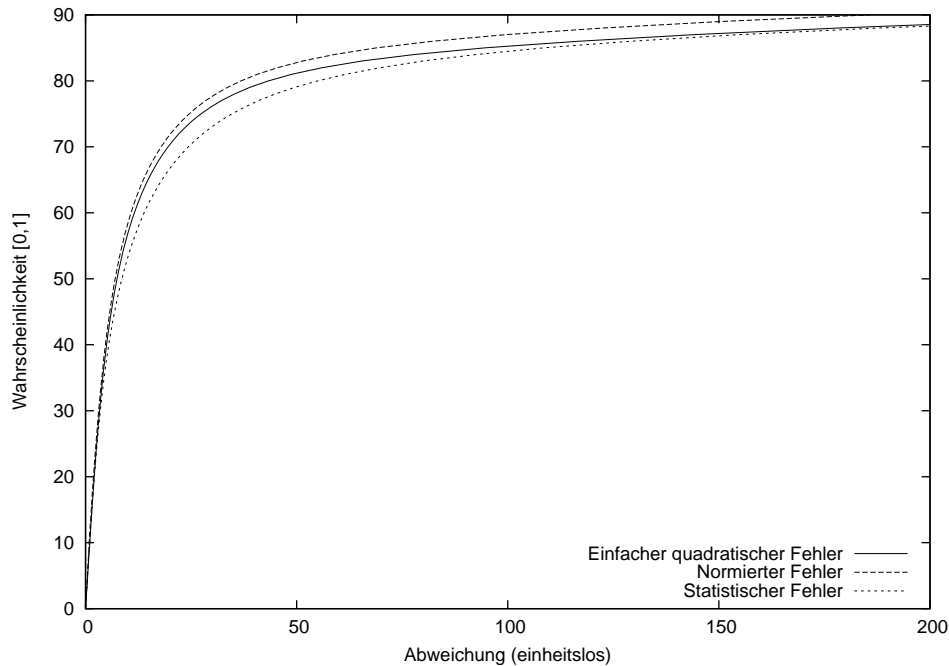


Abbildung 4.1: Kumulative Verteilungsfunktion der einheitslosen Vorhersageabweichungen gemäss Formel (4.1), beschränkt auf eine Abweichung von maximal 200.

unter allen Vorhersagen die gemäss Quantifizierung besten 80 Prozent, liegen bei der normierten Fehlerfunktion sämtliche Abweichungen unter 36.84. Die entsprechende Abweichung für die statistische Fehlerfunktion beträgt hingegen 54.88.

Quantil	80%	90%	95%
Einfach quadratisch	43.31	264.52	674.01
Normiert	36.84	185.81	510.66
Statistisch	54.88	277.29	755.01

Tabelle 4.1: Quantile der absoluten Vorhersageabweichungen

Ebenfalls aufschlussreich sind die Ergebnisse der Ranggenauigkeit. Diese überprüft wie erwähnt, wie stark die Reihenfolge der vorhergesagten Distanzen mit der Reihenfolge der tatsächlichen Distanzen übereinstimmt. In Abbildung 4.2 sind die Resultate ersichtlich. Betrachtet man für die statistische Fehlerfunktion die Vorhersage der n nächsten Hosts, liegt für $n=10$ Prozent aller verfügbaren Hosts die Übereinstimmung mit den realen Distanzen bei 54 Prozent. Für die einfache quadratische Funktion liegt der selbe Wert bei 60 Prozent, für die normalisierte Funktion mit 67 Prozent nochmals deutlich höher. Diese Reihenfolge bleibt für beliebige n erhalten. Wie schon zuvor liefert die statistische Fehlerfunktion also auch bei dieser Auswertung schlechtere Ergebnisse als die beiden anderen Fehlerfunktionen.

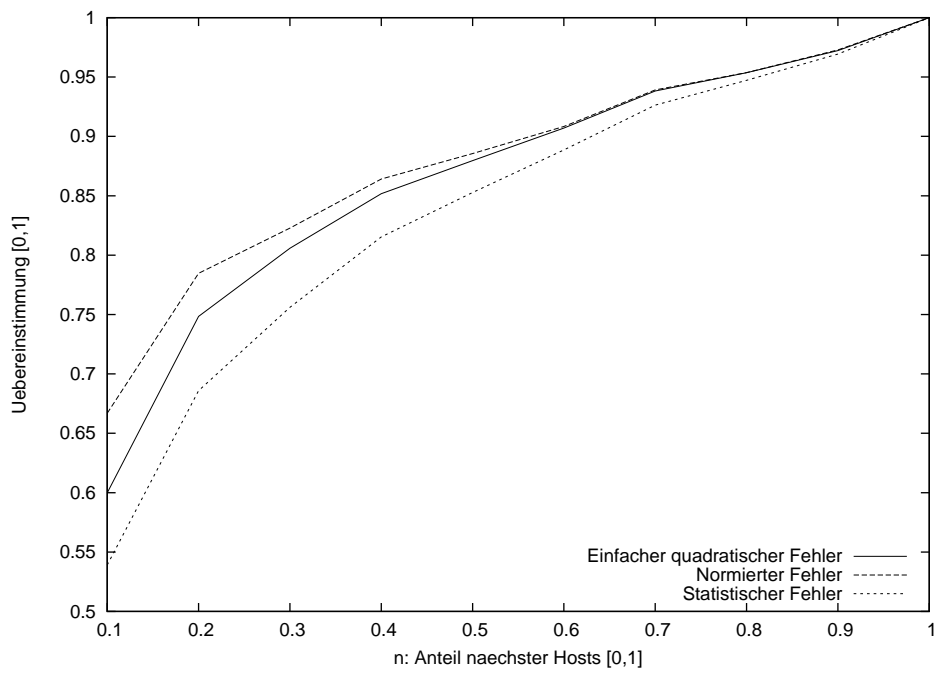


Abbildung 4.2: Ranggenauigkeit

Kapitel 5

Fazit

Die Auswertungen haben klar gezeigt, dass der Einbezug der statistischen Eigenschaften mittels der hergeleiteten Formel nicht zu einer Verbesserung der Vorhersagen von Netzwerkdistanzen führt. Vielmehr lässt sich sogar sagen, dass die neue Fehlerfunktion zu schlechteren Resultaten führt als die bereits bekannten Fehlerfunktionen.

Wir gehen davon aus, dass bereits die in Kapitel 3.1 getroffenen Annahmen über Verzögerungen in Computernetzwerken nicht zutreffend sind. Folglich ist der Zentrale Grenzwertsatz nicht anwendbar und die daraus entstehende Formel führt zu verfälschten Vorhersageergebnissen.

Nach wie vor besteht aber die Vermutung, dass GNP statistische Eigenschaften von Netzwerkdistanzen ignoriert und damit auch ein Verbesserungspotential vorhanden ist. Wir denken, dass dieses durch korrekt getroffene Annahmen ausgenutzt werden kann. Statt der Normalverteilung kann sich je nach Voraussetzungen eine andere Verteilungsfunktion ergeben, welche es für die Multilateration anzupassen gilt. Das Herausfinden der korrekten Annahmen und die mathematische Herleitung einer neuen Fehlerfunktion kann der Inhalt weiterer Arbeiten sein.

Literaturverzeichnis

- [1] **T.S.E. Ng, H. Zhang**
Predicting internet network distance with coordinates-based approaches;
IEEE Infocom02, New York / USA, June 23-27 2002
- [2] **S. Floyd and K. Fall**
Router mechanisms to support end-to-end congestion control;
IEEE/ACM Trans. Netw., vol. 12, no. 6, pp. 993-1006, 2004
- [3] **J.A. Nelder and R. Mead**
A simplex method for function minimization;
Computer Journal, vol. 7, pp. 308-313, 1965
- [4] **Dragan Milic, Torsten Braun**
Optimizing Dimensionality and Accelerating Landmark Positioning for Coordinates Based RTT Predictions;
IEEE BroadNets 2007, Raleigh, North Carolina, USA, September 10 - 14, 2007
- [5] **Frank Dabek, Russ Cox, Frans Kaashoek, Robert Morris**
Vivaldi: A Decentralized Network Coordinate System;
SIGCOMM'04, Portland, Oregon, USA, August 30 - September 03, 2004
- [6] **Cédric de Launois, Steve Uhlig, Olivier Bonaventure**
A Stable and Distributed Network Coordinate System;
Université Catholique de Louvain, December, 2004
- [7] **Marcelo Pias, Jon Crowcroft, Steve Wilbur, Tim Harris, Saleem Bhatti**
Lighthouses for Scalable Distributed Location;
Springer LNCS Peer-to-Peer Systems II, pp. 278-291, 2003
- [8] **L. Corwin and R. Szczarba**
The Structure of Vector Spaces;
CRC Press, Calculus in Vector Spaces: 2nd Edition, pp. 95-132, 1994
- [9] **Hyuk Lim, Jennifer C. Hou, Chong-Ho Choi**
Constructing Internet Coordinate System Based on Delay Measurement;
ICM'03, Miami Beach Florida / USA, October 27-29, 2003

- [10] **I. T. Jolliffe**
Principal component analysis;
Springer Series in Statistics, Principal Component Analysis, 1986

- [11] **Java**
<http://www.java.com>

- [12] **J. W. Eaton**
GNU Octave;
<http://www.gnu.org/software/octave/>

- [13] **C. Yoshikawa**
Planetlab all-sites-pings experiment;
<http://ping.eecs.uc.edu/ping/>