

# Adaptives hierarchisches Ray Tracing Verfahren zur parallelen Berechnung der Wellenausbreitung in Funknetzen

Von der Fakultät für Elektrotechnik und Informationstechnik der  
Rheinisch-Westfälischen Technischen Hochschule Aachen zur Erlangung des  
akademischen Grades eines Doktors der Ingenieurwissenschaften genehmigte Dissertation

vorgelegt von

Diplom-Ingenieur  
Thomas Frach

aus Hranice na Moravě,  
Tschechische Republik

Berichter: Universitätsprofessor Dr.-Ing. Bernhard Rembold  
Universitätsprofessor Dr. rer. nat. Michael Vorländer

Tag der mündlichen Prüfung: 3. Juli 2003

Diese Dissertation ist auf den Internetseiten der Hochschulbibliothek online verfügbar.

171 Seiten mit 83 Abbildungen und 7 Tabellen



## VORWORT

---

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Hochfrequenztechnik der RWTH Aachen.

Mein Dank gilt insbesondere Herrn Prof. Dr.-Ing. B. Rembold für die Anregungen zum Thema der Dissertation, die sehr gute, fortwährende Betreuung und Förderung der Arbeit, sowie die kritische Durchsicht des Textes. Herrn Prof. Dr. rer. nat. M. Vorländer danke ich für die freundliche Übernahme des Korreferats und die Hinweise zum Manuskript.

Mein besonderer Dank gilt Herrn Dipl.-Ing. D. Brückner und der Firma MEDAV für die Bereitstellung der Daten der Channel Sounder Meßkampagne sowie Herrn Dipl.-Phys. R. Daniel von der Forschungsgesellschaft für Angewandte Naturwissenschaften für die angenehme Zusammenarbeit und die Bereitstellung des Modells und der Meßdaten der EMVU Studie. Herren Dipl.-Ing. Ch. Oikonomopoulos und Dipl.-Ing. W. Keusgen danke ich für die Erstellung und Bereitstellung des Modells des Institutslabors.

Desweiteren möchte ich mich bei meinen Kollegen am Institut für Hochfrequenztechnik der RWTH Aachen für die stets sehr gute fachliche und freundliche Zusammenarbeit bedanken. Insbesondere Dipl.-Ing. M. Hülsbusch und Dipl.-Ing. P. Bosselmann danke ich für das gründliche Korrekturlesen der Arbeit.

Herrn Dr.-Ing. M. Steppler möchte ich für die Entwicklung des sehr ansprechenden  $\text{\LaTeX}$  Stils danken, der in dieser Arbeit Verwendung findet. Für die Entwicklung der in dieser Arbeit behandelten Algorithmen wurde freie Software auf der Basis des GNU/Linux Systems verwendet. Ich möchte mich auf diesem Weg bei allen Entwicklern freier Software für ihre Arbeit bedanken.

Meinen Eltern möchte ich für ihre langjährige Unterstützung und Förderung ganz herzlich danken. Ein ganz besonderer Dank gilt meiner Frau Eva.

Aachen, im September 2003

Thomas Frach





## KURZFASSUNG

---

Die stetig wachsende Nachfrage nach mobiler Kommunikation führt zur Entwicklung neuer zellulärer Netzwerke für die Übermittlung von Daten, Sprache und Videosequenzen. Die bereits bestehenden *Global System for Mobile Communication* (GSM) Mobilfunknetze, die einen bedeutenden wirtschaftlichen Erfolg des letzten Jahrzehnts verzeichnet haben, werden in den nächsten Jahren durch Netze nach dem *Universal Mobile Telecommunication System* (UMTS) Standard ergänzt und im Laufe der Zeit ersetzt. Andererseits beginnen die lokalen drahtlosen Datennetze nach dem IEEE 802.11 Standard die bestehenden *Local Area Network* (LAN) Datennetze über die Gebäudegrenzen hinaus zu erweitern und zu lokalen Kommunikationsinseln auszubauen. In beiden Fällen werden hohe Anforderungen an die Kommunikationsbandbreite und die Dienstgüte gestellt. Zur Erfüllung dieser Anforderungen ist neben einer effizienten Nutzung des Frequenzspektrums auch eine genaue Kenntniss der Eigenschaften des Übertragungsmediums – des Kommunikationskanals – nötig.

Neben der statistischen Kanalmodellierung, die allgemeingültige Aussagen über die Eigenschaften des Übertragungsmediums erlaubt, haben sich deterministische Kanalmodelle für die genaue Vorhersage der konkreten Ausbreitungssituation bewährt. Deterministische Kanalmodellierung ist, im Gegensatz zu den statistischen Methoden, mit einem erheblichen Rechenaufwand verbunden. Somit ist die Anwendung deterministischer Modelle nur auf kleinere, stark vereinfachte Szenarien, eingeschränkt. Deterministische Verfahren können in Verbindung mit Messungen zur Parametrierung neuer statistischer Modelle dienen, die für eine Bewertung und Optimierung neuer Kommunikationssysteme eingesetzt werden.

In dieser Arbeit wird ein paralleles und adaptives Verfahren zur deterministischen Kanalmodellierung vorgestellt. Das Verfahren basiert auf der Konstruktion der Ausbreitungswege zwischen Sender und Empfänger mit der geometrischen Strahlverfolgung unter der Berücksichtigung der Beugung an Kanten und der Streuung an rauen Oberflächen. Die Besonderheit des Verfahrens liegt in der schnellen Berechnung der Ausbreitungswege durch die Schnittpunktberechnung mit Hilfe eines modifizierten räumlichen binären Partitionierungsbaumes. Für die Konstruktion optimaler Partitionierungs bäume werden neben Heuristiken die Simulated Annealing Methode und Genetische Algorithmen eingesetzt. Die Strahlendichtenormierung mit markierten Strahlen ermöglicht eine schnelle Identifizierung neuer Ausbreitungswege, wobei die Gesamtzahl der benötigten Strahlen durch eine adaptive Anpassung der Strahlendichte an die Geometrie des Szenario minimiert wird. Durch eine Kombination der Strahlverfolgung mit der Spiegelungsmethode unter Berücksichtigung der Beugung wird die Genauigkeit der Berechnung der Ausbreitungswege erheblich gesteigert und eine genaue Analyse der frequenzselektiven Eigenschaften des Kanals ermöglicht. Die Streuung an fraktalen rauen Oberflächen wird mit der Momentenmethode berechnet. Eine erhebliche Verkürzung der Rechenzeit wird durch den Einsatz der Parallelverarbeitung erzielt. Die Implementierung des parallelen Verfahrens auf symmetrischen Multiprozessoren mit einem gemeinsamen Speicher, auf Rechnern mit verteiltem Speicher und auf hybriden Architekturen wird vorgestellt und die erreichte Beschleunigung der Simulation angegeben. Die Validierung des eingesetzten physikalischen Modells wird durch einen Vergleich mit exakten Methoden und Messungen durchgeführt. Die Arbeit wird durch verschiedene Anwendungen der Methode, unter anderem bei der Funknetzplanung der Transrapid Trasse in Shanghai und bei der Berechnung der *Elektromagnetische Verträglichkeit zur Umwelt* (EMVU) einer Radaranlage, abgeschlossen.



## ABSTRACT

---

The continuous demand for mobile communications leads to the development of new cellular networks for transmission of data, voice and video sequences. Already established GSM mobile cellular networks, a great economic success of the last decade, will be slowly upgraded to and eventually replaced by the new, third generation, cellular networks according to the UMTS standard. On the other hand, local wireless data networks according to the IEEE 802.11 standard will extend the classic LAN computer networks beyond building boundaries and lead to the creation of local communication islands. Both cases imply high requirements in respect to the communication bandwidth and quality of service. Very efficient use of the frequency spectrum and a very accurate knowledge of the communication channel properties are needed to satisfy these requirements.

Statistical communication channel models allow a general classification of the propagation environment properties. In addition to the statistical modeling, deterministic channel models provide very accurate predictions of the radiowave propagation in a scenario of a particular geometry. Deterministic channel modeling is computationally very intensive, contrary to virtually any statistical channel model, and thus applications has been limited to a set of simplified geometries of rather small extents. Together with measurements, deterministic channel models can be used for parameter extraction in new statistical channel models in the process of evaluation and optimization of future communication systems.

A new parallel and adaptive method for the deterministic communication channel modelling has been developed in the context of this work. The method is based on the construction of propagation paths between transmitters and receivers by means of a ray tracing technique together with additional sources taking edge diffraction and rough surface scattering into account. A new fast algorithm based on modified binary space partitioning trees has been deployed for the ray/object intersection test. The construction of nearly optimal binary space partitioning trees for arbitrary geometries has been realized by heuristics, the simulated annealing method and genetic algorithms. A novel ray density normalization algorithm based on tagged rays has been developed to quickly identify new propagation paths. The ray density is adapted locally to the scene geometry to minimize the total number of rays required for the spatial sampling of the geometry. The combination of the classic ray launching method with the source imaging technique allows the precise computation of the shortest propagation paths while maintaining constant local ray density and short simulation runtimes. It allows the accurate computation of the frequency selective properties of the communication channel. Source imaging over multiple arbitrary oriented diffraction edges according to the extension of the Fermat's principle has been solved with the Newton-Raphson iteration and a Simulated Annealing technique. Scattering on rough fractal surfaces has been modelled by the method of moments. Further substantial acceleration of the ray tracing solution of complex electromagnetic wave propagation problems has been achieved through the parallelization of the algorithm. Parallel implementations on symmetric multiprocessors with shared memory and on distributed memory parallel computers are presented together with a new hybrid method for distributed/shared memory architectures. Theoretical limits and achieved speedup of the computation are discussed. The physical propagation model is validated with analytical models and channel sounder measurements. Finally some applications of the deterministic model are presented, among these the radio network planning of the Transrapid track in Shanghai and an *Electromagnetic Compatibility* (EMC) computation of a radar facility.



# INHALTSVERZEICHNIS

---

<b>Kurzfassung</b>	<b>v</b>
<b>Abstract</b>	<b>vii</b>
<b>1 Einleitung</b>	<b>1</b>
<b>2 Physikalische Grundlagen</b>	<b>5</b>
2.1 Die geometrisch-optische Näherung . . . . .	5
2.2 Die Herleitung der strahlenoptischen Näherung . . . . .	6
2.3 Die polarimetrische Übertragungsfunktion . . . . .	8
2.4 Reflexion und Brechung . . . . .	10
2.5 Beugung an Kanten . . . . .	14
2.6 Streuung an rauen Oberflächen . . . . .	17
2.7 Anwendungsbereich des Modells . . . . .	21
2.8 Zusammenfassung . . . . .	22
<b>3 Algorithmische Grundlagen</b>	<b>25</b>
3.1 Monte Carlo Ray Tracing Verfahren . . . . .	25
3.2 Deterministische Ray Tracing Verfahren . . . . .	26
3.2.1 Die Strahlabschussmethode . . . . .	26
3.2.2 Die Cone Tracing Methode . . . . .	27
3.2.3 Die Spiegelungsmethode . . . . .	27
3.3 Der Ray Tracing Kernel . . . . .	28
3.4 Ray Tracing Beschleuniger . . . . .	29
3.4.1 Listenorientierte Beschleuniger . . . . .	29
3.4.2 Bounding Objects . . . . .	30
3.4.3 Gitterorientierte Beschleuniger . . . . .	30
3.4.4 Hierarchische Beschleuniger . . . . .	30
3.4.5 Zellenbasierte Beschleuniger . . . . .	31
3.5 Meta-Hierarchien . . . . .	31
3.6 Zusammenfassung . . . . .	32
<b>4 Konstruktion binärer Partitionierungsbäume</b>	<b>33</b>
4.1 Die Struktur eines binären Partitionierungsbaumes . . . . .	33
4.2 Konstruktion eines modifizierten <i>Binary Space Partitioning</i> (BSP) Baumes . . . . .	35
4.3 Schnelle Schnittpunktberechnung . . . . .	37
4.4 Strahlaufteilung und Empfängerdetektion . . . . .	41
4.5 Beschleunigung der Empfängerdetektion . . . . .	42
4.6 Der 'Punkt in Polygon' Test . . . . .	44
4.7 Darstellung von Mengen mit BSP Bäumen . . . . .	46
4.7.1 Diskrete Binärbäume . . . . .	46
4.7.2 Binäre Partitionierungsbäume in $\mathbb{R}^1$ . . . . .	48
4.7.3 Mengenoperationen mit BSP Bäumen in $\mathbb{R}^1$ . . . . .	49
4.7.4 Partitionierung des BSP Baumes mit einem Partitionierer ( <i>Splitting</i> ) . . . . .	49
4.7.5 Vereinfachung durch Mengenoperation ( <i>Simplification</i> ) . . . . .	50

4.7.6	Entfernung redundanter Knoten ( <i>Pruning</i> )	51
4.7.7	Verallgemeinerte BSP Bäume in $\mathbb{R}^n$	51
4.7.8	Multidimensionale BSP Bäume	52
4.7.9	Die <i>Splitting</i> Operation in $\mathbb{R}^n$	53
4.8	Die 'Punkt in Polygon' Operation in $\mathbb{R}^n$	55
4.9	Beschreibung der Topologie mit BSP Bäumen	56
4.10	Zusammenfassung	60
<b>5</b>	<b>Optimierung binärer Partitionierungsbäume</b>	<b>61</b>
5.1	Optimale Partitionierungsbäume	61
5.2	Definition der Metrik	62
5.3	Geometrie des Vergleichsszenarios	63
5.4	Optimierung mit Heuristiken	64
5.5	Optimierung mit der Simulated Annealing Methode	66
5.6	Genetische Algorithmen	70
5.7	Optimierung durch zufällige Abtastung	72
5.8	Vergleich der Methoden	73
5.9	Weitere Optimierungsmöglichkeiten	74
5.10	Zusammenfassung	75
<b>6</b>	<b>Adaptives Ray Tracing</b>	<b>77</b>
6.1	Das Cone Tracing Verfahren	77
6.2	Identifikation der Ausbreitungspfade	78
6.3	Kombination mit der Spiegelungsmethode	79
6.4	Räumliche Abtastung	80
6.4.1	Statistische Strahlengenerierung	80
6.4.2	Deterministische Strahlengenerierung	81
6.4.2.1	Rekursive Partitionierung Platonischer Körper	81
6.4.2.2	Minimierung der Energie auf der Kugeloberfläche	84
6.4.3	Speicherzugriffsmuster und Effizienz	86
6.5	Adaptives Cone Tracing Verfahren	88
6.6	Berechnung der Beugung	91
6.7	Spiegelung der Beugungsquellen	93
6.7.1	Formulierung des Fermatschen Prinzips	94
6.7.2	Die Newton-Raphson Iteration	95
6.7.3	Präkonditionierung	96
6.7.4	Lösung mit Simulated Annealing	97
6.8	Berechnung der Streuung	98
6.9	Zusammenfassung	100
<b>7</b>	<b>Paralleles Ray Tracing</b>	<b>101</b>
7.1	Parallele Rechnerarchitekturen	101
7.1.1	Parallelrechner mit einem gemeinsamen Speicher	101
7.1.2	Parallelrechner mit einem verteilten Speicher	103
7.2	Amdahlsches Gesetz	104
7.3	Parallelisierung auf SMP Systemen	105
7.4	Parallelisierung auf Systemen mit verteiltem Speicher	107
7.5	Parallele Optimierung	110
7.6	Zusammenfassung	111

<b>8</b>	<b>Verifikation und Anwendungsbeispiele</b>	<b>113</b>
8.1	Verifikation mit exakten Verfahren . . . . .	113
8.1.1	Spiegelung an einer leitenden Ebene . . . . .	113
8.1.2	Spiegelung an einer dielektrischen Ecke . . . . .	114
8.2	Validierung durch Messungen . . . . .	114
8.3	Simulation der Wellenausbreitung in Räumen . . . . .	116
8.4	Funknetzplanung für den Transrapid . . . . .	118
8.5	Berechnung der Streustrahlung eines Radars . . . . .	123
8.6	Statistische Eigenschaften des BSP Baumes . . . . .	126
8.7	Zusammenfassung . . . . .	127
<b>9</b>	<b>Zusammenfassung und Ausblick</b>	<b>129</b>
	<b>Abbildungsverzeichnis</b>	<b>133</b>
	<b>Tabellenverzeichnis</b>	<b>135</b>
	<b>Glossar</b>	<b>137</b>
	<b>Liste der Formelzeichen</b>	<b>145</b>
	<b>Abkürzungsverzeichnis</b>	<b>151</b>
	<b>Literaturverzeichnis</b>	<b>153</b>
	<b>Lebenslauf</b>	<b>159</b>





---

## Einleitung

Das vergangene Jahrzehnt kann zurecht als das Jahrzehnt der mobilen Kommunikation bezeichnet werden. Der explosionsartige Wachstum der Teilnehmerzahl zusammen mit einem stetigen flächendeckenden Ausbau der digitalen zellularen Netze machte die mobile Kommunikation zu einem bedeutenden Wirtschaftszweig. Die Erwartungen, die von der Industrie in die Zukunft der mobilen Kommunikation gesetzt wurden, können am Ergebnis der Versteigerung der *Universal Mobile Telecommunication System* (UMTS) Lizenzen abgeschätzt werden. Diese Lizenzen werden für die Nutzung der zugewiesenen Frequenzen und den damit verbundenen Aufbau eines Kommunikationsnetzes der dritten Generation nach dem UMTS Standard benötigt.

Obwohl der Markt für die Endgeräte nach Jahren des stetigen Wachstums gesättigt scheint, werden neue multimediale Dienste wie das *Multimedia Message System* (MMS) und der mobile Internetzugang i-mode sowie zukünftige, auf UMTS basierte Dienste für einen schnellen mobilen Internetzugang für die Verbraucher kaufentscheidend sein. Der Trend bei der Entwicklung der Endgeräte und die damit verbundene Modifikation der bestehenden *Global System for Mobile Communication* (GSM) Netze geht bereits von dem reinen verbindungsorientierten Sprachdienst mit Messaging Funktionen zu einer Kombination mit einer vollwertigen paketorientierten Datenverbindung auf der Basis von *General Packet Radio Service* (GPRS). Die Endgeräte werden in der Zukunft die Funktionalität eines *Personal Digital Assistants* (PDA) mit Multimedia-Eigenschaften und einer integrierter Telefonfunktion besitzen. Es wird erwartet, dass die mobilen Multimediadienste sowie die ortsbasierten Dienste die Sprachübertragung als Gewinnmotor ablösen werden. Voraussetzung für den kommerziellen Erfolg der Multimediadienste sind neben einem attraktiven Inhalt vor allem eine den Anforderungen entsprechende hohe Kommunikationsbandbreite sowie die Gewährleistung einer ausreichenden Dienstgüte durch das Kommunikationsnetz.

Eine weitere bedeutende Entwicklung findet im Bereich der drahtlosen lokalen Datennetze (*Wireless Local Area Network* (WLAN)) nach dem IEEE<sup>1</sup> 802.11 Standard statt. Diese lokalen Datennetze erweitern den Bereich des klassischen LAN Netzwerkes über die Gebäudegrenzen hinaus, indem sie autorisierten Rechnern mit einer entsprechenden WLAN Karte den Zugang in das interne Netzwerk ermöglichen. Insbesondere in diesem Bereich ist die Anforderung an die Kommunikationsbandbreite besonders hoch, da stets die erheblich höhere Geschwindigkeit des klassischen *Local Area Network* (LAN) von dem Benutzer unbewusst als Maßstab wahrgenommen wird. Die Entwicklung der Mehrantennen-Kommunikationssysteme nach dem *Multiple-Input-Multiple-Output* (MIMO) Prinzip wird hier zu einer deutlichen Steigerung der Kommunikationsbandbreite führen.

Sowohl bei den zukünftigen zellularen Mobilfunknetzen als auch bei den breitbandigen lokalen Datennetzen werden aufwendige Kanalkodierungsverfahren eingesetzt, um eine effiziente Nutzung der nur begrenzt verfügbaren natürlichen Ressource 'Spektrum' zu erreichen. Zusätzlich können mit Mehrantennensystemen durch eine bessere Nutzung der unterschiedlichen Ausbreitungswege in einer Umgebung mit einer starken Mehrwegeausbreitung eine erhebliche Steigerung der Übertragungsraten erreicht werden. Eine genaue

---

<sup>1</sup>*Institute of Electrical and Electronics Engineers*

Kenntnis der Kanaleigenschaften ist daher für die Analyse, Bewertung und Optimierung eines Kommunikationssystems unerlässlich.

Neben der statistischen Kanalmodellierung, die eine allgemeine Bewertung der Eigenschaften des Übertragungsmediums erlaubt, werden zunehmend deterministische Kanalmodelle für die genaue Vorhersage der konkreten Ausbreitungssituation eingesetzt. Solche deterministischen Kanalmodelle basieren meistens auf der Anwendung von geometrisch-optischen Verfahren, die im englischsprachigen Raum als *Ray Tracing* Verfahren bezeichnet werden. Im Gegensatz zu den statistischen Methoden, die auf der Auswertung von verhältnismäßig einfachen Funktionen basieren, sind deterministische Kanalmodelle mit einem erheblichen Modellier- und Rechenaufwand verbunden. Somit ist die Anwendung deterministischer Modelle bisher nur auf kleinere und stark vereinfachte Geometrien eingeschränkt. Andererseits können deterministische Verfahren gewinnbringend in Verbindung mit Messungen zu der Entwicklung und Parametrierung neuer statistischer Ausbreitungsmodelle eingesetzt werden, die für eine Bewertung und Optimierung neuer Kommunikationssysteme benötigt werden.

Der Trend zu höheren Übertragungsraten und die damit verbundene Verkleinerung der Zellengrößen macht den Einsatz deterministischer Verfahren zu einem immer wichtigeren Werkzeug der Funknetzplanung. Die Anwendbarkeit und Akzeptanz deterministischer Modelle hängt jedoch entscheidend von den Simulationslaufzeiten ab. Die Verkürzung der Rechenzeit ist somit das primäre Ziel bei der Entwicklung neuer deterministischer Verfahren. Dieses Ziel kann auf zwei Wegen erreicht werden. Zum Einen kann eine deutliche Beschleunigung der Berechnung durch den Einsatz eines Parallelrechners erreicht werden. Dazu müssen die zugrundeliegende Verfahren parallelisiert und die Ressourcen des Parallelrechners effizient genutzt werden. Zum Anderen kann der Einsatz hierarchischer Algorithmen im Zusammenhang mit einer adaptiven Anpassung der Strahlendichte an die Geometrie des Szenarios die Effizienz des deterministischen Modells weiter steigern. Durch beide Maßnahmen wird die Simulation größerer, komplexerer Geometrien mit einer deutlich verbesserten Detailtreue erst möglich.

Diese Arbeit beschreibt die Entwicklung eines parallelen hierarchischen Verfahrens mit einer adaptiven Anpassung der Simulationsparameter an die Geometrie des Szenario. Das Verfahren berücksichtigt alle für die Ausbreitung in mikro- und pikozellularen Netzen wesentlichen Effekte und erlaubt eine genaue geometrische Konstruktion der Ausbreitungswege. Den Ausgangspunkt der Arbeit bildet die Methode von [SCHÖBERL \(1997\)](#), die neben Reflexionen und Transmissionen der Wellen an mehrlagigen Wänden auch die Beugung an Kanten nach den Gesetzen der vereinheitlichten Beugungstheorie berücksichtigt. Zusätzlich wird die Streuung an rauen Oberflächen, die besonders im Bereich der Millimeterwellen eine besondere Rolle spielt, beachtet. Die komplexe Bestimmung des Schnittpunktes eines Strahl mit der Geometrie wird mittels hierarchischer räumlicher Partitionierungsbäume nach [FUCHS et al. \(1980\)](#) realisiert, die mit Hilfe der Simulated Annealing Methode ([METROPOLIS et al., 1953](#)) und mit genetischen Algorithmen ([GOLDBERG, 1989](#)) optimiert werden. Die Modellierung der Beugung und der Streuung über Sekundärquellen erlaubt eine einfache und effiziente Parallelisierung des Algorithmus sowohl auf symmetrischen Multiprozessoren als auch auf Rechnern mit einem verteilten Speicher. Ein besonderer Augenmerk liegt auf der Parallelisierung für hybride Rechnerarchitekturen, die die Eigenschaften beider Parallelrechnerklassen vereinen und zu den die meisten der heute eingesetzten Hochleistungsrechner gehören. Ebenfalls werden Besonderheiten der Parallelisierung auf Rechnern mit der *Cache-Coherent Non-Uniform Memory Access* (ccNUMA) Architektur betrachtet. Eine sehr einfache und schnelle Strahlendichtenormierung führt zu einer effizienten Identifizierung neuer Ausbreitungswege mit Hilfe zufällig balanzierter Binärbäume. Im Gegensatz zu der Arbeit von [SCHÖBERL \(1997\)](#) werden für die geometrische

Konstruktion der Ausbreitungswege konische Strahlenbündel verfolgt, die in Abhängigkeit von der Geometrie adaptiv verfeinert werden. Die Auswirkung der Strahlengenerierung auf die effiziente Nutzung der Speicherhierarchie eines modernen Rechners wird ebenfalls untersucht und die so gewonnenen Erkenntnisse werden bei der Implementierung des adaptiven Strahlengenerators berücksichtigt.

Im zweiten Kapitel werden nur kurz die wesentlichen physikalischen Grundlagen der deterministischen Kanalmodellierung beschrieben. Das dritte Kapitel beschäftigt sich mit den algorithmischen Grundlagen der Strahlverfolgung. Kapitel 4 ist der Konstruktion der für die schnelle Schnittpunktsuche benötigten räumlichen binären Partitionierungsbäume gewidmet. An dieser Stelle wird ebenfalls die in dieser Arbeit entwickelte Modifikation des klassischen Verfahrens hin zu der allgemeinen Repräsentation räumlicher Daten in multidimensionalen topologischen Bäumen vorgestellt. Im Kapitel 5 werden Methoden für die Konstruktion optimaler Partitionierungsbäume vorgestellt. Die Implementierung des adaptiven Cone Tracing Algorithmus wird zusammen mit der erreichten Effizienz im Kapitel 6 behandelt. Die Parallelisierung auf Multiprozessorsystemen mit gemeinsamem Speicher und die Implementierung auf Parallelrechnern mit verteiltem Speicher wird im folgenden Kapitel 7 betrachtet. Im Kapitel 8 wird das verwendete physikalische Modell durch einen Vergleich mit exakten Verfahren und Channel Sounder Messungen validiert und das Ergebnis einiger Anwendungen der Methode bei der Funknetzplanung und der Ausbreitungsberechnung präsentiert. Eine Zusammenfassung folgt im Kapitel 9.



## Physikalische Grundlagen

Dieses Kapitel enthält eine Übersicht der in den folgenden Kapiteln vorausgesetzten physikalischen Grundlagen. Ausgehend von der Formulierung des Wellenausbreitungsproblems als eine asymptotische Näherung im Hochfrequenzfall werden die wichtigsten kanonischen Fälle wie die Reflexion und die Brechung einer transversalen elektromagnetischen (TEM) Welle an einer geschichteten Wand, die Beugung an Kanten und die Streuung an rauen Oberflächen betrachtet. Vor der Definition einer geeigneten Darstellung der polarimetrischen Übertragungsparameter eines Ausbreitungspfades werden die benötigten Koordinatentransformationen vorbereitend eingeführt. Es werden zwei verschiedene Möglichkeiten der Modellierung einer geschichteten Wand vorgestellt und ihre Vor- und Nachteile diskutiert. Die Grundzüge der Beugung elektromagnetischer Wellen an metallenen und dielektrischen Kanten werden im Rahmen der vereinheitlichten Theorie der Beugung (*Uniform Theory of Diffraction* (UTD)) dargelegt. Abschließend wird ein einfaches Modell für die Berechnung der Streuung an fraktalen rauen Oberflächen mit der Momentenmethode betrachtet.

### 2.1 Die geometrisch-optische Näherung

Im Hochfrequenzfall kann die Interaktion des elektromagnetischen Feldes mit der Umgebung in verschiedene, voneinander unabhängige Effekte, die so genannten kanonischen Fälle, aufgeteilt werden. Zu den wichtigsten Effekten gehören neben der Wellenausbreitung im freien Raum vor allem die Reflexion, die Brechung, die Beugung und die Streuung elektromagnetischer Wellen. Die Ausbreitungspfade der durch Strahlen angenäherten transversalen elektromagnetischen Wellen gehorchen dem durch die folgende Gleichung definierten *Fermatschen Prinzip*:

$$L(C) = \int_C n d\sigma . \quad (2.1)$$

Nach dem Fermatschen Prinzip breiten sich die Wellen entlang derjenigen Kurven  $C$  aus, die das obige Integral minimieren<sup>1</sup>. Dieses Linienintegral definiert die optische Länge  $L$  des Pfades  $C$  bzw. die optische Distanz des Start- und Endpunkts des Strahls als Funktion des positionsabhängigen Brechungsindex  $n(\vec{r})$ .

Bereits Hamilton (1834) konnte zeigen, dass ein beliebiges optisches System eindeutig beschrieben werden kann, indem die optische Länge  $L(\vec{r}_a, \vec{r}_b)$  für alle möglichen Kombinationen der Start- und Endpunkte  $\vec{r}_a$  und  $\vec{r}_b$  eines Strahls im System angegeben wird (DESCHAMPS, 1972). SOMMERFELD und RUNGE (1911) leiteten die Gültigkeitsbedingungen der geometrisch-optischen Näherung aus Betrachtungen des Energietransports innerhalb eines Strahlenbündels her. Sie konnten zeigen, dass für das skalare Feld  $A \exp(jk_0 L)$  mit der Amplitude  $A$  und der Wellenzahl  $k_0$  die Gültigkeit der Näherung für verschwindende  $\Delta A/A$  gegeben ist. Eine ähnliche Folgerung für vektorielle Felder wurde von IGNATOWSKY

<sup>1</sup>Die Beugung und die Streuung elektromagnetischer Wellen wird durch eine Erweiterung des Fermatschen Prinzips beschrieben (KOUYOUMJIAN, 1965). Nach dieser Erweiterung muss das Fermatsche Integral stationär sein – die optische Länge bzw. die Laufzeit der Welle darf also neben der Minimalwerte auch Maximalwerte oder Werte der Sattelpunkte der Funktion annehmen (FEYNMAN, 1970).

(1919) und FRIEDLANDER (1947) direkt aus den Maxwell'schen Gleichungen hergeleitet. In Folge dieser Betrachtung wurde das Versagen der geometrisch-optischen Näherung in der Nähe von Kaustiken festgestellt, wobei in diesen Bereichen das Feld durch andere geeignete Methoden bestimmt werden kann (SCHROTH und STEIN, 1985).

Im folgenden Text wird in der entsprechenden Kürze eine Herleitung der asymptotischen Näherung im Hochfrequenzfall aus den Maxwell'schen Gleichungen angegeben. Anschließend werden die für die Wellenausbreitung in den betrachteten Anwendungsgebieten benötigten Grundlagen sowie die bereits erwähnten kanonischen Fälle betrachtet. Hierzu gehören insbesondere die Ausbreitungseffekte durch mehrschichtige dielektrische Wände, Beugungseffekte an Kanten und die Streuung an zufällig rauen Oberflächen mit verschiedenen spektralen Eigenschaften.

## 2.2 Die Herleitung der strahlenoptischen Näherung

Die Ausbreitung elektromagnetischer Wellen wird durch das System der Maxwell'schen Gleichungen<sup>2</sup> beschrieben (MAXWELL, 1954)

$$\nabla \cdot \vec{D} = \rho \quad (2.2)$$

$$\nabla \times \vec{E} = -\partial_t \vec{B} \quad (2.3)$$

$$\nabla \cdot \vec{B} = 0 \quad (2.4)$$

$$\nabla \times \vec{H} = \vec{J} + \partial_t \vec{D} \quad (2.5)$$

Mit  $\partial_t$  wird abkürzend die partielle Ableitung nach der Zeit bezeichnet. Um diese Gleichungen tatsächlich zur Lösung eines Problems einsetzen zu können, werden noch die folgenden konstitutiven Beziehungen zwischen den Feldgrößen benötigt, die eine makroskopische Auswirkung der Materialparameter auf das Feld definieren

$$\vec{D} = \epsilon \vec{E} \quad (2.6)$$

$$\vec{B} = \mu \vec{H} \quad (2.7)$$

$$\vec{J} = \sigma \vec{D} \quad (2.8)$$

Die Materialparameter werden im allgemeinsten Fall als anisotrope Größen durch die frequenzabhängigen tensoriellen Funktionen  $\epsilon$  für die dielektrischen Eigenschaften und  $\mu$  für die magnetischen Eigenschaften beschrieben. Die Leitfähigkeit  $\sigma$  kann ebenfalls durch einen Tensor erfasst werden, wobei sie und auch die beiden anderen Materialparameter im Rahmen dieser Arbeit als isotrope, innerhalb der Simulationsbandbreite frequenzunabhängige Größen betrachtet werden.

Für eine harmonische Zeitabhängigkeit  $\exp(j\omega t)$  der Felder wird  $\partial_t$  zu  $j\omega$  und der Term  $\exp(j\omega t)$  wird im Folgenden in den Gleichungen weggelassen. Aus den Maxwell'schen Gleichungen können für den Fall verschwindender Raumladung  $\rho = 0$  sowie verschwindender Stromdichte  $\vec{J} = 0$  und einen ortsabhängigen Brechungsindex  $n(\vec{r}) = \sqrt{\epsilon_r(\vec{r})}$  die folgenden

<sup>2</sup> In der Schreibweise der von HESTENES (1966) entwickelten *Space-Time Algebra* (STA) lassen sich diese Gleichungen in einer einzigen, alle elektromagnetischen Effekte beschreibenden Gleichung  $\nabla F = J$  zusammenfassen. Das durch den Multivektor  $F$  beschriebene Feld wird als eine in sich konsistente Kombination des Vektors der elektrischen Feldstärke mit dem Bivektor der magnetischen Feldstärke  $F = \vec{E} + I\vec{H}$  ausgedrückt. Die Größe  $I$  bezeichnet das Pseudoskalar, welches das Analogon zu der imaginären Zahl  $i$  darstellt. Im Gegensatz zu  $i$  besitzt  $I$  jedoch eine genau definierte geometrische Bedeutung. Im Laufe der Zeit konnte in vielen Fällen das Auftreten der komplexen Zahlen bei der Beschreibung physikalischer Vorgänge auf geometrische Gründe zurückgeführt werden (GULL et al., 1993).

Wellengleichungen hergeleitet werden (SCHROTH, 1978)

$$\Delta \vec{E} + k_0^2 n^2 \vec{E} + 2\nabla \left( \vec{E} \cdot \nabla \ln(n) \right) = 0 \quad (2.9)$$

$$\Delta \vec{H} + k_0^2 n^2 \vec{H} + 2\nabla \ln(n) \times \left( \nabla \times \vec{H} \right) = 0 . \quad (2.10)$$

Die Beschreibung des vektoriellen elektrischen Feldes durch eine Luneberg-Kline Reihe (KLINE und KAY, 1965)

$$\vec{E}(\vec{r}) = e^{-jk_0\Psi(\vec{r})} \sum_{m=0}^{\infty} \frac{\vec{E}_m(\vec{r})}{(jk_0)^m} \quad (2.11)$$

sowie der folgende Koeffizientenvergleich führt zu den Eikonalg- und Transportgleichungen (SCHROTH und STEIN, 1985)

$$(\nabla\Psi)^2 = n^2 \quad (2.12)$$

$$\vec{E}_0 \Delta\Psi + 2 \left( \vec{E}_0 \cdot \nabla \ln(n) \right) \nabla\Psi + 2(\nabla\Psi \cdot \nabla) \vec{E}_0 = 0 \quad (2.13)$$

$$\begin{aligned} & \left( \Delta\Psi \bar{\mathbf{1}} + 2\nabla\Psi \nabla \ln(n) + 2\nabla\Psi \cdot \nabla \bar{\mathbf{1}} \right) \cdot \vec{E}_m = \\ & = \Delta \vec{E}_{m-1} + 2\nabla \left( \nabla \ln(n) \cdot \vec{E}_{m-1} \right), \quad \text{für } m = 1, 2, \dots \end{aligned} \quad (2.14)$$

Die Luneberg-Kline Reihe konvergiert für  $\omega \rightarrow \infty$  gegen die exakte Lösung der Wellengleichung und stellt somit eine asymptotische Näherung dieser Lösung dar.  $\Psi(\vec{r})$  wird als die Phasenfunktion,  $\vec{E}_m(\vec{r})$  als die vektorielle Amplitudenfunktion bezeichnet.  $k_0 = \omega/c_0$  ist die Wellenzahl im freien Raum. Das Symbol  $\bar{\mathbf{1}}$  bezeichnet die Einheitsdyade.

Der erste Term der Luneberg-Kline Reihe dominiert im Bereich kleiner Wellenlängen und wird als die geometrisch-optische Lösung bezeichnet

$$\vec{E} = \vec{E}_0 e^{-jk_0\Psi} \quad \text{bzw.} \quad \vec{H} = \vec{H}_0 e^{-jk_0\Psi} . \quad (2.15)$$

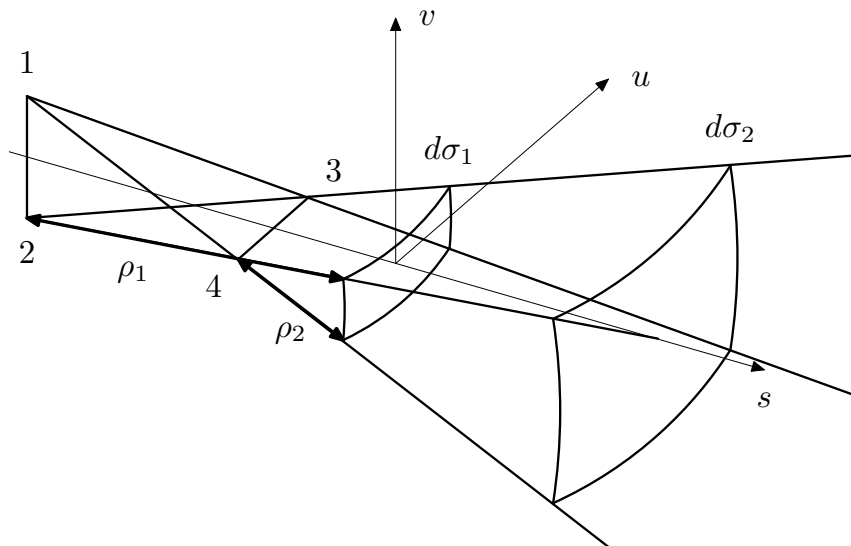
Dieser Ausdruck bildet die Grundlage der strahlenoptischen Verfahren. Der Strahlweg als Funktion des Brechungsindexgradienten wird durch Lösung der Eikonalggleichung bestimmt. Für den Fall einer Ausbreitung in einem homogenen Medium mit einem konstanten Brechungsindex ergibt sich für der Strahlweg  $\vec{R}$

$$\vec{R}(s) = \vec{A}s + \vec{B} . \quad (2.16)$$

Die Strahlen breiten sich in diesem Fall stets entlang gerader Linien aus. Die Phasenfronten stehen hierbei senkrecht zu der Strahlrichtung und die Wellenfronten besitzen Eigenschaften lokal ebener, transversaler elektromagnetischer Wellen (TEM Wellen). Die Wellenfront besitzt eine durch die Quelle und ihre weitere Interaktion mit der Umgebung gegebene Krümmung (Bild 2.1). Durch die Betrachtung der Krümmungsradien der Wellenfronten werden Fokussierungs- und Defokussierungseffekte durch glatte, gekrümmte Flächen der Geometrie berücksichtigt. In der Umgebung der Brennpunkte bzw. der Brennpunkte (*Kaustiken*) wird das Feld singulär und muss mit anderen geeigneten Mitteln berechnet werden. Da jedoch alle Flächen der Geometrien im Anwendungsgebiet dieser Arbeit mit Hilfe planarer Polygonzüge angenähert werden, besitzen alle auftretenden Wellenfronten die Krümmung sphärischer Wellen mit ihren beiden Hauptkrümmungsradien gleich der Strahllänge. Die Krümmung der sphärischen Wellenfronten bleibt auch nach einer Interaktion mit einer planaren Fläche erhalten. Somit stellt der Ursprung der Welle die einzig mögliche Kaustik dar. Die Transportgleichung der nullten Ordnung wird in diesen Sonderfall durch die Greensche Funktion des freien Raumes erfüllt:

$$\vec{E}(s) = \vec{E}(0) \frac{1}{s} e^{-jk_0 s} \quad \text{bzw.} \quad \vec{H}(s) = \vec{H}(0) \frac{1}{s} e^{-jk_0 s} . \quad (2.17)$$

Eine allgemeine und ausführliche Betrachtung der Krümmung der Wellenfronten nach einer Interaktion mit beliebig gekrümmten Grenzflächen sowie die Berechnung des Feldes in der Umgebung von Kaustiken ist zum Beispiel in [SCHROTH und STEIN \(1985\)](#) zu finden.



**Bild 2.1:** Die Definition der Hauptkrümmungsradien  $\rho_1$  und  $\rho_2$  eines Strahlenbündels in einem homogenen Medium. An den Kaustiken (1—2) bei  $x = -\rho_1$  und (3—4) bei  $x = -\rho_2$  wird das Feld singulär und muss mit anderen Verfahren berechnet werden.

### 2.3 Die polarimetrische Übertragungsfunktion

Die elektrischen Eigenschaften eines Ausbreitungsweges zwischen dem Sender und dem Empfänger werden durch die polarimetrische Übertragungsfunktion definiert. Diese Funktion beschreibt neben der relativen Phasenlage und der Dämpfung des Signals auch die Drehung der Polarisation einer Welle, die über diesen Ausbreitungsweg den Empfänger erreicht. Diese Eigenschaften werden allgemein in Form einer komplexen Übertragungsmatrix zwischen den Bezugssystemen des Senders und des Empfängers angegeben. Da das Fernfeld der Antennen in sphärischen Koordinaten besonders einfach ausgedrückt werden kann, liegt die Wahl dieses Koordinatensystems als das Bezugssystem für die Übertragungsfunktion nahe<sup>3</sup>. Die Gleichung

$$\begin{bmatrix} \underline{E}_v \\ \underline{E}_h \end{bmatrix}_{Rx} = \begin{bmatrix} \underline{S}_{vv} & \underline{S}_{vh} \\ \underline{S}_{hv} & \underline{S}_{hh} \end{bmatrix} \cdot \begin{bmatrix} \underline{E}_v \\ \underline{E}_h \end{bmatrix}_{Tx} \quad (2.18)$$

beschreibt vollständig die Übertragung der komplexen Wellengrößen des Senders über den durch die komplexe Matrix  $\underline{S}$  beschriebenen Ausbreitungsweg. Die Indizes der komplexen Wellenamplituden  $\underline{E}$  bezeichnen die vertikale und die horizontale Polarisation. Ebenso

<sup>3</sup>Das sphärische Koordinatensystem bietet außerdem den Vorteil der kleinsten Datenmenge, die für jeden Ausbreitungsweg gespeichert werden muss, da die Speicherung der Daten für die nicht vorhandenen radialen Komponenten des Feldes entfällt. Somit muss für jeden Ausbreitungsweg nur eine komplexe  $2 \times 2$  Matrix gespeichert werden. Bei anderen Koordinatensystemen ist eine Trennung der radialen und der transversalen Komponenten des Feldes nicht möglich, und es muss die vollständige komplexe  $3 \times 3$  Matrix gespeichert werden. Dadurch würde sich der für die Speicherung der Ergebnisse benötigte Speicherplatz mehr als verdoppeln.



geben die Indizes der komplexen Matrixkomponenten auf der Hauptdiagonale der Matrix die jeweiligen kopolaren Übertragungsfaktoren. Die Elemente der Gegendiagonale enthalten die kreuzpolaren Anteile und bestimmen somit die depolarisierenden Eigenschaften des Kanals.

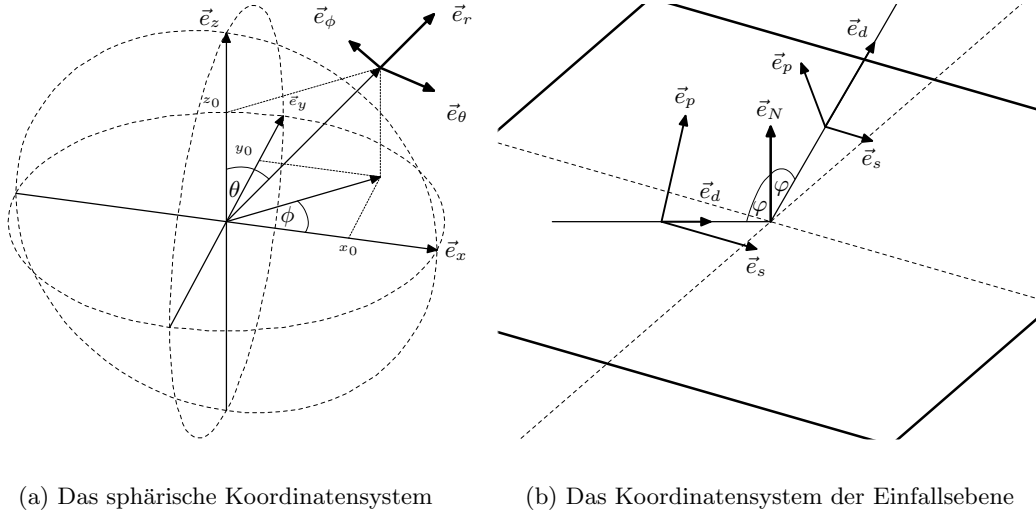
Die sphärische Koordinate  $(\theta, \phi)$  definiert in globalen kartesischen Koordinaten das folgende rechtwinklige Dreibein:

$$\vec{e}_r = \sin(\theta) \cos(\phi) \vec{e}_x + \sin(\theta) \sin(\phi) \vec{e}_y + \cos(\theta) \vec{e}_z \quad (2.19)$$

$$\vec{e}_\theta = \cos(\theta) \cos(\phi) \vec{e}_x + \cos(\theta) \sin(\phi) \vec{e}_y - \sin(\theta) \vec{e}_z \quad (2.20)$$

$$\vec{e}_\phi = -\sin(\phi) \vec{e}_x + \cos(\phi) \vec{e}_y \quad (2.21)$$

Hierbei definiert der Einheitsvektor  $\vec{e}_\theta$  die Richtung der vertikalen Polarisation und der Einheitsvektor  $\vec{e}_\phi$  die Richtung der horizontalen Polarisation einer sich in  $\vec{e}_r$  Richtung ausbreitenden Wellenfront. Falls das lokale Koordinatensystem der Antenne gegenüber dem globalen Koordinatensystem gedreht ist, müssen die Antennenparameter über geeignete Koordinatentransformationen aus dem lokalen Koordinatensystem der Antenne in das globale Koordinatensystem überführt werden. Die Orientierung der sphärischen Koordinaten ist im Bild 2.2a dargestellt.



(a) Das sphärische Koordinatensystem

(b) Das Koordinatensystem der Einfallsebene

**Bild 2.2:** Zur Definition der Koordinatensysteme.

Ein weiteres wichtiges Bezugssystem wird durch die Koordinaten der Einfallsebene einer Welle auf eine Grenzfläche definiert. Der Vektor der elektrischen Feldstärke der einfallenden Welle wird in die Komponenten senkrecht ( $\vec{E}_s$ ) und parallel ( $\vec{E}_p$ ) zur Einfallsebene der Welle zerlegt. Die Einfallsebene ist durch die Ausbreitungsrichtung der Welle  $\vec{e}_d$  und die Richtung des Normalenvektors der Grenzfläche  $\vec{e}_N$  im Auftreffpunkt definiert. Die zugehörige Geometrie ist im Bild 2.2b zu sehen.

Beide Richtungsvektoren spannen die Einfallsebene auf. Außer im Fall des senkrechten Einfalls, in dem beide Richtungsvektoren parallel sind, ist der Einheitsvektor senkrecht zur Einfallsebene  $\vec{e}_s$  gegeben durch

$$\vec{e}_s = \frac{\vec{e}_d \times \vec{e}_N}{\|\vec{e}_d \times \vec{e}_N\|} \quad (2.22)$$

Der Einheitsvektor parallel zur Einfallsebene kann dann leicht aus  $\vec{e}_s$  und der Ausbreitungsrichtung  $\vec{e}_d$  der Welle bestimmt werden

$$\vec{e}_p = \vec{e}_s \times \vec{e}_d . \quad (2.23)$$

Die Koordinatentransformation der Wellengrößen aus den sphärischen Koordinaten in die Koordinaten der Einfallsebene wird durch die Multiplikation mit der  $2 \times 2$  Drehmatrix erreicht.

$$\begin{bmatrix} \underline{E}_{vp} & \underline{E}_{hp} \\ \underline{E}_{vs} & \underline{E}_{hs} \end{bmatrix} = \begin{bmatrix} \vec{e}_v \cdot \vec{e}_p & \vec{e}_h \cdot \vec{e}_p \\ \vec{e}_v \cdot \vec{e}_s & \vec{e}_h \cdot \vec{e}_s \end{bmatrix} \cdot \begin{bmatrix} \underline{E}_v & 0 \\ 0 & \underline{E}_h \end{bmatrix} . \quad (2.24)$$

In gleicher Weise werden die Transformationen der Wellengrößen zwischen den verschiedenen Einfallsebenen entlang des Ausbreitungsweges und die abschließende Transformation in die sphärischen Koordinaten des Empfängers durchgeführt. In der Matrixschreibweise kann die komplexe Übertragungsmatrix als ein Produkt aus Transformations- und Streumatrizen geschrieben werden.

Zu einer weiteren Vereinfachung der Berechnung kann die Tatsache ausgenutzt werden, dass sich in Mobilfunkszenarien die Wellen im freien Raum ausbreiten und nur an polygonalen Wänden gestreut werden. Damit sind entlang des Ausbreitungsweges die Materialparameter des freien Raumes gegeben und die Materialeigenschaften der Wände einschließlich der Wanddicke werden in den Streuparametern berücksichtigt (siehe Abschnitt 2.4). In diesem Fall kann die Greensche Funktion des freien Raumes vor das Produkt geschrieben werden.

$$\underline{\mathbf{S}} = \frac{e^{-jk_0 r}}{r} \mathbf{T}_{N+1} \cdot \underline{\mathbf{S}}_N \cdot \mathbf{T}_N \cdots \underline{\mathbf{S}}_2 \cdot \mathbf{T}_2 \cdot \underline{\mathbf{S}}_1 \cdot \mathbf{T}_1 = \frac{e^{-jk_0 r}}{r} \mathbf{T}_{N+1} \cdot \prod_{n=N}^1 \underline{\mathbf{S}}_n \cdot \mathbf{T}_n \quad (2.25)$$

Die komplexe Übertragungsmatrix  $\underline{\mathbf{S}}$  hat die Form einer Produktreihe aus einander abwechselnden reellen Transformationsmatrizen  $T_n$  und komplexen Streumatrizen  $\underline{\mathbf{S}}_n$ . Die Phasendrehung und Freiraumdämpfung des Ausbreitungspfades wird durch die skalare Greensche Funktion erfasst, wobei  $r$  die Länge des Ausbreitungswegs ist. Die Reihenfolge der Matrixmultiplikationen bei der Berechnung von  $\underline{\mathbf{S}}$  ist nicht beliebig wählbar. Sie muss entsprechend der Gleichung (2.25) beibehalten werden.

## 2.4 Reflexion und Brechung

Die einfache Reflexion an einer Grenzschicht (Bild 2.2b) zwischen zwei isotropen homogenen und nur schwach verlustbehafteten Materialien mit den elektrischen Parametern  $(\epsilon_1, \mu_1)$  und  $(\epsilon_2, \mu_2)$  wird durch die folgende Matrixoperation beschrieben:

$$\begin{bmatrix} \underline{E}_p \\ \underline{E}_s \end{bmatrix}_r = \begin{bmatrix} \underline{R}_p & 0 \\ 0 & \underline{R}_s \end{bmatrix} \cdot \begin{bmatrix} \underline{E}_p \\ \underline{E}_s \end{bmatrix}_i . \quad (2.26)$$

Die komplexen Reflexionsfaktoren  $\underline{R}_s$  für die transversal elektrische Komponente  $\underline{E}_s$  und  $\underline{R}_p$  für die transversal magnetische Komponente  $\underline{E}_p$  der aus dem Medium 1 in das Medium 2 einfallenden elektrischen Feldstärke ergeben sich zu (GENG und WIESBECK, 1998)

$$\underline{R}_p = \frac{\underline{Z}_1 \cos \varphi_i - \underline{Z}_2 \cos \varphi_t}{\underline{Z}_1 \cos \varphi_i + \underline{Z}_2 \cos \varphi_t} \quad (2.27)$$

$$\underline{R}_s = \frac{\underline{Z}_2 \cos \varphi_i - \underline{Z}_1 \cos \varphi_t}{\underline{Z}_2 \cos \varphi_i + \underline{Z}_1 \cos \varphi_t} . \quad (2.28)$$

$\underline{Z}_n$  ist der komplexe Feldwellenwiderstand in dem entsprechenden Medium  $n$ . Auf die gleiche Art wird auch die Transmission der Welle durch die Grenzschicht hindurch bestimmt:

$$\begin{bmatrix} \underline{E}_p \\ \underline{E}_s \end{bmatrix}_t = \begin{bmatrix} \underline{T}_p & 0 \\ 0 & \underline{T}_s \end{bmatrix} \cdot \begin{bmatrix} \underline{E}_p \\ \underline{E}_s \end{bmatrix}_i \quad (2.29)$$

mit

$$\underline{T}_p = \frac{2\underline{Z}_2 \cos \varphi_i}{\underline{Z}_1 \cos \varphi_i + \underline{Z}_2 \cos \varphi_t} \quad (2.30)$$

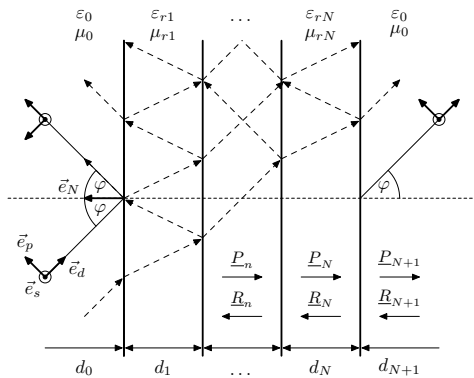
$$\underline{T}_s = \frac{2\underline{Z}_2 \cos \varphi_i}{\underline{Z}_2 \cos \varphi_i + \underline{Z}_1 \cos \varphi_t} \quad (2.31)$$

Der Ausfallswinkel  $\varphi_r$  der reflektierten Welle relativ zum Normalenvektor der Grenzfläche ist nach dem Fermatschen Prinzip gleich dem Einfallswinkel  $\varphi_i$  der einfallenden Welle. Für den Ausfallswinkel  $\varphi_t$  einer aus dem Medium 1 in das Medium 2 transmittierten Welle gilt nach dem Brechungsgesetz von Snellius

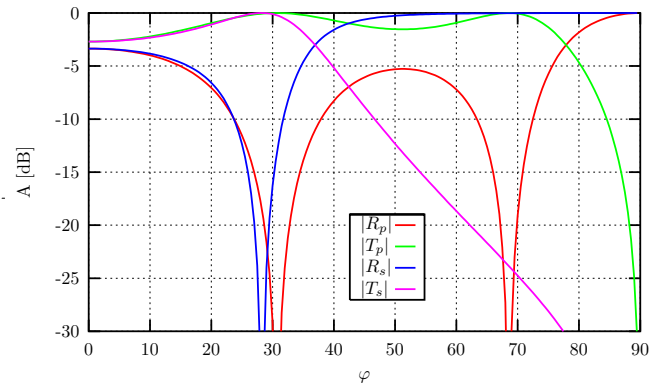
$$k_1 \sin(\varphi_i) = k_2 \sin(\varphi_t) \quad , \quad (2.32)$$

mit  $k_1 = 2\pi f \sqrt{\epsilon_1 \mu_1}$  der Wellenzahl im Medium 1. Analoges gilt auch für Medium 2.

Im Rahmen dieser Arbeit wird die Reflexion und die Refraktion einer ebenen Welle an einem geschichteten Medium (Bild 2.3a) betrachtet. Das analytische Modell berücksichtigt in den Reflexions- und Transmissionskoeffizienten des geschichteten Mediums sowohl die Dicke der einzelnen Schichten des Schichtensystems als auch die unendliche Reihe der Reflexionen und Refraktionen der Wellen innerhalb des Schichtensystems, die zu ausgeprägten Interferenzeffekten und somit zu einer deutlichen Abhängigkeit der Reflexions- und Transmissionsfaktoren von dem Winkel  $\varphi$  der einfallenden Welle führen (Bild 2.3b).



(a) Aufbau des Mehrschichtensystems



(b) Koeffizienten von Fensterglas

**Bild 2.3:** Die Geometrie einer wärmedämmendem Fensterglasscheibe mit den zugehörigen Reflexions- und Transmissionskoeffizienten bei 10.5 GHz, aufgetragen in Abhängigkeit des Einfallswinkels  $\varphi$ . Das Fenster besteht aus zwei 4 mm dicken Glasscheiben ( $\epsilon_r = 6.5 - j0.001$ ), die durch einem 14 mm dicken Luftspalt ( $\epsilon_r = 1$ ) getrennt sind.

Die Dicke des geschichteten Dielektrikums wird bereits bei der Berechnung der Koeffizienten berücksichtigt. Da das geometrische Modell der Wand jedoch von einer Polygondarstellung mit einer verschwindenden Dicke ausgeht, entsteht so ein Phasenfehler, der durch eine einfache Erweiterung des Wandmodells korrigiert werden muss.

Die Wand besteht aus  $N$  Schichten mit den elektrischen Parametern  $\epsilon_i, \mu_i$  mit  $i = 1 \dots N$ . Bei verlustbehafteten Materialien werden diese Parameter zu komplexen Größen. Die erste und die letzte Schicht sind unendlich ausgedehnt und besitzen die Eigenschaften des freien Raumes. Auf das Schichtensystem fällt unter dem Winkel  $\varphi$  (Bild 2.3a) von der linken Seite eine Welle mit der Amplitude  $\underline{P}_0$  ein. Die Reflexion an dem Schichtensystem ist durch die Amplitude der reflektierten Welle in der Schicht 0 mit der komplexen Amplitude  $\underline{R}_0$  gegeben. Die Transmission durch das Schichtensystem hindurch wird durch die in die letzte Schicht  $N + 1$  austretende Welle mit der Amplitude  $\underline{P}_{N+1}$  definiert. Sowohl die reflektierte als auch die transmittierte Welle ergibt sich aus der Berechnung der sich innerhalb des Schichtensystems ausbreitenden Wellen. Die Felder innerhalb einer Schicht können allgemein in vorwärts und rückwärts laufende Wellen mit den komplexen Amplituden  $\underline{P}_n$  und  $\underline{R}_n$  zerlegt werden. Die Richtung der Wellenkomponenten ist ebenfalls Bild 2.3a zu entnehmen.

Mit

$$\gamma_n = 2\pi f \sqrt{\epsilon_{r,n} \mu_{r,n} - \epsilon_0 \mu_0 \sin^2 \varphi} \quad (2.33)$$

kann für die transversal elektrische Komponente des auf die Grenzschicht  $(n, n + 1)$  einfallenden Feldes geschrieben werden (REMBOLD, 2000):

$$\begin{bmatrix} \underline{P}_n \\ \underline{R}_n \end{bmatrix}_s = \frac{1}{2} \begin{bmatrix} e^{+j\gamma_n d_n} & 0 \\ 0 & e^{-j\gamma_n d_n} \end{bmatrix} \cdot \begin{bmatrix} 1 + \frac{\gamma_{n+1} \mu_{r,n}}{\gamma_n \mu_{r,n+1}} & 1 - \frac{\gamma_{n+1} \mu_{r,n}}{\gamma_n \mu_{r,n+1}} \\ 1 - \frac{\gamma_{n+1} \mu_{r,n}}{\gamma_n \mu_{r,n+1}} & 1 + \frac{\gamma_{n+1} \mu_{r,n}}{\gamma_n \mu_{r,n+1}} \end{bmatrix} \cdot \begin{bmatrix} \underline{P}_{n+1} \\ \underline{R}_{n+1} \end{bmatrix}_s \quad (2.34)$$

Ebenso gilt für die transversal magnetische Komponente des Feldes:

$$\begin{bmatrix} \underline{P}_n \\ \underline{R}_n \end{bmatrix}_p = \frac{1}{2} \begin{bmatrix} e^{+j\gamma_n d_n} & 0 \\ 0 & e^{-j\gamma_n d_n} \end{bmatrix} \cdot \begin{bmatrix} 1 + \frac{\gamma_{n+1} \epsilon_{r,n}}{\gamma_n \epsilon_{r,n+1}} & 1 - \frac{\gamma_{n+1} \epsilon_{r,n}}{\gamma_n \epsilon_{r,n+1}} \\ 1 - \frac{\gamma_{n+1} \epsilon_{r,n}}{\gamma_n \epsilon_{r,n+1}} & 1 + \frac{\gamma_{n+1} \epsilon_{r,n}}{\gamma_n \epsilon_{r,n+1}} \end{bmatrix} \cdot \begin{bmatrix} \underline{P}_{n+1} \\ \underline{R}_{n+1} \end{bmatrix}_p \quad (2.35)$$

Die komplexen Wellengrößen  $\underline{P}_n$  und  $\underline{R}_n$  stellen die Amplituden der sich in der Schicht  $n$  in vorwärtige und rückwärtige Richtung ausbreitenden Wellen dar. Die Ausbreitung durch das gesamte Schichtensystem wird dann durch die folgenden Produktreihen beschrieben

$$\begin{bmatrix} \underline{P}_0 \\ \underline{R}_0 \end{bmatrix}_s = \frac{1}{2} \prod_{n=N}^0 \left\{ \begin{bmatrix} e^{+j\gamma_n d_n} & 0 \\ 0 & e^{-j\gamma_n d_n} \end{bmatrix} \cdot \begin{bmatrix} 1 + \frac{\gamma_{n+1} \mu_{r,n}}{\gamma_n \mu_{r,n+1}} & 1 - \frac{\gamma_{n+1} \mu_{r,n}}{\gamma_n \mu_{r,n+1}} \\ 1 - \frac{\gamma_{n+1} \mu_{r,n}}{\gamma_n \mu_{r,n+1}} & 1 + \frac{\gamma_{n+1} \mu_{r,n}}{\gamma_n \mu_{r,n+1}} \end{bmatrix} \right\} \cdot \begin{bmatrix} \underline{P}_{N+1} \\ \underline{R}_{N+1} \end{bmatrix}_s \quad (2.36)$$

$$\begin{bmatrix} \underline{P}_0 \\ \underline{R}_0 \end{bmatrix}_p = \frac{1}{2} \prod_{n=N}^0 \left\{ \begin{bmatrix} e^{+j\gamma_n d_n} & 0 \\ 0 & e^{-j\gamma_n d_n} \end{bmatrix} \cdot \begin{bmatrix} 1 + \frac{\gamma_{n+1} \epsilon_{r,n}}{\gamma_n \epsilon_{r,n+1}} & 1 - \frac{\gamma_{n+1} \epsilon_{r,n}}{\gamma_n \epsilon_{r,n+1}} \\ 1 - \frac{\gamma_{n+1} \epsilon_{r,n}}{\gamma_n \epsilon_{r,n+1}} & 1 + \frac{\gamma_{n+1} \epsilon_{r,n}}{\gamma_n \epsilon_{r,n+1}} \end{bmatrix} \right\} \cdot \begin{bmatrix} \underline{P}_{N+1} \\ \underline{R}_{N+1} \end{bmatrix}_p \quad (2.37)$$

Die Produktreihen werden iterativ unter der Annahme einer verschwindenden rücklaufenden Welle  $\underline{R}_{N+1} = 0$  in der Schicht  $N + 1$  gelöst. Für die vorwärts laufende Welle in dieser Schicht wird  $\underline{P}_{N+1} = 1$  eingesetzt. Die komplexen Reflexionsfaktoren an der ersten Grenzschicht ergeben sich dann zu

$$\underline{R}_s = \begin{bmatrix} \underline{R}_0 \\ \underline{P}_0 \end{bmatrix}_s \quad \text{und} \quad \underline{R}_p = \begin{bmatrix} \underline{R}_0 \\ \underline{P}_0 \end{bmatrix}_p \quad (2.38)$$

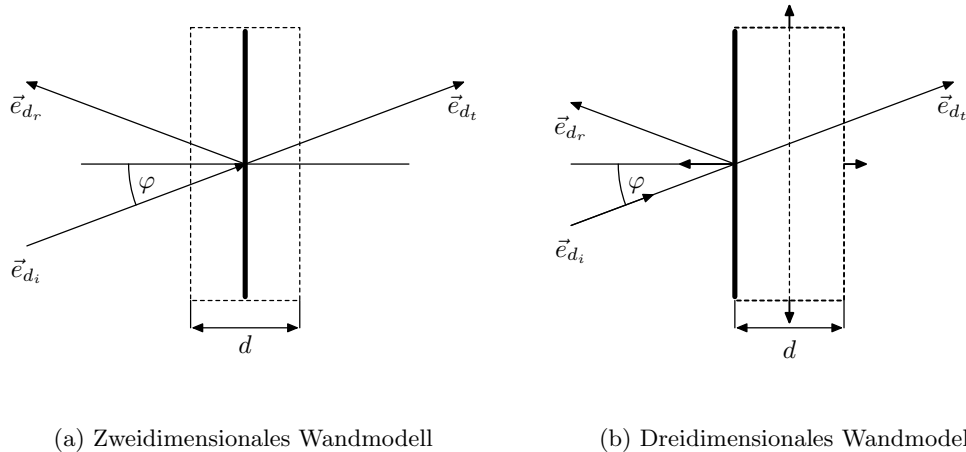
Die Transmissionsfaktoren durch das Schichtensystem können wie folgt berechnet werden

$$\underline{T}_s = \begin{bmatrix} 1 \\ \underline{P}_0 \end{bmatrix}_s \sqrt{\frac{\Re \sqrt{\epsilon_{r,N+1} / \mu_{r,N+1}}}{\Re \sqrt{\epsilon_{r,0} / \mu_{r,0}}}} \quad \text{und} \quad \underline{T}_p = \begin{bmatrix} 1 \\ \underline{P}_0 \end{bmatrix}_p \sqrt{\frac{\Re \sqrt{\mu_{r,N+1} / \epsilon_{r,N+1}}}{\Re \sqrt{\mu_{r,0} / \epsilon_{r,0}}}} \quad (2.39)$$

Mit  $\Re$  wird der Realteil der im Allgemeinen komplexen Wurzel bezeichnet. Unter der Annahme der elektrischen Parameter des freien Raumes auf beiden Seiten des Schichtensystems vereinfachen sich diese Gleichungen zu

$$\underline{T}_s = \left[ \frac{1}{\underline{E}_0} \right]_s \quad \text{und} \quad \underline{T}_p = \left[ \frac{1}{\underline{E}_0} \right]_p . \quad (2.40)$$

Es wurde bereits erwähnt, dass die Dicke der Wand schon bei der Berechnung der Reflexions- und Transmissionskoeffizienten berücksichtigt wird. Da diese elektrischen Parameter später dem geometrischen Modell einer unendlich dünnen Wand zugeordnet werden (Bild 2.4a), entsteht somit ein Phasenfehler, da die Wanddicke ebenfalls in der Strahllänge und damit auch in der Greenschen Funktion nochmals enthalten ist. Dieser Fehler kann entweder durch das Einfügen entsprechender Korrekturterme oder durch eine Modifikation des Wandmodells näherungsweise korrigiert werden. Die erste Maßnahme behält das geometrische Modell der unendlich dünnen Wand bei, um eine möglichst kleine Anzahl von Polygonen bei der Modellbildung zu erhalten. Eine Korrektur des Phasenfehlers wird durch die Multiplikation der Reflexions- und Transmissionskoeffizienten mit dem Faktor  $\exp(ik_0d/\cos\varphi)$  erreicht.



(a) Zweidimensionales Wandmodell

(b) Dreidimensionales Wandmodell

**Bild 2.4:** Verschiedene geometrischen Modelle einer Wand. Das zweidimensionale Modell erlaubt die Nachbildung einfacher Szenarien mit der minimalen Anzahl von benötigten Polygonen, setzt jedoch eine Korrektur der Phase der Streukoeffizienten voraus. Das dreidimensionale Modell bildet die Wand geometrisch richtig nach, fordert aber die Vernachlässigung bestimmter Polygone. Die gestrichelt gezeichneten Wandpolygone stellen die von dem transmittierten Strahl von hinten angestrahlten Flächen dar. Diese Flächen werden von diesem Strahl nicht berücksichtigt.

Das zweite Modell betrachtet die Wand als endlich ausgedehnt und mit einer entsprechenden Anzahl Polygone modelliert. Dieses Wandmodell ist unter anderem auch deswegen nötig, da viele der maschinell generierten Modelle automatisch zu einer solchen Wandgeometrie führen. Die Gesamtzahl der Polygone eines solchen Szenario liegt meistens um den Faktor zwölf höher als bei dem einfachen Wandmodell, da die möglicherweise nicht

konvexen Wandpolygone häufig aus konvexen dreieckigen Polygonen<sup>4</sup> zusammengesetzt werden. Diese Wandgeometrie besitzt, neben der erhöhten Komplexität, auch noch den Nachteil, dass jeder transmittierte Strahl die Wandpolygone auf beiden Seiten der Wand passiert und somit die mit jedem Wandpolygon assoziierten Streuparameter zweimal berücksichtigt werden. Zudem ergeben sich unnötige Mehrfachreflexionen in der Wand, die einerseits bereits in den Streuparametern korrekt berücksichtigt wurden und andererseits auch noch die Rechenzeit vervielfachen.

Um die Fehler des dreidimensionalen Wandmodells zu eliminieren, müssen gezielt Wandpolygone bei der Strahlwegsuche vernachlässigt werden. Die Voraussetzung für die richtige Funktionsweise des Verfahrens ist eine bestimmte gegenseitige Orientierung der Wandpolygone. Da alle Wandpolygone durch den gleichen Algorithmus erzeugt werden, ist die Orientierung der Normalen aller Polygone gleichsinnig und entweder in den Außenraum oder den Innenraum der Wand gerichtet. Im Folgenden wird angenommen, dass alle Normalenvektoren in den Außenraum der Wand orientiert sind. In diesem Fall bildet das dreidimensionale Wandmodell die Wirklichkeit richtig nach, wenn alle, von dem Strahl mit der Richtung  $\vec{e}_d$  von ihrer Rückseite getroffenen Polygone (Bedingung  $\vec{e}_d \cdot \vec{e}_N > 0$ ), als transparent angenommen und bei der Schnittpunktsuche nicht berücksichtigt werden (Bild 2.4b). Die geometrisch richtige Lage der Reflexionsstellen bleibt damit erhalten und es ist keine weitere Korrektur der Streuparameter notwendig. Dies gilt jedoch nur unter der Annahme, dass der Strahl auf der zum Eintrittspunkt entgegengesetzten Seite aus der Wand austritt. Dieses Kriterium dürfte für alle Wände, deren lateralen Abmessungen sehr viel größer als die Wanddicke sind, erfüllt sein. Ebenfalls wird das Feld im Inneren der Wand durch dieses Modell nicht richtig wiedergegeben, wobei diese Bereiche im Einsatzgebiet der Methode ohnehin nicht von Bedeutung sind. Der Rechenaufwand ist weitgehend vergleichbar mit dem einfachen Modell der Wand, wenn auch die Suche nach dem geometrischen Ausbreitungsweg durch das Szenario aufgrund der vielfach höheren Polygonzahl entsprechend aufwendiger ist. Bei der Eingabe der Geometriedaten kann für jedes Polygon festgelegt werden, welches Wandmodell für dieses Polygon benutzt werden soll. Damit ist eine nahtlose Kombination beider Wandmodelle in einer Geometrie möglich.

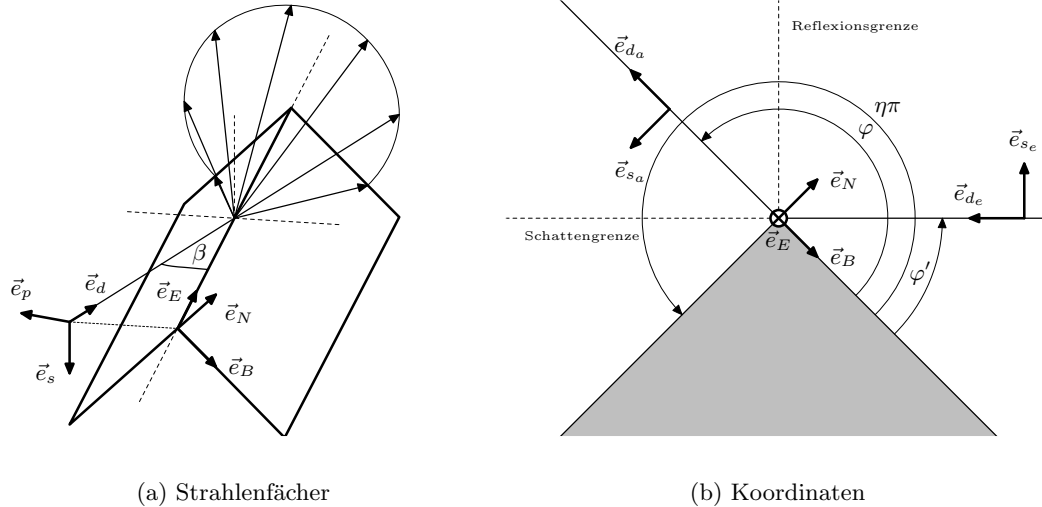
## 2.5 Beugung an Kanten

Neben der Reflexion und Transmission durch Wände stellt die Beugung der elektromagnetischen Wellen an Kanten einen weiteren wichtigen kanonischen Fall dar. Insbesondere im Frequenzbereich der zellularen GSM<sup>5</sup> und UMTS<sup>6</sup> Mobilfunknetze bestimmt die Beugung an Gebäudeecken und Dachkanten wesentlich die Kanaleigenschaften im *No-Line-Of-Sight* (NLOS) Fall. Die geometrische Beugungstheorie beruht einer Erweiterung des Fermatschen Prinzips, wonach ein auf eine Kante treffender Strahl einen kegelförmigen Strahlenfächer erzeugt (Bild 2.5a). Der Öffnungswinkel des Strahlenfächers ist gleich dem Winkel zwischen der Kante und dem einfallenden Strahl. Die Amplitude und die Phase der gebeugten Welle variiert mit dem Azimut  $\varphi$  entsprechend den Beugungskoeffizienten. Im Fall der geometrischen Beugungstheorie (*Geometrical Theory of Diffraction* (GTD)) werden diese Koeffizienten an der Reflexions- und Schattengrenze singulär (KELLER, 1962). Die Singularität kann durch Hinzunahme von Termen höherer Ordnung bei der asymptotischen Reihenentwicklung der Felder behoben werden (KOUYOUMJIAN und PATHAK, 1974).

<sup>4</sup>Der Grund liegt in der algorithmisch nicht trivialen Zerlegung nicht-konvexer Polygone in eine Minimalmenge konvexer Polygone.

<sup>5</sup>*Global System for Mobile Communication*, 900 bzw. 1800 MHz

<sup>6</sup>*Universal Mobile Telecommunication System*, 1900 MHz – 2170 MHz



**Bild 2.5:** Der unter dem Winkel  $\beta$  auf die Kante einfallende Strahl erzeugt einen konischen Strahlenfächer, der nach dem erweiterten Fermatschen Prinzip ebenfalls den Öffnungswinkel  $\beta$  besitzt. Die Orientierung des sphärischen Koordinatensystems der Beugungsquelle wird hierbei durch das kartesische Koordinatensystem der Kante festgelegt. Der Azimutwinkel  $\varphi$  wird auf die Lage der Binormalen  $\vec{e}_B$  bezogen.

Das kantenbezogene Koordinatensystem bestimmt die Orientierung des sphärischen Koordinatensystems der Beugungsquelle (Bild 2.5b). Es wird aus dem Tangentenvektor  $\vec{e}_E$  der Kante und dem Normalenvektor  $\vec{e}_N$  einer der anliegenden Flächen gebildet. Die Binormale  $\vec{e}_B$  ergänzt beide Einheitsvektoren zu einem rechtwinkligen Dreiein

$$\vec{e}_B = \vec{e}_E \times \vec{e}_N . \quad (2.41)$$

Das strahlbezogene Koordinatensystem entspricht dem Koordinatensystem der Einfallsebene bei Reflexion und Transmission mit dem Unterschied, dass die Einfallsebene diesmal durch den Richtungsvektor der Beugungskante und die Richtung des einfallenden Strahls definiert wird. Das Referenzsystem des gebeugten Strahls ist in gleicher Weise definiert, lediglich die unterschiedliche Strahlrichtung führt zu einer um die Beugungskante gedrehten Ausfallsebene. Der Normalenvektor der Ebene ist gleichzeitig die Bezugsrichtung der Feldkomponente senkrecht zur Einfallsebene

$$\vec{e}_s = \frac{\vec{e}_E \times \vec{e}_d}{|\vec{e}_E \times \vec{e}_d|} . \quad (2.42)$$

Die Bezugsrichtung der Feldkomponente in der Einfallsebene wird durch die Strahlrichtung und den Normalenvektor nach Gleichung (2.23) bestimmt.

Die Berechnung der gebeugten Welle geht genauso vor sich wie die Berechnung der Reflexion oder Transmission. Die einfallende Welle wird in ihre Komponenten senkrecht und parallel zur Einfallsebene zerlegt. Beide Komponenten werden mit den komplexen Beugungskoeffizienten multipliziert und ergeben die entsprechenden Komponenten des abgestrahlten Feldes. Im Gegensatz zu der Reflexion sind jedoch die Beugungskoeffizienten auch von der Azimutrichtung  $\varphi$  des gebeugten Strahls abhängig.

Die skalaren Beugungskoeffizienten resultieren aus der asymptotischen Lösung der Wellengleichung im Hochfrequenzfall ( $\lambda \rightarrow 0$ ) unter Annahme der Dirichletschen und der



Neumannschen Randbedingung (SCHROTH und STEIN, 1985). Sie werden in Anlehnung an die Akustik als *Soft Boundary Condition* ( $\underline{D}_s$ ) und *Hard Boundary Condition* ( $\underline{D}_h$ ) bezeichnet. Dabei wird die Feldstärkekomponente in der Einfallsebene mit  $\underline{D}_s$  und die Komponente senkrecht zur Einfallsebene mit  $\underline{D}_h$  multipliziert. Eine elegantere Formulierung wird im Kontext der Dyadenalgebra möglich, sie ist jedoch auch mit einem höheren Rechenaufwand verbunden. Beide Koeffizienten sind gegeben durch

$$\begin{aligned} \underline{D}_{s,h}(\varphi, \varphi', \beta) = & \frac{-e^{-i\frac{\pi}{4}}}{2\eta\sqrt{2\pi k} \sin \beta} \left\{ \cot \left( \frac{\pi + (\varphi - \varphi')}{2\eta} \right) F(kLa^+(\varphi - \varphi')) \right. \\ & + \cot \left( \frac{\pi - (\varphi - \varphi')}{2\eta} \right) F(kLa^-(\varphi - \varphi')) \\ & + R_{s,h}^0 \cot \left( \frac{\pi - (\varphi + \varphi')}{2\eta} \right) F(kLa^-(\varphi + \varphi')) \\ & \left. + R_{s,h}^0 \cot \left( \frac{\pi + (\varphi + \varphi')}{2\eta} \right) F(kLa^+(\varphi + \varphi')) \right\} . \end{aligned} \quad (2.43)$$

Die Funktion  $F(x)$  erweitert die Koeffizienten der geometrischen Beugungstheorie (GTD) derart, dass die Singularitäten an den Reflexions- und Schattengrenzen kompensiert werden. An diesen Stellen enthalten die Beugungskoeffizienten der vereinheitlichten Beugungstheorie (UTD) zwar Diskontinuitäten, die aber überlagert mit dem direkten Feld für einen kontinuierlichen Übergang des Feldes zwischen dem Bereichen sorgen. Die Funktion  $F(x)$  ist die Lösung des Fresnelschen Übergangsintegrals

$$F(x) = 2i\sqrt{x}e^{ix} \int_{\sqrt{x}}^{\infty} e^{-it^2} dt . \quad (2.44)$$

Das Fresnelintegral wird iterativ gelöst und in der Nähe der Singularitäten durch eine Reihenentwicklung angenähert. Der Entfernungsparameter  $L$  berücksichtigt in seiner allgemeinen Form die Krümmung der Wellenfronten (SCHROTH und STEIN, 1985). Im Fall einer lokal ebenen Welle ist  $L = r \sin^2 \beta$  mit  $r$  der Länge des Ausbreitungspfades. Der Ausdruck  $a^\pm(\varphi \pm \varphi')$  ist ein Maß für den Winkelabstand des Beobachterpunktes von der Reflexions- und Schattengrenze. Er ist definiert als

$$a^\pm(\varphi \pm \varphi') = 2 \cos^2 \left( \frac{2\eta\pi N^\pm - (\varphi \pm \varphi')}{2} \right) . \quad (2.45)$$

Die Zahlen  $N^\pm$  sind die ganzen Zahlen, die am Besten die Gleichungen

$$2\pi\eta N^+ - (\varphi \pm \varphi') = +\pi \quad (2.46)$$

$$2\pi\eta N^- - (\varphi \pm \varphi') = -\pi \quad (2.47)$$

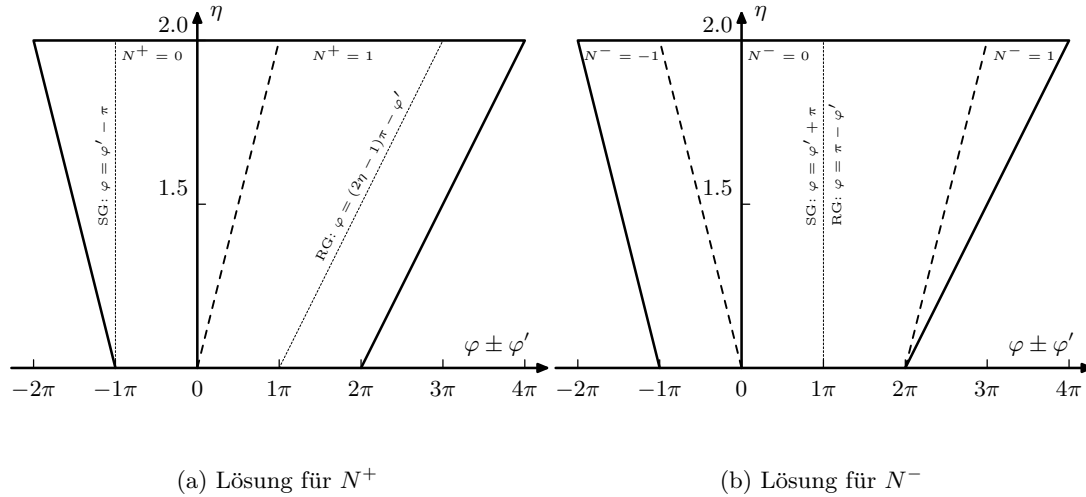
erfüllen. Eine graphische Lösung ist in den Bildern 2.6 dargestellt. Es ist ersichtlich, dass  $N^+$  nur die Zahlen 0 oder +1 und  $N^-$  die Zahlen -1, 0 und +1 annehmen kann.

Die skalaren Beugungskoeffizienten  $\underline{D}_s$  und  $\underline{D}_h$  werden für den Fall einer nicht ideal leitenden Oberfläche der Kante um die Fresnelschen Reflexionskoeffizienten  $\underline{R}_s^0$  und  $\underline{R}_h^0$  der Kantenoberfläche erweitert (LÜBBERS, 1984)

$$\underline{R}_s^0 = \underline{R}_s(\Psi, \epsilon_r) \quad \text{und} \quad \underline{R}_h^0 = \underline{R}_p(\Psi, \epsilon_r) . \quad (2.48)$$

Der Einfallswinkel  $\Psi$  ist der Winkel zwischen dem einfallenden Strahl und dem Normalenvektor der Keilfläche auf der dem Strahl zugewandten Seite (SCHÖBERL, 1997). Die





**Bild 2.6:** Graphische Lösung der Gleichungen (2.46) und (2.47). Ebenfalls eingezeichnet ist die Lage der Reflexionsgrenze (RG) und der Schattengrenze (SG). Nach [Schroth und Stein \(1985\)](#), Seite 279.

Koeffizienten können in einem heuristischen Modell modifiziert werden und die lokale Rauigkeit der Kante berücksichtigen ([LÜBBERS, 1989](#)). Dazu werden beide Koeffizienten mit dem Faktor  $\tilde{\rho}$  erweitert

$$\underline{R}_s^{0,n} = \tilde{\rho} \underline{R}_s(\Psi, \epsilon_r) \quad \text{und} \quad \underline{R}_h^{0,n} = \tilde{\rho} \underline{R}_p(\Psi, \epsilon_r) . \quad (2.49)$$

Der Faktor  $\tilde{\rho}$  ergibt sich zu

$$\tilde{\rho}^2 = e^{-2\delta} \quad \text{mit} \quad \delta = \frac{4\pi\Delta h}{\lambda} \sin \Psi . \quad (2.50)$$

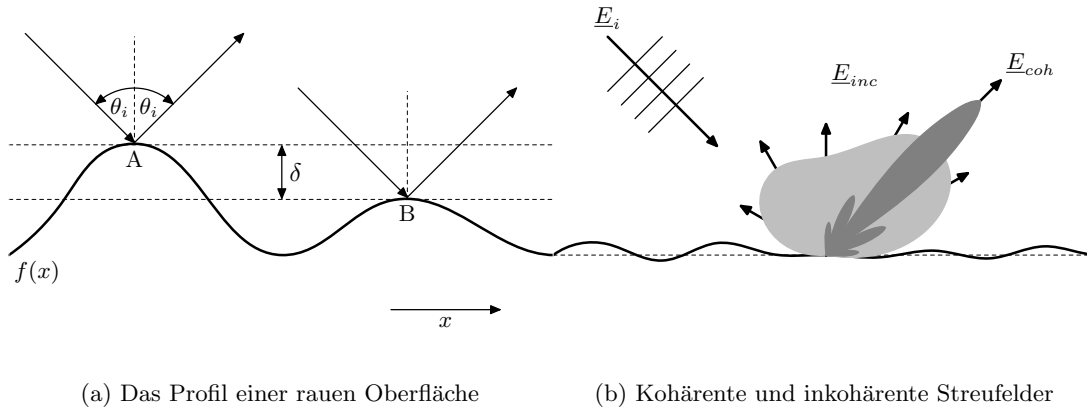
Die Länge  $\Delta h$  stellt die Standardabweichung der als normalverteilt angenommenen lokalen Rauigkeit der Keiloberfläche dar.

## 2.6 Streuung an rauen Oberflächen

Die Streuung der elektromagnetischen Wellen an rauen Oberflächen gehört ebenfalls zu den wichtigsten Effekten, die bei der Funknetzplanung berücksichtigt werden müssen. Der Einfluss der Streuung nimmt bei höheren Frequenzen zu, während der Beitrag der Kantenbeugung zum Gesamtfeld mit steigender Frequenz abnimmt. Problematisch bei der Berücksichtigung der Streuung ist neben der analytischen oder numerischen Modellierung des Effekts vor allem die fehlende Kenntnis der charakteristischen Größen der streuenden Oberfläche. Diese Größen, wie die Rauigkeit oder die elektrischen Parameter der Oberfläche, sind selten genau bekannt und können sich oft stark örtlich ändern. Sie müssen messtechnisch gewonnen werden.

Bei der Bestimmung der Streumatrizen der Reflexion, der Brechung und der Transmission durch eine geschichtete Wand im Abschnitt 2.4 wurde implizit eine ideal glatte Oberfläche bzw. Grenzschicht zwischen den unterschiedlichen Medien vorausgesetzt. Diese Annahme ist nur für Flächen mit einer hinreichen kleinen Oberflächenrauigkeit erfüllt. Die Grenze für den Übergang zwischen einer glatten und einer rauen Oberfläche ist durch

das Rayleigh- und das Fraunhofer-Kriterium gegeben. Beide Kriterien gehen von dem selben Ansatz aus, unterscheiden sich jedoch in der Definition der Grenze für die erlaubte Phasendifferenz. Der Ansatz wird anhand der im Bild 2.7a dargestellten rauen Oberfläche mit der Höhe  $f(x)$  erläutert. Die Phasendifferenz der am Punkt A und der am Punkt B



**Bild 2.7:** Zur Definition der Streuung an einer rauen Oberfläche. Das Kriterium für die Einstufung einer Oberfläche als rau basiert auf der maximalen mittleren Phasendifferenz der gestreuten Wellen. Das Streufeld besteht aus einer Überlagerung kohärenter und diffuser Feldanteile.

gestreuten Welle ergibt sich zu

$$\Delta\phi = 2k_0\delta \cos\theta_i \quad . \quad (2.51)$$

Das Profil der Oberfläche ist eine zufällige Größe und somit durch einer stochastische Funktion  $f(x)$  beschrieben. Die Schwankung der Höhe  $\delta$  ist also ebenfalls eine stochastische Variable, daher lassen sich nur Quadratmittel von  $\Delta\phi$  angeben:

$$\sqrt{\overline{\phi^2}} = 2k_0\sqrt{\overline{\delta^2}} \cos\theta_i = 2k_0\sigma \cos\theta_i \quad . \quad (2.52)$$

Bei einer ideal glatten Oberfläche ist die Standardabweichung der Oberflächenrauigkeit  $\sigma$  und damit auch  $\Delta\phi$  gleich Null. Bei einer rauen Oberfläche muss ein Maximalwert für das Quadratmittel von  $\Delta\phi$  festgelegt werden, so dass sich die Wellenanteile im Fernfeld noch konstruktiv überlagern können. Das Rayleigh-Kriterium geht von einem Grenzwert für den mittleren Phasenfehler von  $\pi/2$  aus. Das Fraunhofer-Kriterium setzt stattdessen für den Grenzwert  $\pi/8$  ein. Damit ergeben sich die folgenden Grenzen für die Standardabweichung der Oberflächenrauigkeit  $\sigma$ :

$$\sigma < \frac{\lambda_0}{8 \cos\theta_i} \quad (\text{Rayleigh}) \quad \text{bzw.} \quad \sigma < \frac{\lambda_0}{32 \cos\theta_i} \quad (\text{Fraunhofer}) \quad . \quad (2.53)$$

Für den praktischen Einsatz sollte das Kriterium nach Fraunhofer gewählt werden, da nur dieses mit experimentellen Ergebnissen konsistent ist (GENG und WIESBECK, 1998). Ist das Kriterium erfüllt, so kann die Oberfläche als glatt angenommen werden.

Im Fall einer rauen Oberfläche besteht das Streufeld aus kohärenten und diffusen Anteilen (Bild 2.7b). Das Verhältnis der kohärent abgestrahlten Energie zu der inkohärent abgestrahlten Energie ist im Wesentlichen durch die Beschaffenheit der Oberfläche bestimmt. Ganz allgemein lässt sich der Streuprozess durch eine komplexe polarimetrische Streumatrix beschreiben:

$$\begin{bmatrix} \underline{E}_v \\ \underline{E}_h \end{bmatrix}_{scat} = \frac{e^{-ik_0 r}}{r} \begin{bmatrix} \underline{S}_{vv} & \underline{S}_{vh} \\ \underline{S}_{hv} & \underline{S}_{hh} \end{bmatrix} \cdot \begin{bmatrix} \underline{E}_v \\ \underline{E}_h \end{bmatrix}_{inc}. \quad (2.54)$$

Diese Matrix beschreibt den Zusammenhang zwischen der einfallenden ebenen Welle, bestehend aus einer horizontalen und einer vertikalen Feldkomponente, und der abgestrahlten sphärischen Welle, die ihrerseits auch beide Polarisationskomponenten besitzen kann. Das sphärische Koordinatensystem ist durch die Lage des Normalenvektors und der Tangentialvektoren an die Oberfläche im Auftreffpunkt des Strahls definiert. Das assoziierte kartesische Koordinatensystem dient gleichzeitig als das Bezugssystem für die Definition der Oberflächeneigenschaften.

Der Mittelwert der Quadrate jeder der vier Streumatrixkomponenten  $\underline{S}_{pq}$  lässt sich in einen kohärenten und einen inkohärenten Anteil zerlegen

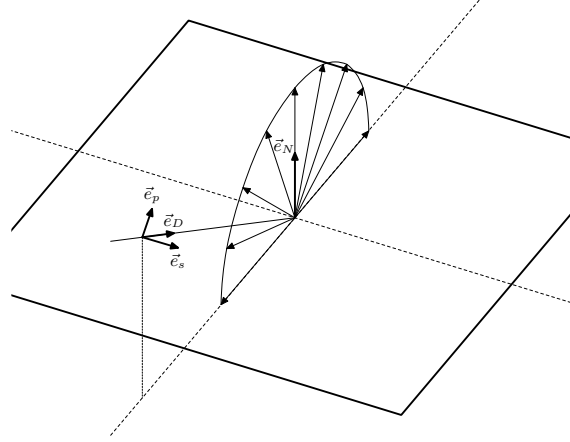
$$\overline{|\underline{S}_{pq}|^2} = |\underline{S}_{pq}^{coh}|^2 + \overline{|\underline{S}_{pq}^{inc}|^2}. \quad (2.55)$$

Sowohl für den kohärenten als auch für den diffusen Anteil existieren analytische Lösungsverfahren. Für schwach raue Oberflächen führt die Kirchhoff-Theorie mit skalarer Approximation zu einer Beschreibung des kohärenten Anteils. Die Oberfläche wird durch Tangentialebenen approximiert und muss somit lokal Krümmungsradien aufweisen, die deutlich größer als die Wellenlänge sind. Der inkohärente Anteil wird durch die Anwendung der Kirchhoff-Theorie mit stationärer Phasenapproximation gewonnen. Bedingung hierfür ist im Gegensatz zu der skalaren Approximation eine sehr große Oberflächenrauigkeit. Ein weiteres analytisches Verfahren ist die *Small Perturbation Method* (SPM), die keine Näherung mit Tangentialebenen voraussetzt und somit keine Einschränkung bezüglich des lokalen Krümmungsradius enthält. Sie ist jedoch nur auf näherungsweise planare Oberfläche mit kleinen Höhendifferenzen  $\delta$  anwendbar.

Im Fall einer schwach rauen Oberfläche lässt sich die Streuung näherungsweise gut durch eine einzige kohärente Reflexion beschreiben, da die diffusen Feldanteile gering sind. Die durch die diffuse Reflexion abgestrahlte Energie wird durch die modifizierten Fresnelschen Reflexionskoeffizienten berücksichtigt

$$\underline{R}_{p,s}^{mod} = \underline{R}_{p,s} e^{-2\left(2\pi \frac{\sigma}{\lambda_0} \cos \theta_i\right)^2}. \quad (2.56)$$

Die individuellen Gültigkeitsbereiche der analytischen Methoden schränken ihre Anwendbarkeit vielfach stark ein. Sehr viel universeller kann die Streumatrix mit Hilfe aufwendiger numerischer Verfahren bestimmt werden. Diese basieren oft auf der Lösung der Feldintegralgleichung mit der Momentenmethode für ein zufällig generiertes 2D Profil der Oberfläche. Die Momentenmethode erfordert die sehr rechenaufwendige Lösung eines vollbesetzten linearen Gleichungssystems (HARRINGTON, 1982) und ist innerhalb eines schnellen strahlenoptischen Verfahrens in dieser Form kaum einsetzbar. Aus diesem Grund wurde in dieser Arbeit lediglich ein eindimensionales Modell einer streuenden Oberfläche implementiert. Das physikalische Modell basiert auf der Bestimmung der bi-statischen Streukoeffizienten einer eindimensionalen bandbegrenzten fraktalen Oberfläche mit der Momentenmethode nach TSANG et al. (2001). Das geometrische Modell grenzt die Menge der gestreuten Strahlen auf die Schar der Strahlen in der Einfallsebene ein, um



**Bild 2.8:** Die Definition der Geometrie und des Koordinatensystems der Streuquelle.

die Rechenzeit noch weiter zu verkürzen. Die Geometrie zusammen mit dem eingesetzten Koordinatensystem ist im Bild 2.8 dargestellt.

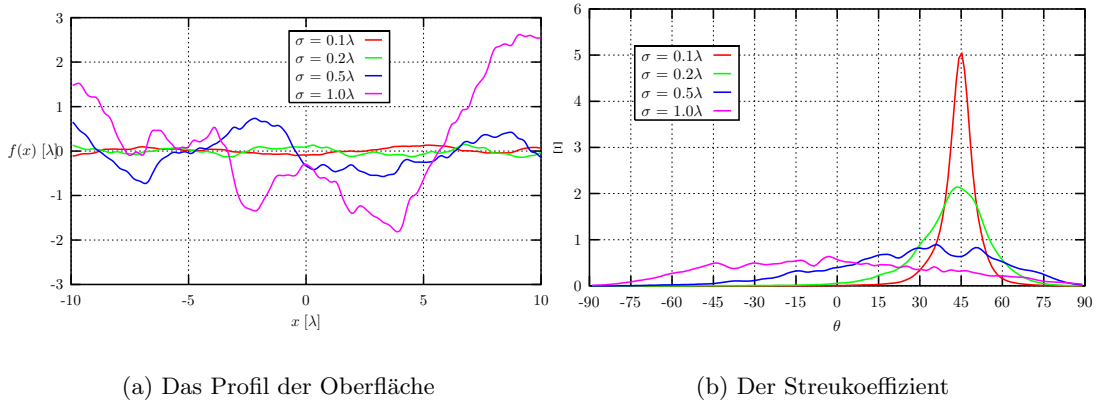
Das Profil der fraktalen Oberfläche wird mit Hilfe der Weierstraß-Mandelbrot Funktion generiert. Diese Funktion ist in ihrer allgemeinen Form durch die unendliche Reihe

$$f(x) = \sum_{k=1}^{\infty} \frac{\sin(\pi k^2 x)}{\pi k^2} \quad (2.57)$$

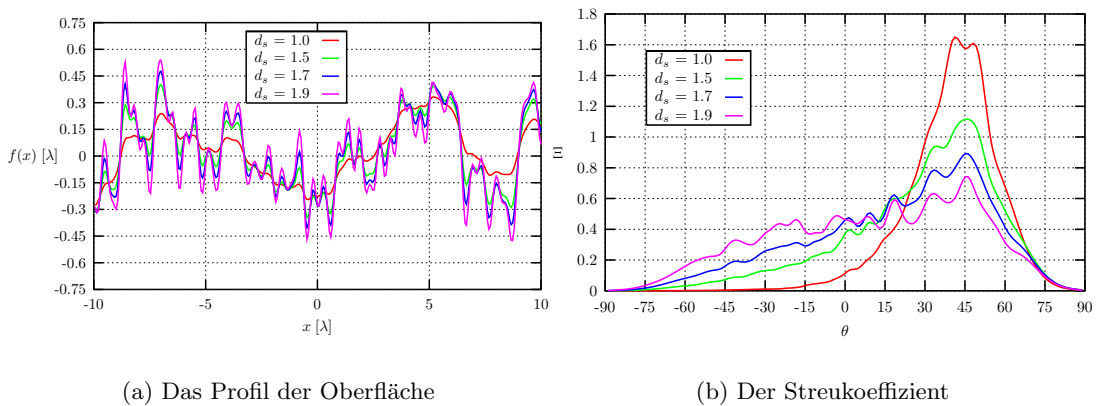
definiert (WEISSTEIN, 2002). Die Funktion ist stetig aber nur für bestimmte Punkte differenzierbar. Die bandbegrenzte Funktion der Oberfläche mit der fraktalen Dimension  $d_s$ , der Standardabweichung der Oberflächenrauigkeit  $\sigma$  und dem Verhältnis der Wellenzahlen an den Bandgrenzen  $\alpha = k_u/k_l$  ist durch

$$f(x) = \sigma \sqrt{2 \frac{1 - \alpha^{2 \frac{1}{N-1}(d_s-2)}}{1 - \alpha^{2 \frac{N}{N-1}(d_s-2)}}} \sum_{n=0}^{N-1} \sin\left(x k_l \alpha^{\frac{n}{N-1}} + \phi_n\right) \alpha^{\frac{n}{N-1}(d_s-2)} \quad (2.58)$$

definiert (TSANG et al., 2001). Die Phase  $\phi_n$  ist normalverteilt mit dem Mittelwert 0 und der Standardabweichung  $2\pi$ . Die ganze Zahl  $N$  gibt die Anzahl der zu berücksichtigenden Frequenzen der endlichen Reihe an. Ein so generiertes Oberflächenprofil für verschiedene Oberflächenrauigkeiten  $\sigma$  ist im Bild 2.9a dargestellt. Die berechneten bistatischen Streukoeffizienten einer Oberfläche der Länge  $20\lambda$  bei  $f = 10.5$  GHz sind im Bild 2.9b zu sehen. Die Oberfläche wird von einer planaren Welle unter einem Winkel von 45 Grad beleuchtet. Mit steigender Oberflächenrauigkeit wird die reflektierte Leistung zunehmend über einen weiten Winkelbereich verteilt. Die fraktale Dimension der Oberfläche ist ein Maß für die Selbstähnlichkeit des Profils und bestimmt somit die lokale Beschaffenheit der Oberfläche. Ein Vergleich der Oberflächenprofile für verschiedene Werte der fraktalen Dimension ist im Bild 2.10a dargestellt. Mit der zunehmenden fraktalen Dimension  $d_s$  nehmen die Amplituden der höherfrequenten Anteile im Spektrum der Funktion zu, die Oberfläche wird lokal rauer. Durch die Bandbegrenzung der Funktion auf den Bereich  $k \in [10,500]$  und die Berücksichtigung von 100 Frequenzen aus diesem Bereich steigt die lokale Rauigkeit des Profils bei hohen Werten von  $d_s$  nur noch geringfügig an. Die Auswirkung der fraktalen Dimension der rauen Oberfläche auf die bistatischen Streukoeffizienten  $\Xi$  ist aus dem Bild 2.10b ersichtlich. Durch die steigende lokale Rauigkeit der Oberfläche steigt ebenfalls



**Bild 2.9:** Das eindimensionale Profil und die Lösung der bistatischen Streukoeffizienten der Oberfläche mit der Momentenmethode nach Tsang et al. (2001).



**Bild 2.10:** Die Abhängigkeit der bistatischen Streukoeffizienten einer fraktalen Fläche von der Variation der fraktalen Dimension.

der diffuse Anteil des Streufeldes. Für die Berechnung der bistatischen Koeffizienten wurden jeweils 100 verschiedene Oberflächen mit den angegebenen Eigenschaften berechnet und der Mittelwert der einzelnen Streukoeffizienten bestimmt.

## 2.7 Anwendungsbereich des Modells

Die Vorhersagegenauigkeit einer Simulation wird durch verschiedene Faktoren bestimmt. Der Einfluss dieser Faktoren auf das Ergebnis der Berechnung entscheidet darüber, ob die Genauigkeit der Simulation für den anvisierten Einsatzbereich als ausreichend anzusehen ist. Die Grenzen eines Modells müssen durch den Vergleich der Simulation mit analytischen Lösungen, Ergebnissen anderer numerischer Verfahren oder Messungen ausgelotet werden. Im Folgenden wird auf einige Faktoren näher eingegangen.

Die Anforderungen an die Genauigkeit bei der Eingabe der Geometrie und der Materialparameter sollten stets besonders hoch gesteckt sein, denn die Einschränkungen der Genauigkeit einer Simulation beginnen oft bereits bei der Modellierung der Geometrie eines Szenario. Die Genauigkeit, mit der das Szenario modelliert werden kann, ist jedoch oft

durch das Fehlen von Plänen in elektronischer Form stark eingeschränkt. Auch wenn ein detaillierter Plan vorliegt, ist es oft nicht möglich, und manchmal aufgrund der höheren Rechenzeiten auch nicht wünschenswert, alle Einzelheiten der Geometrie im Modell zu erfassen. Die absichtliche Einschränkung der Detailtreue ist somit auch eine Quelle der Ungenauigkeit einer späteren Simulation. Der Einfluss einer solchen Vereinfachung der Geometrie sollte stets mit Simulationen bzw. Messungen abgeschätzt werden. Desweiteren sind die elektrischen Parameter der Materialien im gegebene Frequenzbereich oft nicht bekannt und müssen, wenn überhaupt möglich, durch Messungen ermittelt werden.

Ein weiterer wichtiger Faktor ist durch das verwendete physikalische und strahlenoptische Modell gegeben. Die Vernachlässigung von Effekten, wie zum Beispiel der Beugung, führt in manchen Fällen zu signifikanten Abweichungen der Vorhersage von den gemessenen Werten (GENG und WIESBECK, 1998). Da aufgrund der langen Rechenzeiten absichtlich die Anzahl der zu berücksichtigten Effekte und Interaktionen der Strahlen im Szenario reduziert werden muss, entstehen so weitere Abweichungen der gerechneten Ergebnisse von der Realität. Gleichzeitig darf nicht vergessen werden, dass das strahlenoptische Modell an sich nur eine Approximation der physikalischen Verhältnisse im Hochfrequenzfall darstellt und nur unter dieser Voraussetzung gültig ist. Auch hier gilt, dass die Wahl der Simulationsparameter und ihr Einfluss auf die Ergebnisse mit Messungen bestimmt werden muss.

Ebenso hat die Wahl der Geometriedarstellung, entweder als ein einfaches Polygonmodell in der *Boundary Representation* (BREP) Darstellung oder als ein komplexes *Combinatorial Solid Geometry* (CSG) Modell, Einfluss auf den Anwendungsbereich der Verfahrens. Sie hat jedoch gleichermaßen einen entscheidenden Einfluss auf die Rechengeschwindigkeit. Die Wahl der geometrischen Darstellung ist stets als ein Kompromiss zwischen Detailgenauigkeit und Rechenzeit anzusehen. Die komplexen und großflächigen Geometrien im Bereich der Funknetzplanung und Ausbreitungsprognose lassen sich bis zu einer ausreichenden Genauigkeit durch polygonale Modelle beschreiben. Weitere Einschränkungen bei der Definition und der Berechnung von Wänden, die aus mehreren Schichten aufgebaut sind, wurden bereits im Abschnitt 2.4 behandelt.

Die Beschränkung des Beugungsmodells auf die Kantenbeugung erfasst zwar diesen wichtigsten Ausbreitungsmechanismus in Schattenbereiche. Es werden jedoch Beugungseffekte an anderen komplizierteren Formen vernachlässigt. Durch das Fehlen gekrümmter Oberflächen wird die Ausbreitung und Ablösung von Oberflächenwellen implizit ausgeschlossen. Das verwendete Modell der Streuung an rauen Oberflächen stellt somit einen weiteren Kompromiss zwischen der physikalischen Treue und der benötigten Rechenzeit dar. Die Beschränkung auf ein eindimensionales Modell mit einer reduzierten Strahendichte bildet die Streuung in der Einfallsebene richtig nach, vernachlässigt jedoch die azimutale Abhängigkeit der Streuquelle. Ein volumetrisches Streumodell wird zur Zeit ebenfalls nicht berücksichtigt.

Die Genauigkeit einer Simulation ist aber auch durch andere, nicht deterministische Faktoren bestimmt. So stellt die natürliche Präsenz von Menschen, Autos und anderen beweglichen Gegenständen innerhalb eines Funkszenario eine zufallsbestimmte Größe dar. Eine aussagekräftige Betrachtung von solchen dynamischen und nicht deterministischen Szenarien ist letztlich nur durch statistische Methoden möglich.

## 2.8 Zusammenfassung

Dieses Kapitel behandelte die physikalischen Grundlagen dieser Arbeit. Ausgehend von der asymptotischen Näherungslösung der Wellengleichung im Hochfrequenzfall wurde die polarimetrische Übertragungsfunktion eingeführt und die zugehörigen Koordinatentrans-

formationen behandelt. Unter den physikalischen Effekten, die als kanonischen Fälle bezeichnet und unabhängig voneinander behandelt werden, wurden in diesem Kapitel die Reflexion an und die Transmission durch mehrschichtige Wände, die Beugung an Kanten und die Streuung an fraktalen rauen Oberflächen behandelt. Abschließend wurden verschiedene Faktoren genannt, die sich auf die Genauigkeit des Verfahrens auswirken und eine Einschränkung bezüglich der Anwendbarkeit der Methode darstellen.





---

## Algorithmische Grundlagen

Die geometrisch-optische Näherung der Wellengleichung aus Kapitel 2 stellt die Basis für eine Vielzahl von Algorithmen zur Lösung des Wellenausbreitungsproblems dar. Ihre Anwendungen reichen von der klassischen Wellenausbreitung in der Elektrotechnik über die Simulation der akustischen Eigenschaften von Räumen in der Akustik bis hin zur Berechnung der Ausbreitung von Schockwellen eines Erdbebens im Erdmantel. Ein besonders fruchtbares Forschungsgebiet für *Ray Tracing Algorithmen* ist die Computergraphik, bedingt durch die Forderung nach der schnellen Berechnung komplexer photorealistischer Animationen. Viele der dort entwickelten Algorithmen gehören heute zu den effizientesten und schnellsten bekannten Verfahren.

Dieses Kapitel beschäftigt sich mit den algorithmischen Fragen der strahlenoptischen Lösung des Wellenausbreitungsproblems. Anhand der bereits bekannten Verfahren werden Schlüsselfunktionen bei der Implementierung eines klassischen Ray Tracing Algorithmus identifiziert. Die so gewonnenen Erkenntnisse werden in den folgenden Kapiteln benötigt, um ein schnelles Verfahren zur Lösung des strahlenoptischen Problems zu erarbeiten.

### 3.1 Monte Carlo Ray Tracing Verfahren

Eine wichtige Klasse der eingesetzten Ray Tracing Verfahren basiert auf dem Teilchenmodell des Lichts. In diesem Modell werden von der Strahlenquelle Photonen zufällig in alle Richtungen emittiert, ähnlich wie dies bei dem Zerfall radioaktiver Nuklide der Fall ist. Typisch für diese Methode ist die Tatsache, dass jedes der  $N$  Photonen, den  $N$ -ten Teil der Gesamtenergie der Quelle trägt. Durch die geeignete Wahl einer Wahrscheinlichkeitsdichtefunktion beim Sender kann eine beliebige Richtcharakteristik einer Antenne nachgebildet werden. Ebenfalls können auch spektrale und polarisierende Eigenschaften der Quelle berücksichtigt werden. Der Weg eines jeden Photons wird anschließend durch das Szenario verfolgt. Interaktionen des Photons mit der Umgebung werden in der Form von Wahrscheinlichkeitsfunktionen angegeben. Auf diese Weise ist es möglich, alle im Kapitel 2 angegebenen Effekte zu berücksichtigen. Empfänger werden meistens in der Form von Detektionskugeln oder -voxeln, seltener als Empfangsflächen modelliert. Stochastisches Ray Tracing entspricht der Lösung der Gleichung (2.1) mit der *Monte Carlo* Methode.

Der wichtigste Nachteil dieser Methode bezüglich ihres Einsatzes im Bereich der Funknetzplanung in ausgedehnten offenen Szenarien liegt in der Zahl der Teilchen, die benötigt wird, um statistisch relevante Aussagen treffen zu können. Durch die Divergenz der Strahlen sinkt die Photonendichte im freien Raum proportional mit  $1/r^2$ . Dadurch detektieren entfernte Empfänger weit weniger Photonen als die dem Sender nähergelegene Empfänger. Somit sind weit entfernte Empfänger mit größeren statistischen Unsicherheiten versehen. Eine ausreichende Statistik kann nur durch eine Erhöhung der Photonenzahl erreicht werden, was wiederum zu einer entsprechenden Erhöhung der Rechenzeit führt. Ein weiterer Nachteil dieser Methode besteht in der Natur der Detektion der Photonen. Durch die endliche Ausdehnung des Detektors wird die Phase der Photonen örtlich verschmiert, was zu einer entsprechenden Verzerrung der polarimetrischen Übertragungseigenschaften des Kanals führt. Eine Verkleinerung der Detektorabmessungen hilft zwar, diese Verschmierung

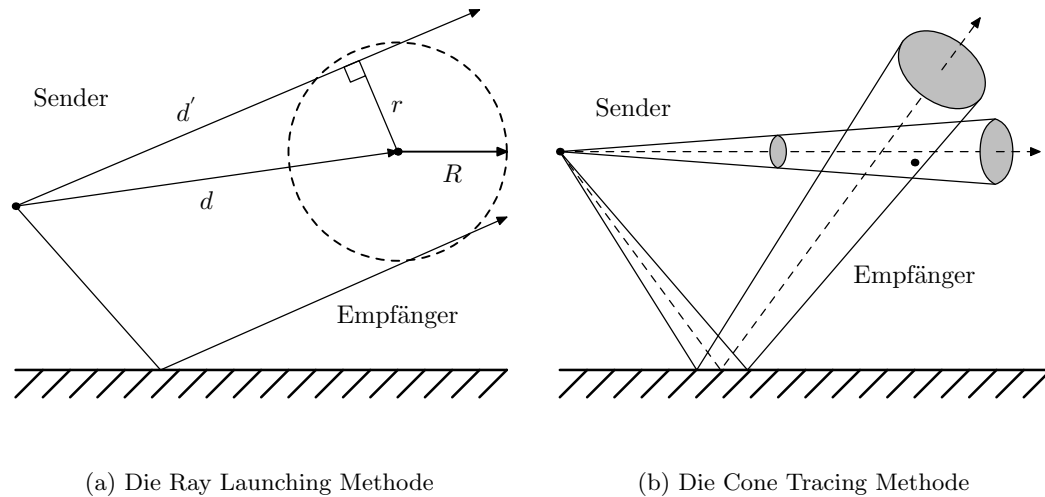
der Phaseninformation in Grenzen zu halten, führt wiederum zu einer Verschlechterung der Statistik. Monte Carlo Ray Tracing Verfahren werden häufig im Bereich der Hochenergiephysik eingesetzt, da sie die statistische Natur der Wechselwirkung der Materie mit energiereichen Photonen ideal nachbilden.

## 3.2 Deterministische Ray Tracing Verfahren

Da die Rechenzeit mit der Anzahl der Strahlen steigt, ist eine Reduktion der benötigten Strahlendichte auf ein Minimum wünschenswert. Eine Lösung bietet die deterministische Variante der Ray Tracing Methode, bei der die Teilchen durch lokal ebene Wellenfronten ersetzt werden. Diese Methode kann dann als eine globale, dreidimensionale Suche nach möglichen Ausbreitungswegen  $C$  der Wellen angesehen werden, die die Position des Senders mit der Position des Empfängers verbinden und für die die Gleichung (2.1) minimal wird. Die Menge der so gefundenen Ausbreitungswege stellt die Impulsantwort des Kommunikationskanals zwischen den Orten des Senders und des Empfängers dar. Diese kann anschließend mit der Antennencharakteristik des Senders und des Empfängers gewichtet werden um die Klemmengrößen und andere Systemparameter zu erhalten. Häufig ist es erwünscht, die Kanaleigenschaften ohne die Antennencharakteristik erfassen zu können, um zum Beispiel die Polarisations-eigenschaften des Raumes untersuchen zu können oder verschiedene Antennenkonfigurationen vergleichen zu können, ohne eine erneute Simulation durchführen zu müssen. Bei einer isotropen Strahlenquelle beschreibt die Kanalimpulsantwort diese Eigenschaften in der allgemeinsten Form. Bei fest vorgegebenen Antennen können andererseits die Antenneneigenschaften auch genutzt werden, um die Simulationszeiten zu verkürzen, indem die Strahlendichte der Quelle der Antennencharakteristik angepasst wird. Diese Eigenschaften machen deterministische Ray Tracing Verfahren zu der Methode der Wahl für die schnelle Berechnung der Wellenausbreitung in komplexen Szenarien. Im Folgenden werden einige deterministische Ansätze vorgestellt und ihre Vor- und Nachteile diskutiert.

### 3.2.1 Die Strahlabschussmethode

Die klassische Lösung bietet die *Strahlabschussmethode*. Bei dieser wird der gesamte Raum mit linienförmigen Strahlen gleichmäßig nach neuen Ausbreitungswegen abgetastet. Die Empfänger werden wie bei den Monte Carlo Verfahren – mit allen Nachteilen – als ausgedehnte, meistens kugelförmige, Objekte implementiert (Bild 3.1a). Die Größe des Empfängervolumens kann konstant gehalten oder an die am Ort des Empfängers zu erwartete Strahlendichte angepasst werden. Die Gründe für diese Anpassung liegen in der Wahrung einer entfernungsunabhängigen Trefferwahrscheinlichkeit jedes Empfängers. Eine Abschätzung für den Radius der Empfängerkugel ist in (SCHÖBERL, 1997) zu finden. Der Einsatz ausgedehnter Empfänger erschwert die Verwendung dieser Methode mit hierarchischen Partitionierungsverfahren, wie sie im folgenden Kapitel 4 vorgestellt werden, da ausgedehnte Empfänger auf beiden Seiten einer Partitionierungsfläche liegen können. Außerdem können solche angepassten Empfänger bei einer entsprechenden Entfernung zum Sender die Abmessungen ganzer Räume annehmen und lassen somit keine Rückschlüsse auf eventuelle Abschattung oder die genaue Phaseninformation des Ausbreitungsweges zu.



(a) Die Ray Launching Methode

(b) Die Cone Tracing Methode

**Bild 3.1:** Zwei Varianten der Strahlabschussmethode. Das klassische Ray-Launching Verfahren setzt kugelförmige Empfänger und linienförmige Strahlen ein. Im Gegensatz dazu werden bei der Cone Tracing Methode punktförmige Empfänger und kegelförmige Strahlen verwendet. Die Cone Tracing Methode ist besser für eine Kombination mit hierarchischen Beschleunigungsverfahren geeignet, da eine eindeutige Zuordnung der punktförmigen Empfänger zu den Teilräumen der Hierarchie besteht (Kapitel 4).

### 3.2.2 Die Cone Tracing Methode

Eine andere Betrachtungsweise der Strahlabschussmethode stellt die *Cone Tracing* Methode dar (Bild 3.1b). Bei der *Cone Tracing* Methode werden punktförmige Empfänger eingesetzt, die von konischen Strahlen getroffen werden. Die Strahlen werden durch ihre Achse und ihren Öffnungswinkel beschrieben. Für die Strahlverfolgung werden ausschließlich die Achsen der Koni verwendet. Ein Empfänger wird dann von einem Strahl getroffen, wenn er innerhalb des Strahlkonus liegt. Punktförmige Empfänger können vorteilhaft mit hierarchischen Verfahren zur Strahlwegsuche eingesetzt werden. Das *Cone Tracing* Verfahren wurde von AMANATIDES (1984) in der Computergraphik zu einem adaptiven Verfahren entwickelt. Der Öffnungswinkel der konischen Strahlen ist eine Funktion der Strahlendichte und des verwendeten Algorithmus zur räumlichen Abtastung. Durch die Abschattung der Strahlen an Kanten kommt es zu *Aliasing Effekten*, die durch eine adaptive Anpassung der Strahlendichte an die Geometrie behoben werden können. Auf die Methoden der konstanten und der adaptiven Strahlengenerierung und die damit verbundene Berechnung der Öffnungswinkel wird im Kapitel 6 näher eingegangen. Sie beeinflussen entscheidend die Effizienz und die Genauigkeit des gesamten Verfahrens.

### 3.2.3 Die Spiegelungsmethode

Die *Spiegelungsmethode* ersetzt die Geometrie des Szenario durch eine Anzahl geeignet gewählter Spiegelquellen (ALLEN und BERKLEY, 1979). Jede Spiegelquelle repräsentiert hierbei einen eindeutigen Ausbreitungsweg vom Sender zum Empfänger. Eine typische Eigenschaft dieser Methode ist die genaue Berechnung des Ausbreitungsweges  $C$  zwischen der Position des Senders und dem punktförmigen Empfänger, so dass alle gefundenen Strahlenwege beide Orte exakt verbinden. Dies war bei den obengenannten Methoden

nicht der Fall. Dadurch können die genauen elektrischen Parameter der Wellenfront am Ort des Empfängers berechnet werden. Diese Eigenschaft ist bei der Berechnung des schnellen Schwunds bei der Funknetzplanung oder bei der Auralisierung in der Akustik von entscheidender Bedeutung. Der wichtigste Nachteil der Spiegelungsmethode liegt in dem mit der Anzahl der Spiegelquellen exponentiell ansteigenden Rechen- und Speicheraufwand. Dieser Nachteil kann jedoch durch eine Kombination der Spiegelungsmethode mit der Strahlabschussmethode behoben werden (VORLÄNDER, 1989).

### 3.3 Der Ray Tracing Kernel

In diesem Abschnitt wird der grundlegende Ray Tracing Algorithmus vorgestellt und analysiert. Hierbei wird zunächst auf die Modellierung von Effekten wie der Beugung und der Streuung, die durch das Einfügen von Sekundärquellen in das Szenario nachgebildet werden, verzichtet. Auf die Implementierung beider Effekte wird im Kapitel 6 genauer eingegangen.

Die von dem Sender ausgesandten Strahlen bilden die Wurzel eines *Strahlenbaumes*. Dabei stellt jede Verzweigung des Baumes die Interaktion eines Teilstrahls mit der Geometrie dar. Durch diese Interaktion entstehen mehrere neue Teilstrahlen, die zu weiteren Verzweigungen des Astes führen. Sollen  $N$  aufeinanderfolgende Interaktionen des Strahls mit der Geometrie betrachtet werden, so besitzt der binäre Strahlenbaum die Höhe  $N$ . Die maximale Anzahl der Interaktion in einem balancierten Baum ist gleich der Gesamtzahl der Knoten, also  $2^N - 1$ . Die Form des Strahlenbaumes wird weitgehend durch die Position und Orientierung des ersten Strahls im Szenario bestimmt, so dass im Allgemeinen der Strahlenbaum nicht balanciert ist.

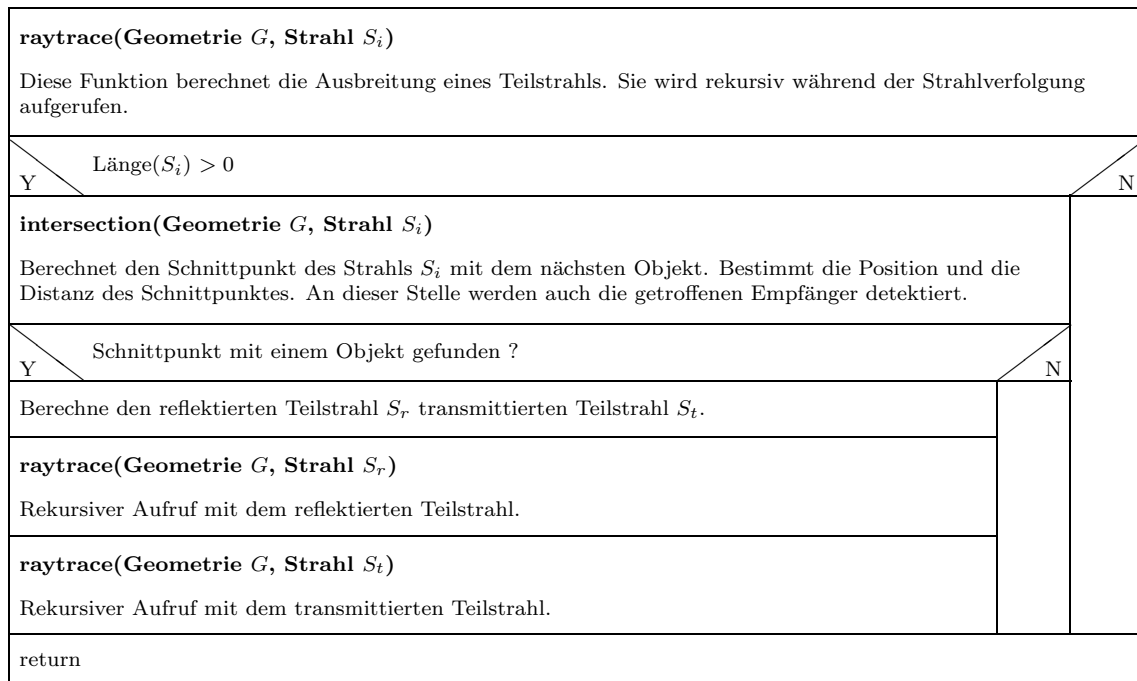
Die vereinfachte Struktur des rekursiven Ray Tracing Algorithmus ist im Diagramm 3.2 dargestellt. Der Strahl wird durch seine Position und Richtung sowie den Sichtbarkeitsbereich bestimmt. Dieser definiert die maximale Strahllänge in Abhängigkeit der von dem Anwender gewünschten Dynamik der Simulation. Da die Strahllänge die Anzahl der Interaktionen mit der Geometrie bestimmt, ist ihre Einschränkung für die Rechenzeit relevant. Eine konservative Annahme für die Strahllänge kann der einfachen Übertragungsgleichung im Freiraum entnommen werden. Nach GROSSKOPF (1970) ergibt sich die Übertragungsdämpfung einer Freiraumstrecke der Länge  $d$  bei der Frequenz  $f$  zu :

$$A/[dB] = 32.5 + 20 \log(f/[MHz]) + 20 \log(d/[km]) - 10 \log(G_t) - 10 \log(G_r) . \quad (3.1)$$

Die Größen  $G_t$  und  $G_r$  sind dabei die Gewinne der Sende- und Empfangsantennen. Aus dieser Gleichung kann unter Berücksichtigung der Sendeleistung und der Empfängerempfindlichkeit die minimale geforderte Strahllänge abgeschätzt werden. Die tatsächliche Strahllänge ist dabei kürzer, bedingt durch die Verluste aufgrund der Streuprozesse des Strahls an der Geometrie. Auch wenn diese Verluste während der Berechnung berücksichtigt werden könnten, werden sie aus Rechenzeitgründen meistens vernachlässigt. Der im Diagramm 3.2 dargestellte Ray Tracing Algorithmus stellt eine einfache rekursive Implementierung dar<sup>1</sup>. Er ermöglicht die Identifikation der für das Ray Tracing zeitbestimmende Funktion der Schnittpunktberechnung. Diese Funktion muss im Fall eines balancierten Strahlenbaums einmal für jeden Teilstrahl aufgerufen werden. Da ein solcher Baum aus  $2^{N+1} - 1$  Teilstrahlen besteht, ist eine effiziente Implementierung des *Schnittpunkttests* für die Geschwindigkeit des gesamten Ray Tracing Verfahrens von sehr großen Bedeutung. Im

<sup>1</sup>Eine iterative Lösung dieser Funktion ist ebenfalls möglich und sogar effizienter, da dadurch viele Stack-Operationen vermieden werden. Aus Gründen der besseren Übersichtlichkeit wird hier auf die Darstellung der iterativen Lösung verzichtet.

## Ray Tracing Algorithmus — Kernel Loop



**Bild 3.2:** Das Beispiel einer Implementierung der Kernfunktion in der inneren Ray Tracing Schleife. Diese Funktion wird rekursiv aufgerufen, um den gesamten Strahlenbaum zu durchqueren.

nächsten Abschnitt wird auf einige Verfahren der Schnittpunktberechnung eingegangen. Viele der Verfahren wurden im Bereich der Computergraphik entwickelt und klassifiziert, sie sind jedoch universell einsetzbar. Untersuchungen haben ergeben, dass ihre Effizienz weitgehend von der Struktur der Geometrie abhängig ist (ARVO, 1990; KIRK und ARVO, 1988; SUBRAMANIAN und FUSSELL, 1990). Die in Folge dessen entwickelten hybriden Verfahren führten zu den im Abschnitt 3.5 vorgestellten Meta-Hierarchien.

### 3.4 Ray Tracing Beschleuniger

Eine Vielzahl von Algorithmen wurde entwickelt um die Berechnung des Schnittpunktes eines Strahls mit der Geometrie zu beschleunigen. Solche sogenannten Ray Tracing Beschleuniger basieren meistens auf einer hierarchischen Aufteilung des Raumes in Teilräume, die die Größe der Suchmenge einschränken und somit eine schnelle hierarchische Suche nach dem *Divide And Conquer* Prinzip erlauben. Eine Übersicht über die verschiedenen Algorithmen ist in ARVO und KIRK (1989) zu finden. Im Folgenden werden die wichtigsten Verfahren zusammengefasst.

#### 3.4.1 Listenorientierte Beschleuniger

Der einfachste Algorithmus zur Schnittpunktberechnung basiert auf der Organisation der geometrischen Objekte in einer einfachen verketteten Liste. Bei der Schnittpunktberechnung muss der Schnittpunkt des Strahls mit jedem Objekt in der Liste bestimmt und der Entfernung nach sortiert werden. Dies führt für Szenarien mit  $M$  homogenen Objekten

zu einer Berechnungskomplexität von  $\mathcal{O}(M)$ , wenn nur der dem Ursprung am nächsten gelegene Schnittpunkt bestimmt werden soll. Dies zusammen mit der Gesamtzahl der Schnittpunkttests für  $N$  aufeinanderfolgende Interaktionen führt zu der exponentiellen Berechnungskomplexität von  $\mathcal{O}(M2^N)$ . Die vollständige Suche ist somit sehr ineffizient für Szenarien mit einer Vielzahl von Objekten.

### 3.4.2 Bounding Objects

Das listenorientierte Verfahren dient als Basis für eine effiziente hierarchische Methode. Hierbei werden um alle komplexen Objekte sog. *Bounding Objekte* gelegt. Diese Bounding Objekte haben meistens eine sehr einfache Form und ermöglichen somit einen schnellen Schnittpunkttest. Andererseits müssen die Bounding Objekte möglichst eng das eigentliche Objekt umschließen. Als Bounding Objekte werden in der Regel Kugeln, Zylinder und, sehr häufig, Quader (Bounding Boxes) verwendet. Da der Schnittpunkttest zunächst mit dem Bounding Objekt durchgeführt wird, werden alle Strahlen, die dieses nicht treffen, von dem eigentlichen Test mit dem komplexen Objekt ausgeschlossen. Viele komplexe Geometrien sind hierarchisch aufgebaut. Dadurch können auch Bounding Objekte um jede Stufe der Hierarchie gelegt werden. Auf diese Weise kann die Komplexität des Schnittpunkttests im optimalen Fall von  $\mathcal{O}(M)$  auf  $\mathcal{O}(\log(M))$  gesenkt werden, wobei der zusätzliche Aufwand durch das Testen der Bounding Objekte vernachlässigt wurde.

### 3.4.3 Gitterorientierte Beschleuniger

Gitterorientierte Verfahren teilen den dreidimensionalen Raum in eine Hierarchie von Zellen (sog. Voxel) auf, in die die Objekte des Szenario einsortiert werden. Da die Schnittpunktberechnung zunächst mit dem Gitter durchgeführt wird, kann die Menge der Objekte, die für einen Schnittpunkttest in Betracht kommen, weitgehend eingeschränkt werden. Die einzelnen Objekte können in sich Bounding Objekte enthalten und aus vielen Einzelteilen hierarchisch aufgebaut sein. Der hauptsächliche Nachteil dieser Methode ist der relativ hohe Speicheraufwand, der für die Speicherung der Gitterdaten benötigt wird. Eine Implementierung mit Hilfe von *Hash Tabellen* kann dieses Problem weitgehend beheben. Gitterorientierte Verfahren sind sehr effizient, wenn sie auf Szenarien mit gleichmäßig verteilten Objekten angewendet werden. Eine Erweiterung des einfachen Gitterverfahrens, indem abhängig von der Objektdichte bestimmte Zellen des Gitters in feinmaschige Untergitter aufgeteilt werden, führt zu Multigrid (Voxelbaum) Methoden (RÜTTEN, 1994).

### 3.4.4 Hierarchische Beschleuniger

Beschleunigungsverfahren auf der Basis von Baumstrukturen gehören zu den am Häufigsten eingesetzten Schnittpunktmethoden überhaupt. Dies gilt umso mehr, wenn hierarchische Erweiterungen des Bounding Objekt und Gitterbeschleunigers als Spezialfälle von Bäumen angesehen werden. Durch die Organisation der geometrischen Daten in der Form einer baumförmigen Hierarchie kann der Aufwand einer Schnittpunktsuche drastisch gesenkt werden. Hierzu muss jedoch die Baumstruktur bestimmte Eigenschaften besitzen, auf die im nächsten Kapitel näher eingegangen wird.

Unter den baumorientierten Verfahren sind der *Octree* und der *k-d Baum* die Standardverfahren. Die Konstruktion der durch Heuristiken optimierten Octree und k-d Bäumen wurde in (SUBRAMANIAN, 1990) untersucht. Die Optimierung dieser Baumstrukturen mit *Simulated Annealing* und *genetischen Algorithmen* wurde ebenfalls untersucht (FRACH und FISCHER, 2002). In den letzten Jahren haben binäre Partitionierungsbäume an Bedeutung gewonnen (FUCHS et al., 1980). Sie haben sich insbesondere bei der effizienten



Segmentierung und bei der Darstellung von dreidimensionalen Datensätzen (SUBRAMANIAN und NAYLOR, 1997; THIBAUT und NAYLOR, 1987) bewährt. Eine mengentheoretische Betrachtung dieser Bäume ermöglicht die Durchführung beliebiger Mengenoperationen (Kapitel 4) und bildet damit die Basis für die Konstruktion topologischer Bäume, die Informationen über die räumlichen Beziehungen verschiedener geometrischer Merkmale eines Objekts in unterschiedlichen Dimensionen beinhalten (COMBA und NAYLOR, 1996). Da binäre Partitionierungsbäume eine Verallgemeinerung des klassischen Binärbaumes auf  $N$ -dimensionale Räume darstellen, gehören die auf ihnen basierten Suchalgorithmen zu den effizientesten bekannten Verfahren ohne Vorwissen dar.

### 3.4.5 Zellenbasierte Beschleuniger

Eine weitere Klasse von Verfahren stellen die zellenbasierten Beschleuniger dar. Diese Verfahren basieren auf der Aufteilung der Geometrie in geschlossene konvexe Zellen, wobei die Begrenzungsflächen jeder Zelle die Information über die benachbarten Zellen enthalten. Die Schnittpunktberechnung beschränkt sich auf die Bestimmung des Schnittpunktes des Strahls in derjenigen Zelle, die den Sender enthält. Über die Auswertung der Wandinformation kann die Strahlverfolgung in der Nachbarzelle weitergeführt werden. Ein solches Verfahren wurde in (SCHÖBERL, 1997) verwendet, wobei die Aufteilung der Geometrie in Zellen manuell durchgeführt wurde. Ein automatisches Partitionierungsverfahren auf der Basis eines topologischen BSP Baums wurde im Rahmen dieser Arbeit entwickelt und in (FRACH et al., 2000) vorgestellt. Ein ähnliches Verfahren wurde von FUNKHOUSER et al. (1998) verwendet. Ein weiterer zellenbasierter Algorithmus unter der Verwendung tetraederförmiger Zellen wurde von ZHANG et al. (2001) vorgestellt.

## 3.5 Meta-Hierarchien

Das Konzept der Meta-Hierarchie wurde von KIRK und ARVO (1988) eingeführt und von ARVO (1990) erweitert. Der Grund für diese Entwicklung lag in der Beobachtung, dass die Effizienz der obengenannten Beschleunigungsalgorithmen stark von der Geometrie selbst abhängig ist. Es war somit wünschenswert, verschiedene Beschleunigungsverfahren für verschiedene Teile der Geometrie (eventuell auch auf verschiedenen Skalen) miteinander kombinieren zu können.

Die *Meta-Hierarchie* abstrahiert die Repräsentation einer Geometrie und definiert eine eindeutige Schnittstelle zum Ray Tracing Algorithmus. Durch die Abstrahierung des geometrischen Objekts wird das Ray Tracing Verfahren unabhängig von der Geometrie, da die Eigenheiten der geometrischen Objekte hinter der funktionalen Schnittstelle des Meta-Objekts im Sinne der objektorientierten Programmierung versteckt werden.

Andererseits erlaubt diese Trennung der Funktionalität die nahtlose Kombination verschiedener Beschleunigungsverfahren zu komplexen hybriden Beschleunigern, ohne die innere Ray Tracing Schleife modifizieren zu müssen. Da die Hybridisierung der Beschleunigungsverfahren notwendig ist, um ein global effizientes Ray Tracing Verfahren zu erhalten, stellen die Meta-Hierarchien einen wichtigen Baustein bei der Entwicklung eines effizienten strahlenoptischen Verfahrens dar. Zugleich erlaubt das Konzept die kosteneffiziente Wiederverwendung von Kernroutinen für strahlenoptische Simulationen auch in anderen Anwendungsgebieten.

### 3.6 Zusammenfassung

In diesem Kapitel wurde auf die algorithmischen Fragen bei der Implementierung eines strahlenoptischen Verfahrens eingegangen. Ausgehend von der Lösung des Fermatschen Integrals (2.1) mit der Monte Carlo Methode wurden alternative deterministische Methoden vorgestellt mit dem Ziel, die Anzahl der benötigten Strahlen zu reduzieren. An der rekursiven Implementierung eines einfachen Ray Tracing Algorithmus wurde deutlich, dass die Schnittpunktberechnung eines Strahls mit der Geometrie und die damit verbundene Suche nach dem nächsten Objekts, das von dem Strahl getroffen wird, die Geschwindigkeit bestimmende Operation darstellt. Weiterhin wurden die wichtigsten hierarchischen Beschleunigungsverfahren dieser Schnittpunktsuche erläutert und ihre Verknüpfung zu hybriden Ray Tracing Beschleunigungsverfahren unter dem Dach der abstrakten Meta-Hierarchien vorgestellt.



## Konstruktion binärer Partitionierungsbäume

Im letzten Kapitel wurde die Schnittpunktberechnung eines Strahls mit der Geometrie des Szenarios als die zeitbestimmende Kernfunktion eines Ray Tracing Verfahrens identifiziert. In diesem Kapitel wird nun die Konstruktion *räumlicher binärer Partitionierungsbäume* behandelt, die unter anderem die Implementierung einer schnellen und effizienten Schnittpunktberechnung erlauben. Das Konzept des klassischen binären Partitionierungsbaumes wird zu einem hierarchischen Partitionierungsbaum erweitert und unter mengentheoretischen Gesichtspunkten neu interpretiert. Es wird eine Methode zu einer schnellen Durchführung von mengentheoretischen Operationen nach dem Vorbild der CSG Verfahren behandelt und zur Lösung des Prioritätenproblems von sich überlappenden Polygonen in der Ebene eingesetzt. Durch die Addition von topologischen Informationen werden die Zusammenhänge der geometrischen Einheiten auf den verschiedenen Ebenen der Hierarchie bereits bei der Konstruktion des Baumes erfasst und stehen für die weitere Verarbeitung zur Verfügung. Auf diese Weise können mögliche Beugungskanten schon in der Konstruktionsphase identifiziert und bezüglich ihrer elektrischen Eigenschaften analysiert werden. Die topologischen binären Partitionierungsbäume eignen sich für eine universelle Darstellung beliebiger polygonbasierter Geometrien. Eine solche Darstellung erlaubt zum Beispiel eine einfache Rekonstruktion der Oberfläche eines komplexen CSG Modells und führt ebenfalls zu einem Algorithmus zur automatischen Partitionierung einer beliebigen Geometrie in konvexe Zellen, wie sie zum Beispiel von SCHÖBERL (1997) zur Strahlwegsuche eingesetzt werden.

### 4.1 Die Struktur eines binären Partitionierungsbaumes

Binäre Partitionierungsbäume (BSP) stellen eine baumförmige hierarchische Struktur im  $N$ -dimensionalen Raum dar und können somit als die Verallgemeinerung des klassischen Binärbaumes von dem 0-dimensionalen Raum der reellen Zahlen  $\mathbb{R}^0$  auf beliebig dimensionale Räume betrachtet werden (NAYLOR et al., 1990).

Eine für die Struktur des BSP Baumes charakteristische Eigenschaft ist die Aufteilung des  $N$ -dimensionalen Raumes in zwei Teilräume mit Hilfe eines  $(N - 1)$ -dimensionalen Partitionierers, zum Beispiel in der Form einer orientierten Hyperebene. Diese wird in ihrer parameterischen Form durch das Skalarprodukt ihres  $N$ -dimensionalen Normalenvektors  $\vec{a} = (a_1, \dots, a_N)$  mit dem ebenfalls  $N$ -dimensionalen Ortsvektor  $\vec{x} = (x_1, \dots, x_N)$  der Punkte in der Ebene beschrieben:

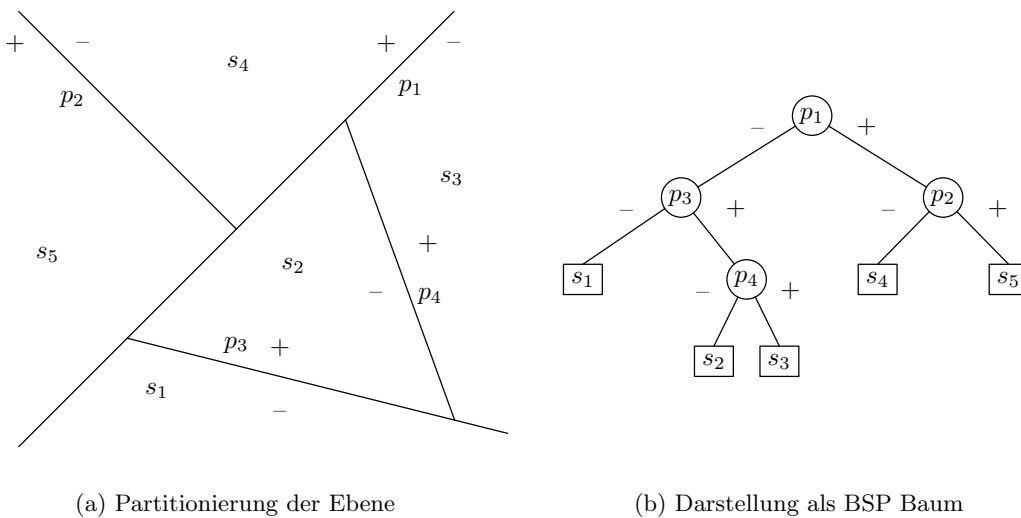
$$a_N x_N + \dots + a_2 x_2 + a_1 x_1 = \sum_{i=1}^N a_i x_i = d \quad . \quad (4.1)$$

Der Parameter  $d$  entspricht dem Skalarprodukt eines beliebigen Punktes  $\vec{p}$  in der Ebene mit dem Normalenvektor  $\vec{a}$ . In der Vektorschreibweise lässt sich die obige Gleichung durch  $(\vec{x} - \vec{p}) \cdot \vec{a} = 0$  abkürzen. Diese definiert die Hyperebene als die Menge aller Punkte  $\vec{x}$ , für die der Differenzvektor  $(\vec{x} - \vec{p})$  zu einem beliebigen Punkt  $\vec{p}$  in der Ebene senkrecht auf den Normalenvektor  $\vec{a}$  der Ebene steht. Im Falle des zweidimensionalen Raumes  $\mathbb{R}^2$  ist der Partitionierer eine durch die Gleichung  $a_2 x_2 + a_1 x_1 = d$  implizit definierte Linie,

die einen eindimensionalen Teilraum in  $\mathbb{R}^2$  darstellt. Die Richtung der Linie definiert die Orientierung des Partitionierers. Der Einsatz nichtlinearer gekrümmter Partitionierer ist zwar auch denkbar, jedoch aufgrund der damit verbundenen analytischen Schwierigkeiten sowie des höheren Rechenaufwands praktisch (noch) nicht relevant.

Der Partitionierer stellt die Wurzel des künftigen BSP Baumes dar. Durch die eindeutige Orientierung der Hyperebene liegt ein Teilraum auf der positiven und der andere Teilraum auf der negativen Seite der Ebene. Diese Aufteilung spiegelt sich im Aufbau des Binärbaumes wieder, wo beide Teilräume, und damit auch alle in ihnen enthaltene Objekte, dem entsprechenden Blatt zugeordnet werden. Der Teilraum des Partitionierers, d.h. die Menge aller Punkte, die die Gleichung (4.1) erfüllen, wird der Wurzel des Baumes zugeordnet.

Im folgenden Schritt werden durch weitere geeignet gewählte Hyperebenen die beiden Teilräume in analoger Weise aufgeteilt. Die so neu entstandenen Teilbäume ersetzen die entsprechenden Blätter im ursprünglichen BSP Baum. Die Prozedur wird rekursiv fortgeführt, bis ein definiertes Abbruchkriterium erfüllt ist. Das Ergebnis der Partitionierung ist ein Binärbaum mit  $(N - 1)$ -dimensionalen Hyperebenen in den inneren Knoten des Baumes und  $N$ -dimensionalen konvexen Teilräumen in den Blättern. Die Repräsentation einer Ebene durch einen BSP Baum ist im Bild 4.1 veranschaulicht.



**Bild 4.1:** Die Partitionierung der Ebene und der dazugehörige binäre Partitionierungsbaum. Die gerichtete Linie  $p_1$  teilt die Ebene in zwei Teilebenen auf und stellt die Wurzel des BSP Baumes dar. Jede der beiden Teilebenen wird durch weitere Liniensegmente nochmals unterteilt. Diese Linien stellen die inneren Knoten des Baumes dar. Die Teilbereiche  $s_1$  bis  $s_5$  befinden sich in den Blättern des Baumes. Das Verfahren ist auf Räume mit beliebiger Anzahl von Dimensionen anwendbar.

Bei der Partitionierung können im Raum befindliche Objekte von einer Hyperebene geschnitten werden. In diesem Fall muss ein solches Objekt in Teilobjekte auf beiden Seiten der Hyperebene aufgeteilt werden können, die den entsprechenden Teilbäumen zugeteilt werden. Der damit verbundene Rechenaufwand kann bei analytischen Objekten, die zum Beispiel durch *Non-Uniform Rational B-Splines* (NURBS) beschrieben werden, sehr groß werden. Alternativ, um eine Aufteilung zu vermeiden, kann das Objekt auf beiden Seiten der Hyperebene *referenziert* werden, d.h. es wird in beiden Teilräumen indirekt durch

Zeiger auf dieses Objekt verwiesen. Diese Methode setzt jedoch die Implementierung einer komplexen Speicherverwaltung voraus. Generell sollte die Lage der Hyperebenen so gewählt werden, dass die Anzahl der durch die Ebenen geschnittenen Objekte minimal wird. Wird die Lage jeder Hyperebene derart gewählt, dass sich die Anzahl der Objekte auf beiden Seiten der Ebene nur um maximal eins unterscheidet, so entsteht ein balancierter BSP Baum mit der minimalen Baumhöhe. Die Schnittpunktsuche mit einem balancierten Baum hat im besten Fall die Komplexität  $\mathcal{O}(\log N)$  wenn  $N$  die Anzahl der im Baum enthaltenen Objekte ist.

## 4.2 Konstruktion eines modifizierten BSP Baumes

Wie im letzten Abschnitt bereits erwähnt, stellt die Auswahl geeigneter Partitionierungsebenen bei der Konstruktion eines BSP Baumes für eine beliebige Objektanordnung ein für die Geschwindigkeit der späteren Schnittpunktberechnung wichtiges aber nicht einfach zu lösendes Problem dar. Bei einer polygonalen Darstellung der Geometrie liegt es nahe, die Partitionierungsebenen durch die bereits im Szenario enthaltenen Polygone zu definieren. Deswegen eignen sich Mobilfunkszenarien, deren Geometrie durch Polygone beschrieben wird, besonders für den Einsatz von BSP Bäumen. Da sich in diesem Fall Polygone in den Partitionierungsebenen befinden, muss das Konzept des BSP Baumes erweitert und für diesen Einsatz spezialisiert werden.

Allgemein werden Mobilfunkszenarien durch die Lage der polygonalen Wände definiert. Eine erste Modifikation des BSP Baumes für diesen Anwendungsfall besteht darin, die Baumknoten um Listen für die in den Partitionierungsebenen befindlichen Polygone zu erweitern. Da diese Polygone in den Ebenen liegen, können sie bei der Konstruktion des Baumes in das lokale Koordinatensystem der jeweiligen Ebene transformiert werden, um Speicherplatz und Rechenzeit zu sparen. Da eine Beschreibung der Geometrie mit einem BSP Baum unabhängig von der Positionen des Senders und der Empfänger ist, muss der BSP Baum unter der Annahme einer statischen Geometrie in einem Vorverarbeitungsschritt nur einmal konstruiert werden und steht dann für die späteren Simulationen zur Verfügung. Vor der jeweiligen Simulation muss dann der Baum nur um die fehlenden Positionen der Sender und der Empfänger ergänzt werden. Diese Eigenschaft stellte sich besonders bei der zeitlich aufwendigen Konstruktion optimierter BSP Bäume als sehr vorteilhaft heraus.

Eine weitere Modifikation betrifft die Blätter des BSP Baumes. Diese müssen um geeignete Datenstrukturen erweitert werden, welche die in den zugehörigen Teilräumen platzierten punktförmigen Empfänger enthalten. Im einfachsten Fall werden die im Teilraum befindlichen Empfänger in einer linearen Liste abgelegt. Da die Größe eines Teilraums im Allgemeinen nicht beschränkt ist, können die Teilräume unter Umständen sehr viele Empfänger enthalten. Dadurch wird die Schnittpunktberechnung des Strahls mit der Liste der Empfänger zu einer  $\mathcal{O}(N)$  Operation, die sich negativ auf die Geschwindigkeit der Simulation auswirken kann. Im Abschnitt 4.5 wird daher eine Methode vorgestellt, die eine Beschleunigung der Empfängerdetektion durch eine einfache Erweiterung des BSP Baumes ermöglicht. Da die Rückwirkung der Umgebung auf den Sender in der Regel nicht berücksichtigt wird und deswegen keine Schnittpunktberechnung eines Strahls mit dem Sender erwünscht ist, müssen die Senderpositionen nicht zwingend im BSP Baum abgelegt werden.

Die Konstruktion des modifizierten BSP Baumes verläuft analog zu der eines klassischen BSP Baumes. Die im Szenario enthaltenen Wandpolygone werden in einer vorgegebenen Reihenfolge in den Baum eingefügt. Dabei wird die Lage jedes Polygons zu der aktuellen Partitionierungsebene geprüft und das Polygon durch die Ebene in zwei neue

Polygone aufgespalten, falls es von der Ebene geschnitten wird. Das Ergebnis dieses Vergleichs wird in den entsprechenden Teilbaum auf der positiven und/oder negativen Seite der Ebene weitergereicht, wo dieselbe Operation solange rekursiv durchgeführt wird, bis sich alle Fragmente des ursprünglichen Polygons in den zugehörigen Blättern des Baumes befinden. Diese Blätter werden anschließend durch diese Polygone in jeweils zwei neue Teilräume aufgeteilt und die Polygone in das lokale Koordinatensystem der Ebenen in den neuen Baumknoten transformiert und abgelegt. Bei dem Vergleich des Wandpolygons mit einer Partitionierungsebene kann es vorkommen, dass das Polygon in der Partitionierungsebene liegt. In diesem Fall wird das Polygon ebenfalls in das Koordinatensystem der Ebene transformiert und in die Liste der bereits in der Ebene befindlichen Polygone aufgenommen. Der Algorithmus ist im Diagramm 4.2 zu sehen.

### Konstruktion des modifizierten BSP Baumes — Einfügen eines Polygons

<b>insert(BSP <math>B</math>, Polygon <math>P</math>)</b> Diese Funktion fügt das Polygon $P$ in den Baum $B$ ein.	
Ist $P$ ist ein gültiges Polygon ?	
J	N
Ist $B$ ein Knoten des BSP Baumes ?	
J	N
<b>split(Polygon <math>P</math>, Ebene <math>B.p</math>, Polygon <math>P_+</math>, Polygon <math>P_-</math>)</b> Diese Funktion bestimmt die Lage des Polygons $P$ relativ zu der aktuellen Partitionierungsebene $B.p$ . Falls das Polygon ausschließlich auf einer Seite der Ebene liegt, wird es als das entsprechende Polygon $P_+$ oder $P_-$ zurückgegeben. Wird das Polygon von der Ebene geschnitten, so werden die beiden Fragmente in $P_+$ und $P_-$ zurückgegeben. Liegt das Polygon in der Ebene, so sind die Polygone $P_+$ und $P_-$ leer.	In diesem Fall kann $B$ nur ein Blatt des Baumes sein. Da sich zu diesem Zeitpunkt noch keine Empfänger im Baum befinden, ist das Blatt leer. Ersetze das Blatt mit einem neuen Knoten. Die Ebene des Knotens, die zur Partitionierung des Teilraums verwendet wird, ist durch das Polygon $P$ bestimmt. Der Normalenvektor, das lokale Koordinatensystem der Ebene, und die Transformation des Polygons in die lokalen Koordinaten der neuen Ebene werden berechnet und das 2D Polygon wird in die leere Polygonliste des neuen Knotens $B$ eingefügt. Außerdem wird die initiale planare Bounding Box des Knotens berechnet.
Liegt das Polygon $P$ in der Ebene $B.P$ ?	
J	N
Das Polygon wird in das lokale Koordinatensystem der Ebene transformiert und in die Polygonliste des Knotens $B$ eingefügt. Polygone werden in der Polygonliste des Knotens in der Reihenfolge ihrer Priorität sortiert. Zusätzlich wird die planare Bounding Box der Polygonliste aktualisiert.	<b>insert(BSP <math>B_+</math>, Polygon <math>P_+</math>)</b> Fügt das Polygonfragment $P_+$ in den positiven Teilbaum ein.
	<b>insert(BSP <math>B_-</math>, Polygon <math>P_-</math>)</b> Fügt das Polygonfragment $P_-$ in den negativen Teilbaum ein.
return	

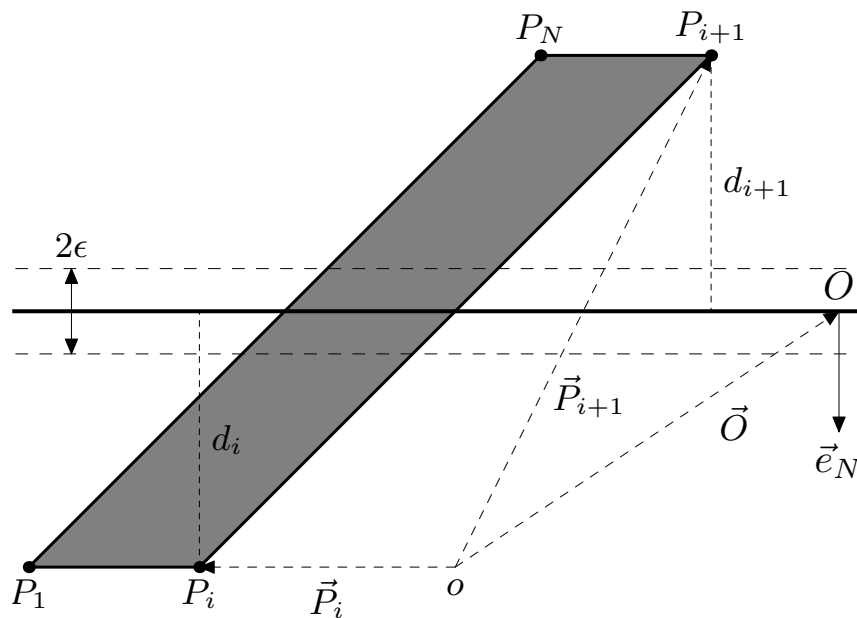
**Bild 4.2:** Eine Beispielimplementierung der Konstruktion des modifizierten BSP Baumes. Die Funktion `insert()` wird rekursiv mit dem Polygonfragment aufgerufen und fügt dieses in den entsprechenden Teilbaum ein. Die Funktion `split()` bestimmt dabei die relative Lage des Polygons zur Partitionierungsebene des jeweils aktuellen Knotens. Auf die Implementierung dieser Funktion wird unter Anderem in der [BSP-FAQ \(2002\)](#) ausführlicher eingegangen.

Die Lage eines Polygons relativ zu der Partitionierungsebene kann durch einen Vergleich der vorzeichenbehafteten Abstände  $d_i$  der Polygoneckpunkte  $\vec{P}_i$  zur Ebene leicht bestimmt werden (Bild 4.3). Der Abstand wird mit Hilfe des Ursprungsvektors  $\vec{O}$  des ebenenbezogenen Koordinatensystems und des Einheitsvektors der Flächennormalen  $\vec{e}_N$ ,

der die Orientierung der Partitionierungsebene im Raum bestimmt, berechnet:

$$d_i = \vec{e}_N \cdot (\vec{P}_i - \vec{O}) \quad . \quad (4.2)$$

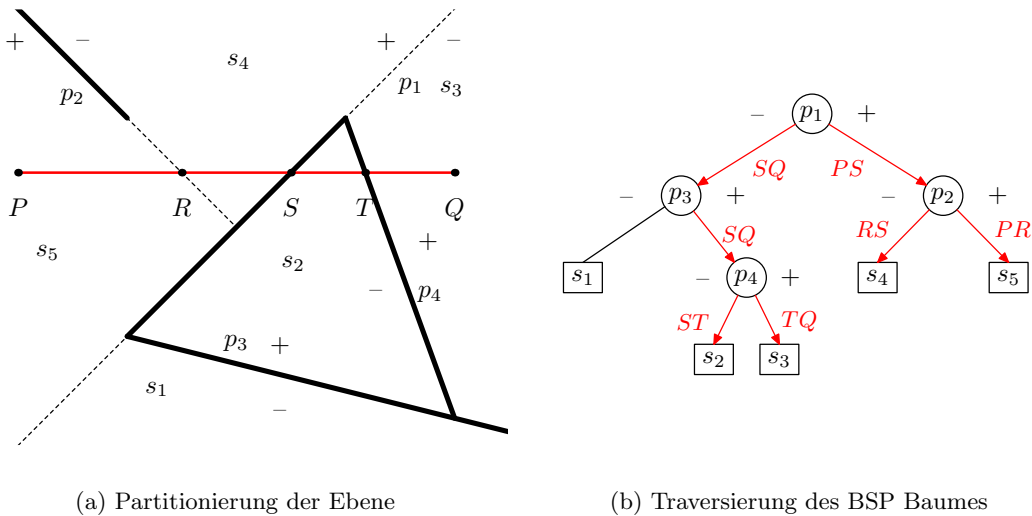
Das Vorzeichen von  $d_i$  gibt die Seite der Ebene an, auf der sich der Punkt befindet. Besitzen alle Abstände das gleiche Vorzeichen, so liegt das Polygon ausschließlich auf der positiven oder der negativen Seite der Ebene. Ist der Abstand einiger Punkte gleich Null, so berührt das Polygon in diesen Punkten die Partitionierungsebene. Falls diese Punkte eine gemeinsame Kante bilden, liegen auch alle Punkte der Kante in der Partitionierungsebene. Zuletzt können zwei aufeinanderfolgende Punkte auf verschiedenen Seiten der Ebene liegen. Damit wird die von diesen Punkten gebildete Polygonkante durch die Ebene geteilt und beide Teile in die Datenstruktur der zugehörigen Polygonfragmente auf der positiven und auf der negativen Seite der Ebene abgespeichert. Um das Verfahren numerisch stabil zu machen, muss die Ebene mit einer numerischen Dicke  $2\epsilon$  versehen werden. Diese Dicke definiert die Toleranz des Algorithmus für Rundungsfehler, die durch die Verwendung von Fließkommazahlen mit einer endlichen Wortbreite entstehen.



**Bild 4.3:** Die Partitionierung eines Polygons mit einer Ebene basiert auf dem Vergleich der Vorzeichen der Entfernung  $d_i$  zweier aufeinander folgender Polygoneckpunkte  $\vec{P}_i$  von der Partitionierungsebene.

### 4.3 Schnelle Schnittpunktberechnung

Mit der BSP basierten Schnittpunktsuche können alle Objekte, die von einem Strahl geschnitten werden, mit nur einer Abfrage des Baumes bestimmt werden. Zusätzlich werden auch alle im Strahlkegel liegenden Empfänger bestimmt. Die Partitionierung in der Ebene und der zugehörige BSP Baum, der für die folgende Erläuterung der Vorgehensweise bei der Schnittpunktsuche benutzt wird, sind im Bild 4.4 zu sehen. Die gleiche Vorgehensweise ist auf Räume mit einer beliebigen Dimension übertragbar.



**Bild 4.4:** Die Schnittpunktsuche mit Hilfe eines modifizierten BSP Baumes. Der Strahl  $PQ$  wird an der Ebene  $p_1$  in die Teilstrahlen  $PS$  und  $SQ$  geteilt. Beginnend mit  $PS$  werden beide Teilstrahlen in beide Teilbäume weitergereicht. Die Blätter des Baumes werden in der Reihenfolge ihrer Entfernung zum Strahlursprung  $P$  besucht. Die Schnittpunkte mit den Ebenen werden mit den Polygonen in der Ebene verglichen und die Punkte  $S$  und  $T$  als Treffer gespeichert. Auf die Darstellung der Polygonlisten wurde verzichtet. Der Weg der Teilstrahlen durch den BSP Baum ist farblich gekennzeichnet.

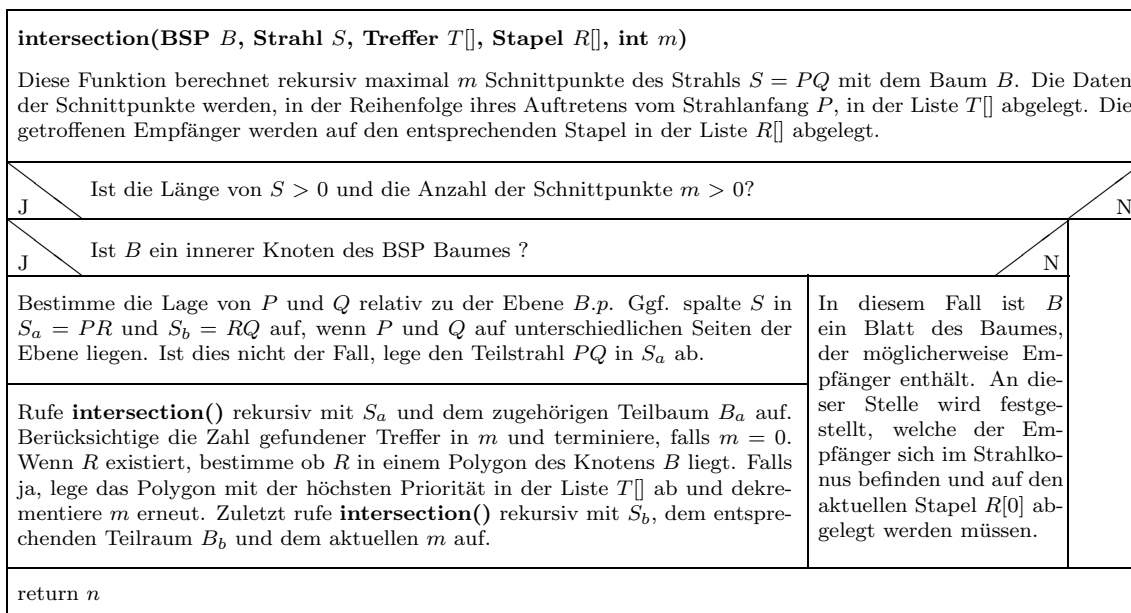
Die schnelle Schnittpunktsuche mit einem BSP Baum beginnt mit der Definition der Strahlparameter, also des Strahlursprungs, der Strahlrichtung und der Länge des Strahls. Diese Parameter werden entweder durch den Strahlengenerator des Senders oder durch die Verfolgung der reflektierten Teilstrahlen in der inneren Ray Tracing Schleife festgelegt. Die Gesamtlänge des Strahlweges wird zu Anfang durch den maximal geforderten Dynamikbereich begrenzt. Mit dem Fortschreiten der Strahlverfolgung wird die Länge der Teilstrahlen sukzessive verkürzt.

Das Prinzip der BSP-basierten Schnittpunktsuche beruht auf der Klassifizierung des durch ein Liniensegment beschriebenen Strahls bezüglich der jeweiligen Partitionierungsebene. Die Klassifizierung erfolgt ähnlich dem Einfügen der Polygone in den Baum, indem die Abstände der beiden Endpunkte des Strahls zur Partitionierungsebene miteinander verglichen werden. Sind die Vorzeichen identisch oder liegt einer der Punkte in der Ebene, so liegt der Strahl gänzlich auf einer der beiden Seiten der Ebene. Sind beide Abstände gleich Null, so liegt der Strahl in der Ebene. Dieser Fall bedarf einer speziellen Behandlung, tritt aber sehr selten auf. Die letzte Möglichkeit besteht darin, dass beide Abstände unterschiedliche Vorzeichen besitzen und der Strahl damit durch die Ebene in zwei Teilstrahlen aufgespalten wird. In jedem Fall wird das Ergebnis der Klassifizierung in den/die entsprechenden Teilbäume des aktuellen Knotens weitergereicht, wo mit den Teilstrahlen dieselbe Prozedur rekursiv durchgeführt wird, bis alle Teilstrahlen die Blätter des Baumes erreichen. An dieser Stelle werden die sich im Blatt befindlichen Empfänger getestet und Zeiger auf die Empfänger, die von dem Teilstrahl getroffen wurden, für die weitere Verarbeitung gespeichert. Die Rekursion endet an dieser Stelle für diesen Teilstrahl und wird nach der Rückkehr der Funktion zu der nächsthöheren Rekursionsebene mit den verblei-

benden Teilstrahlen fortgesetzt. Der Weg der Teilstrahlen durch den BSP Baum ist im Bild 4.4 farblich gekennzeichnet.

Bei der Traversierung des Baumes ist es wichtig, dass nach einer Aufteilung des Strahls in zwei Teilstrahlen durch die Partitionierungsebene der sendeseitige Teilstrahl zuerst in den entsprechenden Teilbaum weitergereicht wird. Der andere Teilstrahl wird, falls vorhanden, zusammen mit den Daten des Schnittpunktes für eine spätere Bearbeitung abgelegt. Nach einer Rückkehr des rekursiven Aufrufs der Funktion mit dem sendeseitigen Teilbaum wird die Funktion auf der aktuellen Stufe der Rekursion wieder aktiv. Falls ein Schnittpunkt des Strahls mit der Partitionierungsebene existiert, prüft diese Funktion zunächst, ob der Strahl einen Polygon in dieser Ebene getroffen hat. Falls dies der Fall ist, wurde ein neuer Schnittpunkt gefunden und die Objektdaten werden in der Trefferliste abgelegt. Anschließend wird die Rekursion mit dem Teilstrahl auf der anderen Seite der Ebene fortgesetzt, falls dieser existiert. Falls kein Schnittpunkt mit der Ebene existierte oder sich der Endpunkt des Teilstrahls in der Ebene befindet, terminiert diese Funktion ebenfalls und die Prozedur kehrt zu der nächsthöheren Rekursionsebene zurück. Der erste Funktionsaufruf gibt als Ergebnis die Liste aller getroffenen Objekte sowie die Liste aller Empfänger zwischen zwei Objekttreffern zurück. Eine rekursive Implementierung des Algorithmus ist im folgenden Diagramm 4.5 dargestellt. Sie eignet sich aufgrund der hierarchischen Form des BSP Baumes am besten zur anschaulichen Erklärung des Algorithmus. Eine iterative Realisierung des Algorithmus ist ebenfalls möglich und sogar effizienter, da Funktionsaufrufe eingespart und wenige Daten auf den internen Stapelspeicher abgelegt werden müssen. Da jedoch die iterative Lösung weit weniger übersichtlich ist, wurde hier auf eine ausführliche Beschreibung verzichtet.

### Modifizierter BSP Baum — Schnittpunktssuche



**Bild 4.5:** Die rekursive Implementierung der schnellen Schnittpunktssuche mit modifizierten BSP Bäumen.

Der Sonderfall eines Strahls in der Partitionierungsebene wird im Diagramm 4.5 nicht berücksichtigt. In diesem Fall wird die Rekursion beendet, da keine weitere Zuordnung zu



den Teilbäumen möglich ist. Eine Betrachtung der Schnittpunkte mit den Polygonkanten in der Ebene, würde nicht zum Gesamtergebnis beitragen, da solche Polygone nicht von dem Strahl gesehen werden.

Die Daten jedes Schnittpunktes werden in einer Liste mit der Datenstruktur **Tref-fer** abgelegt. Diese Datenstruktur stellt zusammen mit der Funktion **intersection()** die Schnittstelle der Geometrie zu der inneren Ray Tracing Schleife dar und muss daher alle notwendigen Informationen enthalten, die zur weiteren Strahlverfolgung benötigt werden. Dazu gehört die Information über die Oberflächenbeschaffenheit des getroffenen Polygons sowie der Normalenvektor der Polygonebene. Weitere Parameter sind nötig, um eine eindeutige Erkennung der Ausbreitungspfade zu ermöglichen. Auf diese Parameter wird ebenfalls im Kapitel 6 näher eingegangen.

Aus Gründen der Rechengeschwindigkeit werden so weit wie möglich keine Daten im Kernspeicher bewegt sondern Zeiger auf die Daten übergeben. Da im Gegensatz zu der Anzahl der Objektstreffer, die durch die vom Benutzer eingestellte maximale Anzahl der Transmissionen limitiert ist, die Zahl der getroffenen Empfänger beliebig sein kann, müssen zur Speicherung der Zeiger dynamische Datenstrukturen verwendet werden, deren Größe sich zur Laufzeit des Programms ändern kann. Solche dynamischen Datenstrukturen können im einfachsten Fall in der Form eines Stapels (engl. *stack*) realisiert werden. Stapel sind dynamische Datenstrukturen, in den die Daten nach dem *Last-In-First-Out* (LIFO) Prinzip verwaltet werden. Wenn Daten auf den Stapel abgelegt werden, wächst der Stapel und der dafür erforderliche Speicher muss beim Betriebssystem angefordert werden. Diese Operation erfordert einen zeitaufwendigen Systemaufruf und sollte daher nur selten aufgerufen werden. Auf einem Parallelrechner mit einem gemeinsamen Speicher stellt dieser Systemaufruf zusätzlich auch einen Synchronisationspunkt aller parallelen Funktionen des Programms und verschlechtert dadurch erheblich das Skalierungsverhalten der Programms auf Rechnern mit vielen Prozessoren. Die Anzahl dieser Systemaufrufe kann erheblich reduziert werden, wenn der Speicher blockweise für mehrere Einträge im Voraus angefordert wird. Um weitere zeitaufwendige Systemaufrufe zu sparen, wird der für den Stapel angeforderte Speicher nach dem Entnehmen der Empfänger nicht wieder freigegeben und steht somit für die Bearbeitung des nächsten Strahls zur Verfügung. Dadurch wird auf Kosten eines höheren Speicherverbrauchs die Anzahl der Systemaufrufe minimiert und die Rechengeschwindigkeit maximiert. Jeder Stapel muss daher mit zusätzlichen Variablen versehen werden, die die aktuelle und die maximale Anzahl der Einträge in der Stapelliste protokollieren.

Die Reihenfolge der Polygonschnittpunkte und der Empfängerstapel ist eng miteinander verbunden. Der erste Stapel  $R[0]$  in der Stapelliste enthält alle Empfänger, die sich vor dem ersten getroffenen Objekt  $T[0]$  befinden. Analog enthält der zweite Stapel  $R[1]$  alle Empfänger, die zwischen dem ersten und dem zweiten getroffenen Polygon  $T[1]$  liegen. Damit auch Empfänger nach  $N$  Transmissionen getroffen werden können, muss die Liste der Empfängerstapel um einen Eintrag länger als die Liste der Polygonschnittpunkte sein. Die Anzahl der getroffenen Polygone wird als Rückgabewert von der Funktion **intersection()** an die innere Ray Tracing Schleife zurückgegeben.

Das Verfahren ist ohne Einschränkungen auch iterativ zu implementieren. Die Funktionsweise ist weitgehend identisch mit der rekursiven Implementierung mit dem Unterschied, dass die aufwendige Verwaltung der Funktionsaufrufe und des internen Stapels durch einfache Verwaltung listenbasierter Stapel, die einmal zu Beginn der Simulation angefordert werden, ersetzt wurde. Die Rekursion kann durch die Verschachtelung von zwei Schleifen realisiert werden, wobei die innere Schleife die Traversierung des Baumes von der Baumwurzel zum Sender hin realisiert und dabei gleichzeitig alle Schnittpunkte der Teilstrahlen mit den Partitionierungsebenen sowie die senderferne Teilstrahlen in die



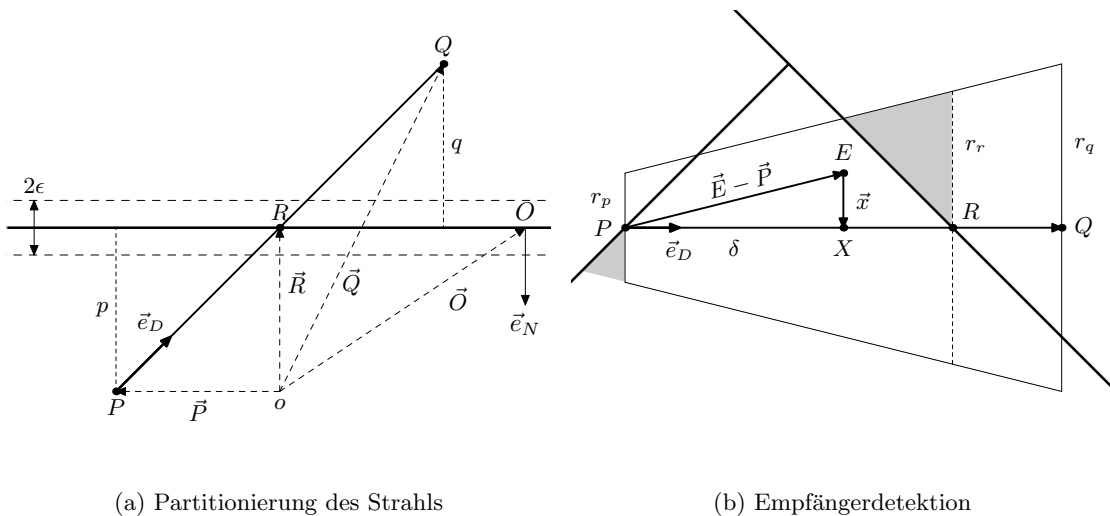
Listen ablegt. Die innere Schleife wird beendet, sobald ein Blatt des Baumes erreicht ist. Die äußere Schleife hat dagegen die Aufgabe, die gewünschte Anzahl der Objektstreffer zu bestimmen und auch die getroffenen Empfänger zu detektieren sowie die innere Schleife mit dem nächsten Teilstrahl aufzurufen. Durch eine iterative Implementierung können Funktionsaufrufe gänzlich vermieden werden.

#### 4.4 Strahlaufteilung und Empfängerdetektion

Der Schnittpunkt des Strahls mit der Partitionierungsebene lässt sich einfach aus den Strahlparametern und der zur Klassifizierung bereits berechneten Distanz des Aufpunktes  $\vec{P}$  des Strahls zur Ebene bestimmen. Die Entfernung des Punktes zur Ebene wird nach der Gleichung (4.2) berechnet. Der Schnittpunkt  $\vec{R}$  des Strahls mit der Ebene ergibt sich nach der im Bild 4.6a dargestellten Geometrie aus der folgenden Beziehung:

$$\vec{R} = \vec{P} - \frac{p}{\vec{e}_N \cdot \vec{e}_D} \vec{e}_D . \quad (4.3)$$

Der Ausdruck  $\vec{e}_N \cdot \vec{e}_D$  entspricht dabei dem Kosinus des Einfallswinkels und  $p$  ist die kürzeste Entfernung der Punktes  $\vec{P}$  zur Ebene nach Gleichung (4.2).



**Bild 4.6:** Zur Strahlaufteilung an einer Ebene und der Empfängerdetektion im BSP Baum. In den grauschattierten Bereichen liegt der Punkt  $X$  vor dem Beginn des Partialstrahls und die Projektion  $\delta$  des Empfängers wird negativ. Bei der Berechnung des Kegelradius am Ort des Empfängers muss die gesamte Länge des Strahls von der Quelle bis zum Punkt  $X$  berücksichtigt werden.

Die Geometrie zur Empfängerdetektion ist im Bild 4.6b zu sehen. Ziel der Berechnung ist es, die kürzeste Distanz des Empfängers zur Symmetrieachse des Strahlkonus zu bestimmen. Mit der Empfängerposition  $\vec{E}$  und den Strahlparametern  $\vec{P}$  und  $\vec{e}_D$  ergibt sich für die Projektion des Empfängers auf die Strahlachse zu

$$\delta = (\vec{E} - \vec{P}) \cdot \vec{e}_D . \quad (4.4)$$

Damit ergibt sich der Vektor der kleinsten Distanz des Empfängers zum Strahl zu

$$\vec{x} = \vec{E} - \vec{P} - \delta \vec{e}_D . \quad (4.5)$$

Die Entfernung zur Strahlachse entspricht der Länge dieses Vektors. Ist diese Entfernung innerhalb des Konus, also

$$\|\vec{x}\| \leq \alpha(l_p + \delta) = r_p + \alpha\delta \quad . \quad (4.6)$$

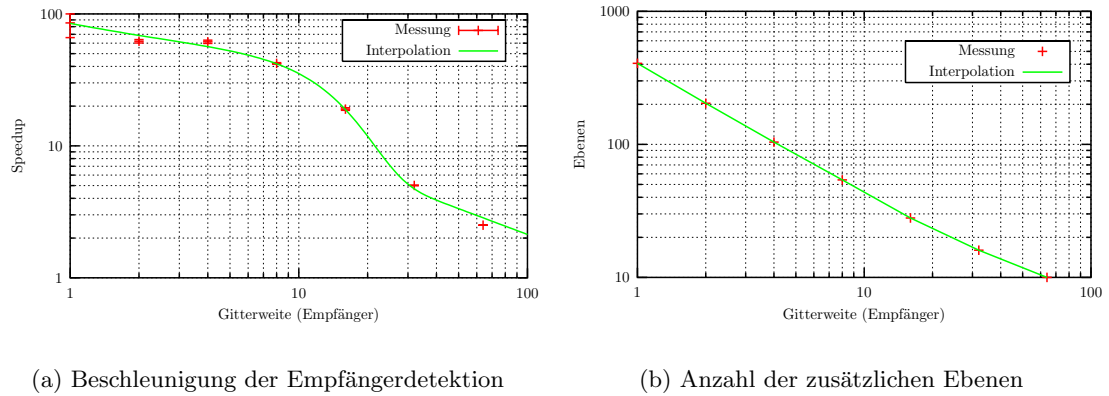
mit  $\alpha$  dem Öffnungswinkel und  $l_p$  der Länge des Strahls von der Quelle bis zum Punkt  $P$ , liegt der Empfänger im Strahlkonus und wird von dem Strahl getroffen. Alternativ können die Radien des Konus  $r_p$  berechnet und um den Beitrag  $\alpha\delta$  ergänzt werden. Hierbei ist zu beachten, dass  $\delta$  in den schattierten Bereichen im Bild 4.6b negativ wird, obwohl die gesamte Länge des Strahls von der Quelle bis zum Punkt  $X$  positiv ist. Eine Prüfung der positiven Gesamtlänge wird automatisch mit in der Berechnung des Kegelradius berücksichtigt.

## 4.5 Beschleunigung der Empfängerdetektion

Die Empfänger werden, wie bereits erwähnt, in den Blättern des Baumes gespeichert. Die Speicherung erfolgt in der Form einer linearen Liste, die für jeden Strahl, der den entsprechenden Teilraum durchquert, vollständig durchsucht werden muss. Da die vollständige Suche eine Berechnungskomplexität  $\mathcal{O}(N)$  besitzt, ist sie insbesondere bei großen Teilräumen mit vielen Empfängern für einen Großteil der gesamte Rechenzeit verantwortlich. Es wurden Möglichkeiten untersucht, die Empfängerdetektion unter Ausnutzung der Eigenschaften des BSP Baumes zu verkürzen.

Die jeweilige Methode der Empfängerdetektion hängt weitgehend von dem eingesetzten Ray Tracing Modell ab. Die bei Cone Tracing eingesetzte Methode wurde im vorangehenden Abschnitt behandelt. Für sie ist die kürzeste Entfernung des Empfangsortes zur Symmetrieachse des Konus entscheidend. Andere Ray Tracing Dialekte nutzen andere Klassifikatoren zur Empfängerdetektion aus, die auf der Basis von Detektionskugeln oder Ebenen realisiert werden. Da jedoch bei allen Methoden dieser Test mit jedem Empfänger in der Liste des Teilraumes durchgeführt werden muss, hat die Reduktion der Anzahl der Empfänger im Teilraum die vorrangige Priorität. Diese Verringerung der Anzahl kann durch eine weitere Partitionierung des Teilraumes mit geeignet gewählten virtuellen Partitionierungsebenen erreicht werden. Diese zusätzlichen Ebenen dürfen die Geometrie des Szenario nicht verändern und enthalten daher keine Polygone. Die Lage der Partitionierungsebenen kann vorteilhaft an die jeweilige Empfängerkonstellation angepasst werden. Für eine gleichmäßig flächenhafte Empfängeranordnung in einem regulären zweidimensionalen Gitter können die Ebenen jeweils zwischen zwei Empfängerreihen in beiden Richtungen angebracht werden. Die Anzahl der zusätzlichen Ebenen kann durch die Zahl der Empfängerreihen zwischen zwei Ebenen beeinflusst werden. Wird diese Zahl in beiden Richtungen gleich gewählt so entstehen quadratische Zellen. Es sind aber auch andere ungleichmäßige Verteilungen der Gittergröße denkbar. Durch das Gitter wird die Anzahl der Empfänger pro Blatt in einer definierten Weise eingeschränkt. Das Gitter wirkt dabei wie eine Art der Bounding Box um die jeweilige Empfängergruppe. Da jeder Strahl dann nur ein begrenztes Volumen und damit auch eine kleinere Anzahl der neuen Blätter durchquert, wird die Zahl der durchzuführenden Empfängertests stark eingeschränkt. Durch die neuen Partitionierungsebenen steigt jedoch die Baumhöhe und damit auch der Aufwand für die Schnittpunktberechnung an. Daher ist es zu erwarten, dass die Beschleunigung der Empfängerdetektion als Funktion der Empfängerzahl pro Gitterzelle für eine bestimmte Anzahl der Empfänger ein Maximum besitzt, in dem die Schnittpunktberechnung und die Empfängerdetektion insgesamt zu dem kleinsten Rechenaufwand führen. Die Lage des Maximums ist von der Geometrie des Szenario und dem Empfängermodell abhängig.

Um die Beschleunigung der Empfängerdetektion zu messen, wurden  $200 \times 200$  punktförmige Empfänger in einem quadratischen Gitter angeordnet. Es wurden unterschiedliche Zellengrößen eingestellt, wobei die Anzahl der Empfängerreihen zwischen zwei Ebenen variiert wurde. Es wurden keine weiteren geometrischen Strukturen berücksichtigt. Das Ergebnis der Simulation für verschiedene Gittergrößen ist im Bild 4.7a dargestellt. Durch das Einfügen der virtuellen Partitionierungsebenen konnte die Detektion der flächendeckend angeordneten Empfänger maximal um den Faktor 85 beschleunigt werden. Das Maximum liegt bei der kleinsten Gitterzelle mit nur einem Empfänger, ein Indiz dafür, dass die Empfängerdetektion insgesamt sehr viel aufwendiger als die Schnittpunktberechnung mit dem BSP Baum ist und damit den gesamten Rechenaufwand dominiert. Die erreichte Beschleunigung



(a) Beschleunigung der Empfängerdetektion

(b) Anzahl der zusätzlichen Ebenen

**Bild 4.7:** Messung und Interpolation der Empfängerbeschleunigung durch das Einfügen von virtuellen Partitionierungsebenen. Die Anzahl der zusätzlichen Partitionierungsebenen ist ebenfalls dargestellt.

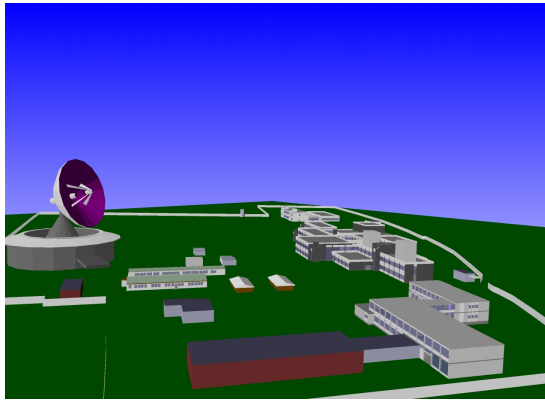
nigung zeigt eine deutliche Sättigung im Bereich der kleinen Zellengrößen. Die Varianz steigt ebenfalls an, ein Hinweis auf die Bedeutung der Gesamtstruktur des Baumes, die durch einen genetischen Algorithmus (Kapitel 5) optimiert wurde. Durch das Einfügen zusätzlicher Ebenen wird die Anzahl der Freiheitsgrade bei der Optimierung erhöht und damit auch die Möglichkeiten der Optimierung erweitert. Eine Interpolation der gemessenen Beschleunigung lässt sich durch die Kombination einer Sättigungsfunktion mit einem polynomialen Anteil näherungsweise beschreiben

$$f(x) = Ae^{-\frac{x^2}{\delta}} + Bx^{-\alpha} . \quad (4.7)$$

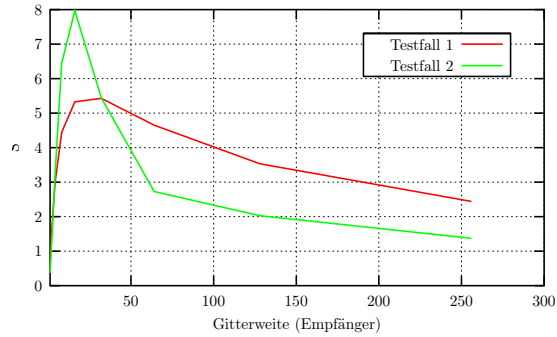
Die Parameter der Funktion sind von der Struktur des Baumes abhängig und müssen durch Messungen bestimmt werden. Für die dargestellte Interpolation ist  $A = B = 42.5$ ,  $\delta = 200$  und  $\alpha = 0.65$  gewählt worden. Die Anzahl der eingefügten virtuellen Ebenen in Abhängigkeit von der Zellengröße ist im Bild 4.7b dargestellt.

Die Verwendung punktförmiger Empfänger zusammen mit der Cone Tracing Methode stellte sich bei der Beschleunigung der Empfängerdetektion als sehr vorteilhaft heraus. Das Einfügen ausgedehnter Empfänger wie z.B. der Detektionskugeln der Strahlabschussmethode führt dazu, dass ein Empfänger durch eine Partitionierungsebene geschnitten wird. In diesem Fall wird der Empfänger auf beiden Seiten der Ebene einsortiert, da sich der Empfänger in beiden Teilräumen befindet. Dadurch steigt die Anzahl der Empfänger in den jeweiligen Teilräumen an. Das Ergebnis der Beschleunigung der Empfängerdetektion in einem realen Szenario (Bild 4.8a) mit ausgedehnten Empfängern, die die Form einer

Detektionskugel besitzen, ist im Bild 4.8b für zwei verschiedene Positionen der Antenne dargestellt. Die Empfänger wurden in einem  $1 \times 1 \text{ m}^2$  Raster in 1.5 m Höhe über das



(a) Testgeometrie



(b) Erreichte Beschleunigung

**Bild 4.8:** Die Testgeometrie und die erreichte Beschleunigung der Empfängerdetektion durch ein Gitter für zwei unterschiedliche Positionen der Antenne. Die Empfänger wurden in einem  $1 \times 1 \text{ m}^2$  Raster in 1.5 m Höhe über das gesamte Gelände verteilt. Die Entfernung der Gitterebenen wurde variiert und ist als die Anzahl der zwischen zwei Gitterebenen befindlichen Empfänger angegeben. Die größte Beschleunigung wird erreicht, wenn sich 32 Empfängerreihen zwischen zwei aufeinanderfolgenden Ebenen befinden.

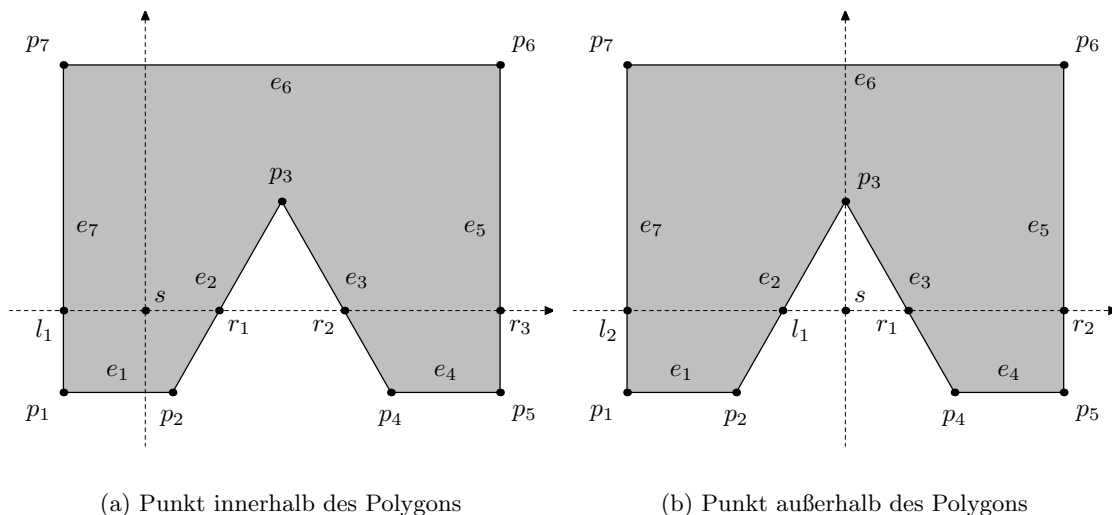
gesamte Gelände verteilt. Die Entfernung der Gitterebenen und damit die Anzahl der Empfänger in einer Gitterzelle wurde variiert und ist im Bild als die Anzahl der zwischen zwei Gitterebenen befindlichen Empfängerreihen angegeben. Eine Empfängerreihe wurde hierbei durch die Mittelpunkte der Detektionskugeln definiert. Die Partitionierung mit virtuellen Ebenen ist ansonsten analog zu dem vorangegangenen Beispiel erfolgt. Es sind deutlich ausgeprägte Maxima der Beschleunigung bei bestimmten Gitterweiten zu sehen, da durch die Vervielfachung der Empfänger bei kleinen Zellengrößen in diesem Fall der Aufwand der Empfängerdetektion überproportional steigt. Aus diesem Grund ist die Verwendung ausgedehnter Empfänger mit einer räumlichen Partitionierungstechnik zu vermeiden.

## 4.6 Der 'Punkt in Polygon' Test

Mit dem vorgestellten BSP Baum-basierten Algorithmus konnte die komplexe Operation der Schnittpunktsuche im dreidimensionalen Raum auf den einfachen Test eines Punktes in einem zweidimensionalen Polygon zurückgeführt werden. Der 'Punkt in Polygon' Test stellt eine Standardoperation im Bereich der Computergraphik dar und somit existiert eine Reihe von Algorithmen zu dessen effizienter Berechnung. Einige der Algorithmen setzen die Konvexität des Polygons voraus, wobei nicht konvexe Polygone durch eine Triangulation in Dreiecke zerlegt werden können. Im Folgenden wird ein einfacher 'Punkt in Polygon' Algorithmus vorgestellt, der ebenfalls für nicht konvexe Polygone geeignet ist. Der Algorithmus erlaubt eine Klassifizierung des Punktes  $s$  in die Menge der inneren Punkte

$I$ , der äußeren Punkte  $O$ , der Kantenpunkte  $E$  und der Eckpunkte  $V$ . Eine ausführliche Beschreibung des Algorithmus ist in O'ROURKE (1998) zu finden. Eine alternative Methode, basierend auf zweidimensionalen mengentheoretischen BSP Bäumen, wird im folgenden Abschnitt 4.7 vorgestellt.

Da sich verschiedene Polygone in einer Ebene auch überlappen können, erhält jedes Polygon eine von dem Benutzer bei der Geometrieingabe definierte Priorität zugewiesen. Entsprechend ihrer Priorität werden Polygone bei einem Mehrfachtreffer als sichtbar oder als verdeckt erkannt. Auf diese Weise können zum Beispiel Fenster oder Türen modelliert werden, indem das Fensterpolygon auf das die Wandpolygon gelegt wird. Damit das Fenster bei einem Treffer eindeutig erkannt wird, muss es mit einer höheren Priorität als das Wandpolygon versehen werden. Um die spätere Schnittpunktsuche zu beschleunigen, werden Polygone in der absteigenden Reihenfolge ihrer Priorität in die Polygonliste des Knotens einsortiert. Da diese Liste bei der Schnittpunktsuche sequentiell in dieser Reihenfolge durchsucht wird, kann somit gewährleistet werden, dass das erste getroffene Polygon auch das mit der höchsten Priorität ist. Um den Schnittpunkttest noch weiter zu beschleunigen, wird in jedem Knoten des BSP Baumes eine planare *Bounding Box* verwaltet, die die Gesamtheit aller in der Liste enthaltenen Polygone umschließt. Durch das vorgezogene Testen der Bounding Box wird die Anzahl der tatsächlich durchzuführenden Polygontests erheblich reduziert. Bei dem Beispiel im Bild 4.9 zur Erklärung des Algorithmus wird auf die Darstellung der Bounding Box verzichtet.



**Bild 4.9:** Der 'Punkt in Polygon' Test bestimmt die Anzahl der Schnittpunkte der Polygonkanten  $e_k$  mit einer beliebigen, durch den Punkt  $s$  gehenden Linie (hier z.B. durch die horizontale Achse dargestellt). Aus dem Vergleich der Parität wird auf die Lage von  $s$  relativ zu den Polygonkanten geschlossen.

Die Realisierung des 'Punkt in Polygon' Tests basiert auf der Berechnung der Anzahl der Schnittpunkte einer beliebigen, durch den Punkt  $s$  gehenden Linie mit allen Kanten  $e_k$  des Polygons. Als Schnittlinie wird häufig eine der Koordinatenachsen eingesetzt. In dem im Bild 4.9a dargestellten Fall befindet sich der Punkt  $s$  innerhalb des Polygons. Die Berechnung der Schnittpunkte der horizontalen Achse mit den Polygonkanten ergibt die Punkte  $r_1 \dots r_3$  auf der positiven Seite und den Punkt  $l_1$  auf der negativen Seite von  $s$ . Aus der ungeraden Anzahl der Schnittpunkte  $r_i$  wird auf die Lage des Punktes innerhalb

des Polygons geschlossen. Der im Bild 4.9b dargestellte Fall führt zu einer geraden Anzahl der Schnittpunkte auf beiden Seiten. In diesem Fall liegt der Punkt  $s$  außerhalb des Polygons. Für Punkte auf den Polygonkanten erhält man unterschiedliche Paritäten auf beiden Seiten von  $s$ . Punkte, die mit den Polygonecken  $p_k$  zusammenfallen, können auf diese Weise jedoch nicht erkannt werden. Falls der Punkt  $s$  mit einem der Eckpunkte  $p_k$  übereinstimmt, muss dies durch einen vorgezogenen direkten Vergleich erkannt werden. Der so beschriebene 'Punkt in Polygon' Test ist auf Polygone mit einer fast beliebigen Form anwendbar. Die einzige Einschränkung besteht darin, dass das Polygon nicht sich selbst überschneiden darf.

## 4.7 Darstellung von Mengen mit BSP Bäumen

Der 'Punkt in Polygon' Test kann auch mit Hilfe binärer Partitionierungsbäume in der Ebene durchgeführt werden. Es ist zu erwarten, dass ein solcher Algorithmus insbesondere bei komplexen Polygonen mit einer Vielzahl von Kanten schneller als der obengenannte klassische Test ist, da eine lineare Abfrage von der Ordnung  $\mathcal{O}(N)$  durch eine Baumoperation, unter der Voraussetzung eines balancierten Baumes, mit der Ordnung  $\mathcal{O}(\log(N))$  ersetzt wird. Um einen solchen Test durchführen zu können, müssen die mengentheoretischen Attribute 'Innen' und 'Außen' in die Struktur des binären Partitionierungsbaumes integriert werden. Ebenfalls müssen Algorithmen zur Durchführung der grundlegenden Mengenoperationen bereitgestellt werden, mit denen verschiedene, durch BSP Bäume dargestellte Objekte miteinander kombiniert werden.

Eine Realisierung verschiedener Mengenoperationen ist bereits mit den klassischen diskreten Binärbäumen möglich. Algorithmen zur Durchführung von Mengenoperationen mit diskreten Bäumen stellen ein nachwievor sehr aktives Forschungsgebiet dar, da sie schnelle Suchoperationen mit großen Datenbanken, wie sie z.B. in Internet-Suchmaschinen benötigt werden, ermöglichen. Zu den neusten Beiträgen auf diesem Gebiet gehört die Repräsentation großer Datenmengen mit zufällig balancierten Bäumen (*Tree+Heap* (TREAP)) (SEIDEL und ARAGON, 1996) sowie die Durchführung der mengentheoretischen Suchoperation mit diesen Bäumen (BLELLOCH und REID-MILLER, 1998).

Im Folgenden wird die mengentheoretische Betrachtungsweise des klassischen Binärbaums auf die reelle Achse ausgeweitet und ein Algorithmus zur Durchführung von Mengenoperationen mit BSP Bäumen im eindimensionalen Raum vorgestellt. In einem weiteren Schritt wird dieser Algorithmus auf BSP Bäume in beliebig dimensionalen Räumen verallgemeinert.

### 4.7.1 Diskrete Binärbäume

Ein klassischer diskreter Binärbaum ist im Bild 4.10(b) dargestellt. Mit der Bezeichnung *diskreter Baum* soll der Aufbau des Baumes aus einer abzählbar endlichen Menge diskreter Elemente hervorgehoben werden<sup>1</sup>. Jeder Knoten des Binärbaumes ist mit einer Partitionierung der geordneten Menge  $S_i$  durch ein Element  $E_i \in S_i$  in drei disjunkte Teilmengen  $S_{i+1}^+$ ,  $E_i$  und  $S_{i+1}^-$  verbunden. Der Index  $i$  identifiziert die Position der Menge in der Hierarchie des Baumes. Durch das Ordnungskriterium der Menge kann eine eindeutige Zuordnung jedes Elements in eine der beiden Teilmengen durchgeführt werden. Dazu wird geprüft, ob das Element  $X$  kleiner oder größer als das zur Partitionierung eingesetzte

<sup>1</sup>Im Gegensatz dazu konstruieren allgemeine binäre Partitionierungsbäume eine Baumstruktur über einer unendlichen kontinuierlichen Menge, z.B. der reellen Achse  $\mathbb{R}^0$ , und verallgemeinern damit den Begriff 'Binärbaum'.

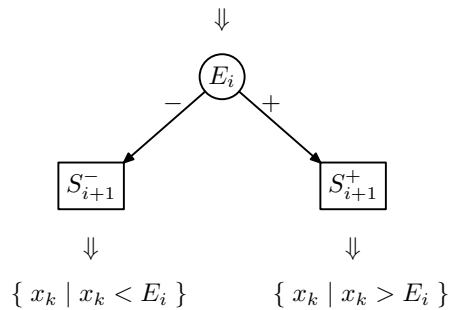
Element  $E_i$  ist. Damit gilt  $\forall X \in S_i$  auch  $X \in \{S_{i+1}^-, E_i, S_{i+1}^+\}$  und

$$S_i^+ \cap S_i^- = \emptyset \quad (4.8)$$

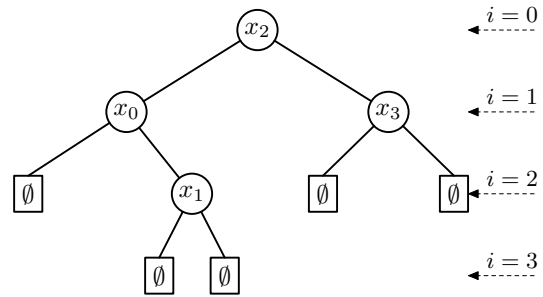
$$S_{i+1}^- \cup E_i \cup S_{i+1}^+ = S_i . \quad (4.9)$$

Der Partitionierer  $E_i$  ist nicht in  $S_{i+1}^+$  und  $S_{i+1}^-$  enthalten und kann daher nicht noch einmal im Baum auftreten. Der Schritt der Partitionierung der Menge  $S_i$  ist im Bild 4.10(a) schematisch dargestellt.

$$S_i = \{x_1 \dots x_N \mid x_k < x_{k+1}\} \wedge E_i \in S_i$$



(a) Binäre Partitionierung



(b) Klassischer Binärbaum

**Bild 4.10:** Die Konstruktion eines klassischen Binärbaumes durch die Partitionierung einer diskreten Menge mit abzählbar vielen Elementen. Das Ordnungskriterium der Elemente legt die Zugehörigkeit der Elemente zu einer der beiden Teilmengen bezüglich des Partitioniers fest.

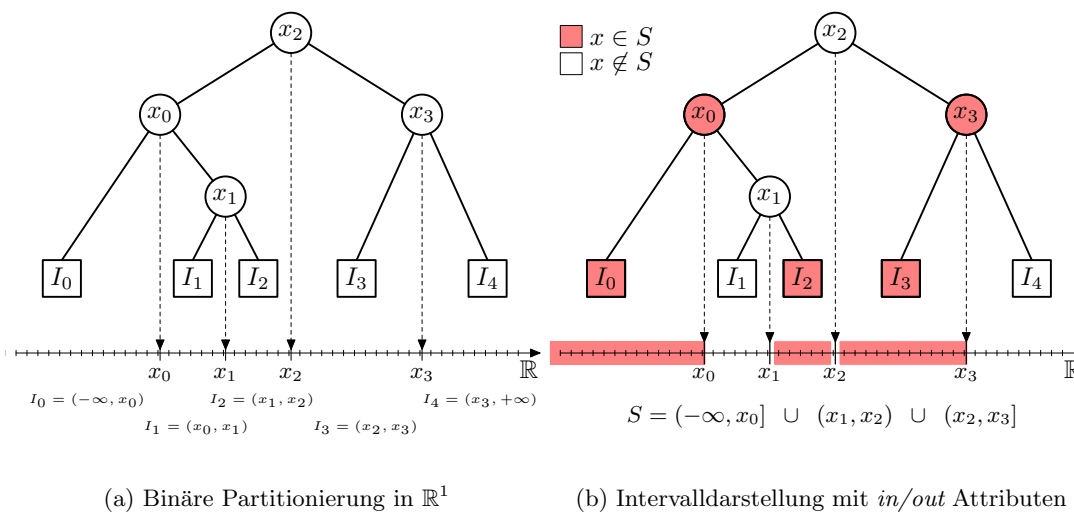
Der Partitionierer  $E_i$  stellt den Knoten des Baumes auf der  $i$ -ten Stufe der Hierarchie dar. Die beiden Teilmengen  $S_{i+1}^+$  und  $S_{i+1}^-$  werden, sofern sie nicht leer sind, rekursiv weiter partitioniert, und stellen anschließend den Teilbaum auf der jeweiligen Seite von  $E_i$  dar. Die Blätter eines klassischen diskreten Binärbaumes enthalten keine Elemente und werden demnach immer durch eine leere Menge  $\emptyset$  repräsentiert.

Durch eine geeignete Wahl des Elements  $E_i$  kann die Menge  $S_i$  so aufgeteilt werden, dass sich die Anzahl der Elemente in beiden Teilmengen nur um eins unterscheidet. Wird dieses Kriterium während der gesamten Partitionierung beibehalten, so unterscheidet sich die Höhe der Blätter um maximal  $\pm 1$  von der mittleren Baumhöhe. Damit ist die maximale Anzahl der Elemente  $N$  in einem Baum der Höhe  $h$  gleich  $2^{(h+1)} - 1$ . Die Anzahl der Vergleichsoperationen, die nötig sind, um eines der Elemente zu finden, ist maximal gleich  $h + 1$  und damit von der Ordnung  $\mathcal{O}(\log(N))$ . Sofern die Menge keine innere Struktur besitzt, die für eine Hash-basierte Suche ausgenutzt werden kann, stellen binäre Bäume die schnellsten Suchverfahren dar. Die Voraussetzung ist jedoch stets das Vorhandensein eines balancierten Baumes mit der kleinsten maximalen Höhe. Für klassische diskrete Binäräume existieren Verfahren, die einen bereits existierenden beliebigen Binärbaum in einen balancierten Binärbaum überführen (WIRTH, 2000). Solche Verfahren sind zur Zeit für den allgemeinen BSP Baum nicht bekannt.



### 4.7.2 Binäre Partitionierungsbäume in $\mathbb{R}^1$

Die Erweiterung des klassischen diskreten Binärbaumes zu einem binären Partitionierungsbaum über der reellen Achse besteht darin, alle Elemente der unendlichen Menge, die zwischen zwei benachbarten Partitioniererelementen  $x_i$  und  $x_{i+1}$  liegen, den entsprechenden Blättern des Baumes zuzuordnen. In diesem Fall repräsentiert jedes Blatt des Baumes ein offenes Intervall über der reellen Achse (Bild 4.11a). Die Linie der reellen Zahlen  $\mathbb{R}$  (Dimension 1) wird durch die Partitionierer  $\{x_i \in \mathbb{R} \mid x_i < x_{i+1}\}$  der Dimension 0 in  $n$  Intervalle aufgeteilt. Durch einen einfachen Vergleich einer beliebigen Zahl  $y$  mit dem Partitionierer  $x_i$  des jeweiligen Knotens kann schnell entschieden werden, ob sich  $y$  in dem positiven Teilbaum oder in dem negativen Teilbaum befindet oder mit dem Partitionierer  $x_i$  übereinstimmt.



**Bild 4.11:** Darstellung der reellen Achse mit einem BSP Baum. Die Partitionierer  $x_i$  teilen die geordnete Menge in endlich viele Intervalle ein, die durch die Blätter des Baumes repräsentiert werden. Durch die Zuweisung des Attributs *in* oder *out* zu jedem Element (und damit jedem Knoten des Baumes) können durch einen BSP Baum beliebige Mengen beschrieben werden.

Eine vollständige mengentheoretische Beschreibung der Eigenschaften des jeweiligen Raumes entsteht erst durch die Zuweisung des Attributes *in* oder *out* zu jedem Knoten des Baumes (inklusive der Blätter), um die Zugehörigkeit der durch diese Knoten beschriebenen Teilmengen eindeutig zu bestimmen. Dadurch werden alle Elemente des Raumes in die Gruppe der Elemente in der Menge (*in*) und die Gruppe der Elemente außerhalb der Menge (*out*) aufgeteilt. Die Zuweisung zusätzlicher Attribute zu den Elementen in der Menge ist dann ebenfalls möglich. Solche Attribute können z.B. die Materialparameter der durch die Elemente der Menge repräsentierten Raumpunkte definieren. Eine Darstellung der Menge

$$S = (-\infty, x_0] \cup (x_1, x_2) \cup (x_2, x_3]$$

ist im Bild 4.11(b) zu sehen. Die Attribute der Knoten und damit die Zugehörigkeit der mit ihnen definierten Elemente zu der Gesamtmenge aller Punkte im Intervall ist farblich hervorgehoben.



### 4.7.3 Mengenoperationen mit BSP Bäumen in $\mathbb{R}^1$

Die Verschmelzung (*Merging*) von BSP Bäumen kann zur Realisierung von beliebigen Mengenoperationen eingesetzt werden (THIBAUT und NAYLOR, 1987). Der Algorithmus besteht aus drei aufeinanderfolgenden Phasen, die mit allen Teilbäumen der beiden Operanden  $A$  und  $B$  durchgeführt werden. Die Phasen sind:

1. Partitionierung eines Baumes mit einem Partitionierer (*Splitting*),
2. Vereinfachung durch eine Mengenoperation mit einem Blatt (*Simplification*),
3. Entfernung redundanter Knoten (*Pruning*).

In der ersten Phase wird der Baum  $A$  mit dem Partitionierer  $b_0$  aus dem Wurzelknoten des Baumes  $B$  in die Teilbäume  $A^+$ ,  $A^-$  und  $A^0$  aufgeteilt. Die Teilbäume  $A^+$  und  $A^-$  enthalten all die Elemente der Gesamtmenge  $A$ , die sich auf der positiven oder der negativen Seite des Partitionierers  $b_0$  befinden. Die Teilmenge  $A^0$  enthält alle Elemente, die mit der Menge des Partitionierers  $B^0$  zusammenfallen. Durch diese Aufteilung liegt  $A^+$  auf der positiven Seite von  $b_0$  und muss daher nur mit den Elementen des positiven Teilbaumes  $B^+$  verknüpft werden. Gleiches gilt für  $A^-$  und  $B^-$  bzw.  $A^0$  und den Teilraum des Partitionierers  $B^0$ . Die Aufteilung wird rekursiv so lange durchgeführt, bis eines der beiden Operanden ein Blatt des BSP Baumes erreicht, wo keine weitere Unterteilung möglich ist.

Da die durch das Blatt beschriebene Teilmenge homogen ist, kann in der zweiten Phase die Mengenoperation das Attribut des Blattes auf alle Knoten des anderen Teilbaumes anwenden. Besitzt z.B. das Blatt das Attribut *in* und ist als Operation die Vereinigungsmenge  $\cup$  beider Bäume gesucht, so ist das Ergebnis der Operation in jedem Fall *in* der Menge und wird ebenfalls durch ein Blatt mit diesem Attribut beschrieben. Die Struktur des anderen Teilbaumes geht durch diese Operation verloren, da alle Punkte, die das Attribut *out* hatten, nach der Operation das Attribut *in* haben und nun ebenfalls zur Menge gehören. Deswegen kann der Teilbaum verworfen werden und das Blatt als das Ergebnis der Operation zurückgegeben werden. Die Rekursion wird an dieser Stelle beendet und die Ausführung der Operation mit den anderen Teilbäumen fortgesetzt. Nach der Rückkehr des rekursiven Aufrufs für die beiden Teilbäume muss die Mengenoperation auch noch mit  $A^0$  und dem Knoten des Partitionierers  $B^0$  durchgeführt werden.

Letztlich kann in der dritten Phase das Ergebnis der rekursiven Operationen mit beiden Teilbäumen und dem Partitionierer der gleichen Menge angehören. So ist es möglich, dass sowohl beide Blätter als auch der Knoten identische Eigenschaften besitzen, da sie z.B. durch die Mengenoperation das Attribut *out* erhalten haben. Ein solcher Teilbaum enthält keine zusätzliche Information und kann ohne Einschränkung durch ein Blatt mit dem Attribut *out* ersetzt werden. Das Entfernen redundanter Information (*Pruning*) aus dem BSP Baum ist ein sehr wichtiger Faktor für die Effizienz des resultierenden BSP Baumes.

Das Ergebnis der Mengenoperation mit BSP Bäumen ist wieder ein BSP Baum. Der Algorithmus dem Diagramm 4.12 zu entnehmen. Im Folgenden wird auf die einzelnen Schritte des Algorithmus und deren Realisierung genauer eingegangen.

### 4.7.4 Partitionierung des BSP Baumes mit einem Partitionierer (*Splitting*)

Die Voraussetzung für die Aufteilung eines  $N$ -dimensionalen BSP Baumes  $A$  mit einem Partitionierer  $b$  der Dimension  $N - 1$  ist ein Vergleich der relativen Lage und der Orientierung der Partitionierer in  $A$  mit  $b$ . Im eindimensionalen Fall ist ein solcher Vergleich durch die Operationen  $<$ ,  $>$  oder  $=$  sehr einfach realisierbar. In höherdimensionalen Räumen kann die relative Lage der Partitionierer in sieben verschiedenen Sonderfällen behandelt

Mengenoperationen in  $\mathbb{R}^1$  — Rekursion

<b>setop(BSP A, Operation X, BSP B)</b>	
Diese Funktion führt die Mengenoperation $X$ mit den beiden Argumenten $A$ und $B$ durch. Das Ergebnis wird in $B$ zurückgegeben. Die Operation wird <i>in-place</i> durchgeführt, wodurch die beiden Operanden zerstört werden. Die Funktion durchquert rekursiv den Baum $B$ und spaltet solange den Baum $A$ mit dem Partitionierer $b$ aus $B$ in Teilbäume auf, bis mindestens eines der Operanden nur aus einem Blatt besteht. Die Durchführung der Mengenoperation mit der homogenen Menge des Blattes und einem Teilbaum sowie die anschließende Entfernung redundanter Knoten beenden die Rekursion.	
Sind $A$ und $B$ innere Knoten des BSP Baumes ?	
J	N
Spalte den Baum $A$ mit der Funktion <b>split()</b> und dem Partitionierer $b$ in die Teilbäume $C^+$ , $C^-$ und der Menge über dem Partitionierer $C^0$ auf. Die Bereiche der Teilbäume decken sich mit den Bereichen von $B^-$ , $B^+$ und $B^0$ .	Rufe <b>simplify(A,X,B)</b> auf, um die Mengenoperation $X$ der durch den Baum dargestellten Menge mit dem durch das Blatt beschriebenen homogenen Teilraum zu berechnen. Der Baum $B$ enthält das Ergebnis der Operation, welches unter Umständen aus nur einem Blatt bestehen kann.
Die rekursiven Aufrufe <b>setop(C<sup>-</sup>,X,B<sup>-</sup>)</b> , <b>setop(C<sup>+</sup>,X,B<sup>+</sup>)</b> und <b>setop(C<sup>0</sup>,X,B<sup>0</sup>)</b> berechnen die Mengenoperation mit den disjunkten Teilräumen von $B$ . Die Ergebnisse der Operation werden in dem veränderten zweiten Argument der Funktion <b>setop()</b> zurückgegeben. Auf die Bedeutung der Mengenoperation von $C^0$ mit $B^0$ wird im folgenden Abschnitt genauer eingegangen.	
Rufe die Funktion <b>prune(B)</b> auf, um redundante Knoten aus dem Ergebnisbaum zu entfernen. Das Argument wird durch die Funktion modifiziert.	
return	

Bild 4.12: Durchführung einer Mengenoperation mit BSP Bäumen.

werden (Abschnitt 4.7.7). Die rekursive Funktion für den eindimensionalen Fall ist im Diagramm 4.13 angegeben. Mit  $a$  ist der Partitionierer des jeweiligen Knotens  $A^0$  gemeint,  $A^+$  und  $A^-$  sind die beiden Teilbäume des Knotens. Nach der Aufteilung stehen in  $C^+$  und  $C^-$  die BSP Darstellungen der Mengen auf den beiden Seiten von  $b$ .  $C^0$  enthält die Elemente der Menge über dem Partitionierer  $b$ .

4.7.5 Vereinfachung durch Mengenoperation (*Simplification*)

Mit der *Simplification* ist die Durchführung der eigentlichen Mengenoperation über der regulären Menge<sup>2</sup> des durch die *Splitting* Phase eingegrenzten gemeinsamen Teilraumes gemeint. Die Mengenoperation setzt zwei Argumente voraus, wobei eines der beiden Argumente aus je einem Blatt (homogene Menge) bestehen muss. Das Ergebnis der Operationen für beliebige Kombinationen der Operanden kann der Tabelle 4.1 der Vereinfachungsregeln entnommen werden (THIBAUT und NAYLOR, 1987). Alle Operationen, die auf regulären Mengen definiert sind, werden mit dem Stern  $*$  bezeichnet. Die Inversion  $\sim^*$  eines BSP Baumes beschreibt die Bildung der zur Menge  $A$  komplementären regulären Menge  $\sim^* A$ . Diese Operation kann leicht durch die rekursive Umwandlung aller Knoten mit dem Attribut *in* in Knoten mit dem Attribut *out* und umgekehrt realisiert werden.

<sup>2</sup>Eine Menge ist regulär, wenn sie neben ihren inneren Elementen ebenfalls die Elemente ihrer Berandung enthält.

Mengenoperationen in  $\mathbb{R}^1$  — Splitting

<b>split(BSP <math>A</math>, Partitionierer <math>b</math>, BSP <math>C^-</math>, BSP <math>C^0</math>, BSP <math>C^+</math>)</b> Rekursive Funktion zum Aufteilen des eindimensionalen BSP Baumes $A$ mit dem reellen Partitionierer $b$ in die Bäume $C^-$ und $C^+$ und die Menge mit dem Punkt $C^0$ .			
Ist $A$ ein Knoten des BSP Baumes ?		N	
Vergleich der Partitionierer $a$ und $b$		$C^- = C^0 = C^+ = \text{Blatt}$	
$b = a$	$b < a$		$b > a$
$C^- = A^-, C^0 = A^0$ $C^+ = A^+$	$C^+ = A$ $\text{split}(A^-, b, C^-, C^0, A^-)$		$C^- = A$ $\text{split}(A^+, b, A^+, C^0, C^-)$
return			

**Bild 4.13:** Partitionierung eines eindimensionalen BSP Baumes mit einem Partitionierer der Dimension 0. Die Operation wird *in-place* durchgeführt, d.h. der Operand  $A$  wird durch die Aufteilung zerstört.

**4.7.6 Entfernung redundanter Knoten (Pruning)**

Die *Pruning* Phase wird während des aufsteigenden Astes der Rekursion angetreten. Das Ziel der Operation ist die Entfernung aller redundanten Information aus dem BSP Baum. Redundante Information äußert sich durch das Vorhandensein von Teilbäumen, die die Elemente einer und derselben Menge enthalten und somit keine weiteren Strukturen der Menge beschreiben. Gehörten zum Beispiel die Punkte im Intervall  $I_4$  (Bild 4.11b) ebenfalls zur Menge, so würde der gesamte Teilbaum mit dem Knoten  $x_3$  keine zusätzliche Information darstellen und könnte durch ein Blatt  $I_n$  mit dem Attribut *in* ersetzt werden. Voraussetzung für diese Ersetzung ist allerdings, dass neben der Mengenzugehörigkeit auch alle anderen Attribute der Elemente der durch das Baum beschriebenen Teilmengen identisch sind (so z.B. die Materialparameter). Durch die einfache *Pruning* Operation wird die Struktur des Baumes vereinfacht und Teilbäume entfernt, die durch die vorangehende Mengeoperation überflüssig geworden sind. Damit ist diese Operation besonders für die Konstruktion effizienter BSP Bäume von Bedeutung. Der einfache Algorithmus ist im Diagramm 4.14 zu sehen. Es bleibt zu erwähnen, dass bereits während der *Simplification* Phase abhängig von der durchgeführten Operation ein Teil der redundanten Teilbäume entfernt wird.

**4.7.7 Verallgemeinerte BSP Bäume in  $\mathbb{R}^n$**

Durch die Erweiterung des BSP Baumes in den dreidimensionalen Raum können Mengenoperationen mit beliebigen Polyedern durchgeführt werden. Der Vorteil der Darstellung der Geometrie in der Form eines BSP Baumes, verglichen mit anderen Formen (BREP), liegt unter anderem darin, dass sich mit einem BSP Baum leicht auch offene Mengen wie Halbräume darstellen lassen (die Menge  $S$  im Bild 4.11 ist ein Beispiel dafür in  $\mathbb{R}^1$ ). Dies ist mit einer polygonalen Oberflächenbeschreibung der Geometrie ohne Weiteres nicht möglich.

Eine Verallgemeinerung des eindimensionalen BSP Baumes auf beliebig dimensionale Räume bringt jedoch zwei Schwierigkeiten mit sich. Die erste Schwierigkeit offenbart sich

Operation	Symbol	linker Operand	rechter Operand	Ergebnis
Vereinigungsmenge	$\cup^*$	$S$	$in$	$in$
		$S$	$out$	$S$
		$in$	$S$	$in$
		$out$	$S$	$S$
Schnittmenge	$\cap^*$	$S$	$in$	$S$
		$S$	$out$	$out$
		$in$	$S$	$S$
		$out$	$S$	$out$
Differenzmenge	$-^*$	$S$	$in$	$out$
		$S$	$out$	$S$
		$in$	$S$	$\sim^* S$
		$out$	$S$	$out$

**Tabelle 4.1:** Regeln zur Vereinfachung der Mengenoperationen einer regulären Menge  $S$  mit einem homogenen Teilraum. Entnommen aus [Thibault und Naylor \(1987\)](#)

in der *Splitting* Phase während des Vergleichs der nun mehrdimensionalen Hyperebenen. Dieser Vergleich ist nicht mehr durch die einfachen Vergleichsoperationen zu bewerkstelligen. Vielmehr muss die gegenseitige Orientierung der Partitionierer im Raum beachtet werden. Die `split()` Funktion kann durch die Aufteilung und die getrennte Behandlung von sieben verschiedenen Sonderfällen der gegenseitigen Orientierung der Hyperebenen realisiert werden. Auf die Realisierung wird im Folgenden noch genauer eingegangen.

Die zweite Schwierigkeit hängt ebenfalls mit dem Partitionierer zusammen und besteht in der Darstellung der Menge aller Punkte  $A^0$  über dem Partitionierer. Im eindimensionalen Fall bestand diese Menge lediglich aus einem Punkt, dem das Attribut *in* oder *out* zugeordnet wurde. In höherdimensionalen Räumen definiert jeder Partitionierer jedoch eine unendliche Punktemenge. Es muss also eine geeignete Darstellung der Struktur der Menge im Bereich des Partitionierers gefunden werden.

#### 4.7.8 Multidimensionale BSP Bäume

Gerade das letztgenannte Problem führt zu einer Verallgemeinerung des BSP Baumes zu einer Hierarchie von BSP Bäumen. In einem  $N$ -dimensionalen Raum wird die Struktur, außer in den Hyperebenen, durch einen BSP Baum der Dimension  $N$  vollständig beschrieben. Die Struktur in den Hyperebenen kann aber, außer in den Hyperlinien, durch einen  $N - 1$  dimensionalen BSP Baum beschrieben werden. Die Struktur der Hyperlinien, die als  $N - 2$  dimensionale Partitionierer des Baumes der Hyperebene angesehen werden, kann wiederum mit BSP Bäumen einer noch niedrigeren Dimension vollständig beschrieben werden. Das Konzept lässt sich so bis  $\mathbb{R}^0$  fortsetzen:

$$\mathbb{R}^N \rightarrow \mathbb{R}^{N-1} \rightarrow \dots \rightarrow \mathbb{R}^0 . \quad (4.10)$$

Die Struktur des Baumes besteht nach wie vor aus inneren Knoten, die den Partitionierer enthalten, und den Blättern, die die Struktur der Teilräume beschreiben. Zusätzlich zu dem Partitionierer enthalten jedoch die Knoten auch einen BSP Baum mit der Dimension und dem lokalen Koordinatensystem des Partitionierers. Im zweidimensionalen Fall würde jeder Knoten des 2D BSP Baumes einen weiteren Baum der Dimension  $N = 1$  enthalten, der die Struktur der Punktemenge auf der Partitionierungslinie beschreiben würde. Jeder Knoten des eindimensionalen Baumes enthält neben dem Partitionierer  $x_i$

Mengenoperationen in  $\mathbb{R}^1$  — Pruning

<b>prune(BSP A)</b>	
Diese Funktion prüft, ob die Mengen der beiden Teilbäume $A^+$ und $A^-$ zusammen mit der Menge der Elemente des Knotens $A^0$ eine zusammenhängende Menge darstellen. Ist dies der Fall, so kann der durch diesen Knoten beschriebene Teilbaum durch ein einziges Blatt mit identischen Attributen ersetzt werden.	
J	Sind $A^+$ und $A^-$ Blätter des BSP Baumes ?
J	Besitzen $A^+$ und $A^-$ sowie $A^0$ gleiche Attribute?
Ersetze den Teilbaum $A$ durch ein Blatt mit den gleichen Attributen. Der Speicher für den Teilbaum $A$ kann freigegeben werden.	
return	

**Bild 4.14:** Die *Pruning* Phase entfernt redundante Information aus dem BSP Baum und stellt somit eine wichtige Funktion in Bezug auf die Effizienz dar.

dann auch einen Baum der Dimension  $N = 0$ , der in dieser Dimension nur aus einem Blatt mit dem Attribut des Punktes  $x_i$  besteht. Der Wert  $x_i$  gibt die Position des Punktes im lokalen Koordinatensystem der eindimensionalen Partitionierungslinie an. Ein einfaches Beispiel zur Veranschaulichung der multidimensionalen BSP-Baum Hierarchie ist im Bild 4.15 dargestellt.

Mit Hilfe der beschriebenen multidimensionalen BSP-Baum Hierarchie kann die Struktur von Räumen einer beliebigen Dimension eindeutig beschrieben werden. Dabei sorgt die Konstruktion des Baumes mit Mengenoperationen für eine konsistente Darstellung der Struktur. Unter der Annahme einer optimalen Konstruktion könnten solche Bäume ebenfalls als eine speichereffiziente hierarchische Repräsentation multidimensionaler Daten angesehen werden. Eine nochmalige Erweiterung des BSP Baumes um die topologischen Verbindungen zwischen Strukturen in verschiedenen Dimensionen erlaubt zudem eine Analyse der Objektform (z.B. Geometrie einer Beugungskante). Auf diese Erweiterung wird im Abschnitt 4.9 genauer eingegangen.

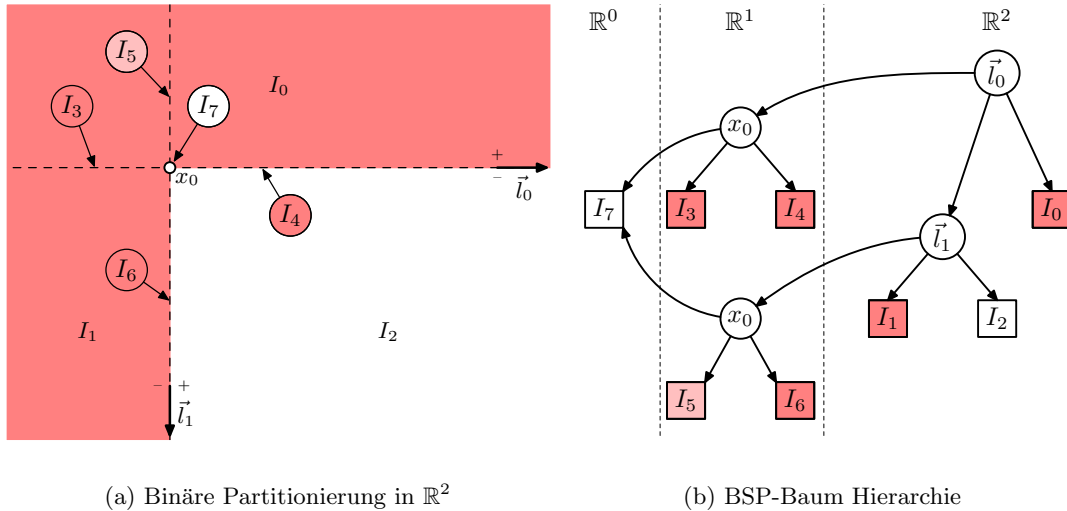
#### 4.7.9 Die *Splitting* Operation in $\mathbb{R}^n$

Während der *Splitting* Phase muss die Orientierung der beiden Partitionierer beachtet werden. Für die Orientierung ergeben sich sieben verschiedene Sonderfälle, die getrennt voneinander behandelt werden können (NAYLOR et al., 1990). Diese Fälle lassen sich in drei Gruppen zusammenfassen:

**I. Beide Partitionierer liegen übereinander ( $a = b$ ).**

In diesem Fall sind die beiden Teilbäume des Knotens bereits das Ergebnis der *Splitting* Phase. Es muss jedoch auch beachtet werden, ob die Partitionierer gleichsinnig oder entgegengesetzt orientiert sind und die Teilbäume entsprechend vertauscht werden müssen. Dieser Fall entspricht dem Fall  $=$  in  $\mathbb{R}^1$ . Die Menge der Punkte über dem Partitionierer ist durch den Baum  $A^0$  des Knotens gegeben. Diese Gruppe enthält nur einen Sonderfall mit zwei verschiedenen Orientierungen der Partitionierer.

**II. Der Partitionierer  $b$  liegt ganz im positiven oder negativen Teilraum von  $a$ .**



**Bild 4.15:** Ein Beispiel der Repräsentation einer einfachen Struktur in der Ebene mit einer mehrdimensionalen BSP-Baum Hierarchie. Die Zugehörigkeit der Punkte zur Menge ist farblich gekennzeichnet. Die Punkte im Bereich der Partitionierer mit Ausnahme des Schnittpunktes  $x_0$  besitzen das Attribut *in*. Alle Attribute befinden sich jetzt ausschließlich in den Blättern der Bäume (Teilmengen  $I_0 \dots I_7$ ). Das Attribut von  $I_5$  ist ohne Belang, da der Baum des Partitionierers  $\vec{l}_1$  nur im negativen Teilraum von  $\vec{l}_0$  gültig ist.

In diesem Fall muss der positive oder der negative Teilbaum mit  $b$  aufgeteilt und eines der Teilbäume entsprechend der gegenseitigen Orientierung von  $a$  und  $b$  den positiven bzw. den negativen Teilbaum ersetzen. Der Knoten und der andere Teilbaum bilden zusammen mit dem Baum  $A^0$ , der während der rekursiven Partitionierung bestimmt wird, das Ergebnis. Dieser Fall entspricht  $<$  und  $>$  in  $\mathbb{R}^1$ . Es existieren zwei verschiedene Sonderfälle mit je zwei Möglichkeiten der Orientierung.

### III. Die Partitionierer schneiden einander.

Diese Gruppe behandelt die vier verschiedenen Fälle der Orientierung, die zwei Partitionierer zueinander besitzen können. Die Definition der Orientierung hängt hierbei von der Definition der Hyperebene ab. Im Falle eines zweidimensionalen Baumes sind die Partitionierer gerichtete Linien. Der Richtungsvektor des einen Partitionierers kann entweder von dem negativen in den positiven Teilraum gerichtet sein oder umgekehrt. Das Gleiche gilt für den zweiten Partitionierer. Diese vier Fälle ergeben die verschiedenen Möglichkeiten der Zusammensetzung der vier zweidimensionalen Teilbäume, die durch die Partitionierung entstehen, zu zwei Bäumen auf beiden Seiten des Partitionierers. Die Lösung des *Splitting* Problems wird ausführlich in [NAYLOR et al. \(1990\)](#) behandelt.

## 4.8 Die 'Punkt in Polygon' Operation in $\mathbb{R}^n$

Die Klassifizierung eines Punktes besteht darin, dem Punkt eindeutig ein Attribut zuzuordnen. Dies könnten zum Beispiel die gesuchten Materialparameter an dem Ort des Punktes sein. Somit ist unter einer Klassifizierung die globale Suche der Zugehörigkeit des Punktes zu einer Menge zu verstehen. Eine solche Suche ist mit dem verallgemeinerten BSP Baum in beliebig dimensionalen Räumen leicht möglich. Da die Lage des Punktes relativ zu der Hyperebene in  $\mathbb{R}^N$  eindeutig ist, kann der Punkt entweder dem positiven Teilbaum oder dem negativen Teilbaum oder der Hyperebene zugeordnet werden. Nach der Klassifizierung wird das Verfahren mit dem entsprechenden Teilbaum weitergeführt, bis ein Blatt des Baumes erreicht worden ist. Liegt der Punkt in der Ebene, so werden die Koordinaten des Punktes in der Dimension  $N$  in die lokalen Koordinaten der Ebene, also in die Dimension  $N - 1$  transformiert. Die Klassifizierung wird mit dem  $N - 1$  dimensionalen Baum der Hyperebene fortgesetzt und ebenfalls erst dann beendet, wenn ein Blatt der multidimensionalen Baumhierarchie erreicht wurde. Das Attribut des Blattes stellt das Ergebnis der Klassifizierung dar. Der Algorithmus ist im Diagramm 4.16 zu sehen. Aus

### BSP-Baum Hierarchie in $\mathbb{R}^n$ — Klassifizierung eines Punktes

<b>search(BSP A, Punkt P)</b>		
Diese Funktion klassifiziert rekursiv den Punkt bezüglich der BSP-Baum Hierarchie und bestimmt so die Zugehörigkeit des Punktes zu einer Teilmenge. Das gesuchte Attribut der Teilmenge wird zurückgegeben.		
J	Ist A ein Blatt des BSP Baumes ?	N
Berechne die Distanz $d$ des Punktes $P$ zu der Hyperebene $a$ .		
Ist die Position des Punktes zur Hyperebene		
$d < -\epsilon$	$-\epsilon \leq d \leq \epsilon$	$d > \epsilon$
Rufe <b>search</b> ( $A^-, P$ ) rekursiv mit dem negativen Teilbaum auf.	Transformiere $P$ in das Koordinatensystem der Hyperebene. Rufe die Funktion <b>search</b> () des $(N-1)$ -dimensionalen Baumes auf.	Rufe <b>search</b> ( $A^+, P$ ) rekursiv mit dem positiven Teilbaum auf.
Gebe das Attribut des Blattes zurück		

**Bild 4.16:** Klassifizierung eines Punktes mit der BSP-Baum Hierarchie.

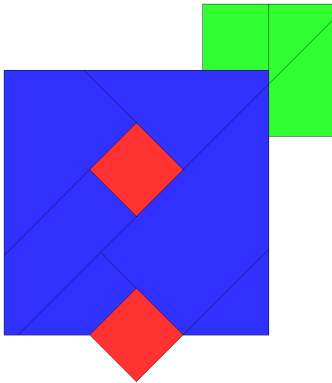
Gründen der numerischen Stabilität wird die Hyperebene mit einer Dicke  $2\epsilon$  angenommen. Die untere Grenze von  $\epsilon$  hängt von der Wortbreite der Fließkomma-Arithmetik des eingesetzten Rechners ab. Die obere Grenze ist durch die Anforderungen an die Detailtreue der Geometrie bestimmt.

Der 'Punkt in Polygon' Test mit Hilfe des verallgemeinerten BSP Baumes entspricht der oben beschriebenen Klassifizierung in  $\mathbb{R}^2$  und ist leicht zu realisieren. Um dem am Anfang des Kapitels beschriebenen Modell der Geometrie treu zu bleiben, müssen bei der Konstruktion des zweidimensionalen BSP Baumes die Prioritäten<sup>3</sup> korrekt beachtet werden. Dies wird durch die Erweiterung der Tabelle 4.1 um eine neue Mengenoperation

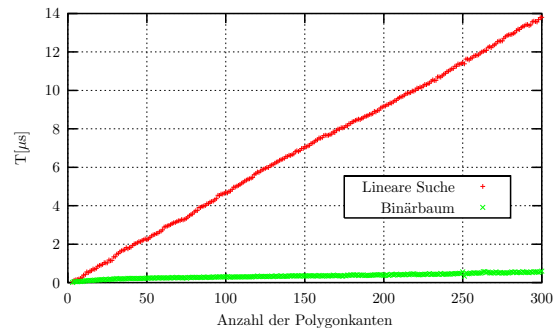
<sup>3</sup>Gemeint sind die Lagen der Polygone. Da sich Polygone überlappen dürfen, wird über die Lage (Priorität) der Polygone festgelegt, welches der Polygone sich 'oben' befindet und von einem Strahl gesehen wird.



erreicht. Die neue Operation ist einfach zu realisieren: Alle *in* Attribute der Teilmenge mit einer niedrigen Priorität werden durch Attribute mit einer höheren Priorität ersetzt. Das Ergebnis einer solchen Operation ist im Bild 4.17a anschaulich dargestellt. Zu sehen sind vier, sich teilweise überlappende, Quadrate mit unterschiedlichen Prioritäten. Die rot gezeichneten Quadrate besitzen die höchste Priorität, das grüne Quadrat besitzt die niedrigste Priorität. Die Partitionierungslinien sind ebenfalls eingezeichnet und geben einen Aufschluss über die Struktur des BSP Baumes.



(a) Verdeckung von Polygonen



(b) Beschleunigung durch BSP Bäume

**Bild 4.17:** Die Verdeckung von Polygonen mit unterschiedlichen Prioritäten kann durch einen zweidimensionalen BSP Baum behandelt werden. Der Teilraum mit der niedrigsten Priorität wird durch Teilräume mit höherer Priorität ersetzt. Die Beschleunigung der BSP-basierten 'Punkt in Polygon' Suche im Vergleich zu dem klassischen Algorithmus aus Abschnitt 4.6 hängt vor allem von der Anzahl der Polygonkanten ab.

Die Beschleunigung der 'Punkt in Polygon' Operation ergibt sich aus dem Vergleich der Laufzeiten der im Abschnitt 4.6 beschriebenen Funktion mit der BSP-basierten Suche. Die gemessenen mittleren Laufzeiten einer Suchoperation sind in Abhängigkeit der Anzahl der Polygonkanten im Bild 4.17b aufgetragen. Die Form des Polygons wurde mit einem Zufallsgenerator erzeugt und war in der Regel nicht konvex. Die Menge der gesuchten Punkte wurde ebenfalls gleichverteilt über die gleiche Fläche erzeugt. Es wurde stets das gleiche Polygon für beide Suchverfahren verwendet. Dem Bild ist zu entnehmen, dass sich der Einsatz der BSP-basierten 'Punkt in Polygon' Suche bereits bei Polygonen mit einer relativ kleinen Anzahl von Kanten lohnt, jedoch erst bei sehr komplizierten Polygonen eine deutliche Geschwindigkeitssteigerung zu erwarten ist. Da die 'Punkt in Polygon' Suche die zentrale Funktion des Ray Tracing Verfahrens bildet, macht sich bereits ein kleiner Geschwindigkeitszuwachs bei dieser Funktion in der Laufzeit der Simulation bemerkbar.

## 4.9 Beschreibung der Topologie mit BSP Bäumen

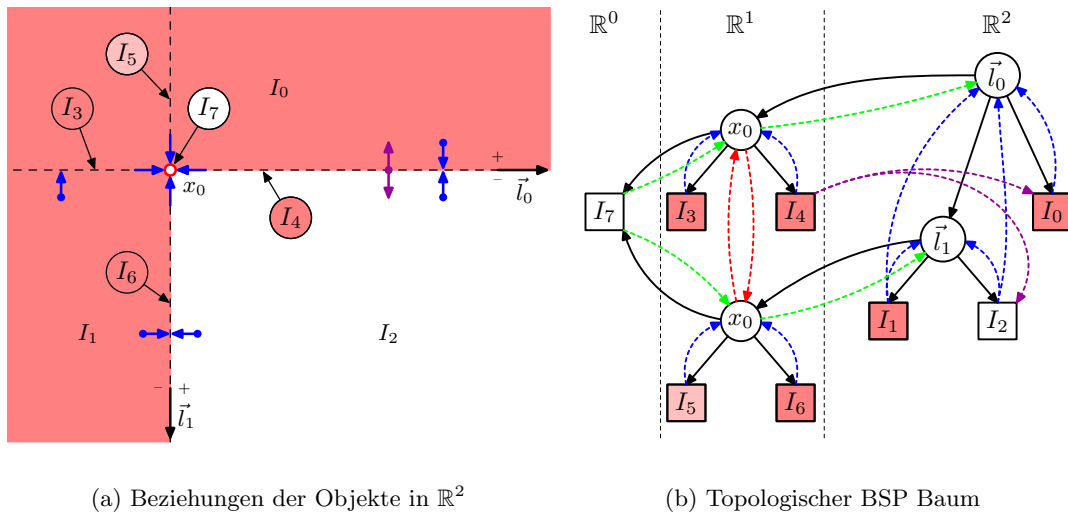
Während der Konstruktion eines BSP Baumes werden Daten über die Nachbarschaftsbeziehungen der Objekte, die als ein Nebenprodukt bei der Verschmelzung von zwei BSP Bäumen anfallen, nicht verwendet. Der so konstruierte Baum enthält somit nur Verweise auf Objekte in einer niedrigeren Ebene der Hierarchie oder Dimension. Durch das Einfügen



geeigneter Zeiger können jedoch auch topologische Beziehungen der Objekte festgehalten und analysiert werden. Solche BSP Bäume werden topologische binäre Partitionierungsbäume (*Topological Binary Space Partitioning Trees* (TBSP)) genannt. Sie können zum Beispiel zur Konvertierung eines BSP Baumes in eine Oberflächendarstellung (COMBA und NAYLOR, 1996) oder die Partitionierung des Raumes in disjunkte konvexe Zellen (FRACH et al., 2000) genutzt werden.

Die topologische Information tritt durch den Vergleich der Partitionierer während der *Splitting* Phase der BSP-Baum Verschmelzung zu tage. Sie beschreibt die Beziehungen von Objekten in benachbarten Dimensionen – z.B. die Zugehörigkeit des Schnittpunktes (Dimension 0) zu den sich schneidenden Partitionierungslinien (Dimension 1) – oder innerhalb der gleichen Dimension, z.B. durch den Verweis auf alle Kopien eines Schnittpunktes in allen BSP Bäumen derselben Dimension. Die komplexen Beziehungen sollen am folgenden Beispiel dargelegt werden. Die unterschiedlichen topologischen Zeiger sind farblich gekennzeichnet. Der Übersichtlichkeit halber wurde auf die Darstellung aller topologischen Zeiger im Bild 4.18 verzichtet.

Der Schnittpunkt von zwei Partitionierern der Dimension  $N$  ergibt einen Partitionierer der Dimension  $N - 1$ , der in den beiden zugehörigen  $(N - 1)$ -dimensionalen BSP Bäumen enthalten ist. So verbindet bei dem im Bild 4.18 gezeigten Beispiel der Schnittpunkt  $x_0$  die beiden Kanten, die durch die Partitionierungslinien  $\vec{l}_0$  und  $\vec{l}_1$  dargestellt werden. Diese

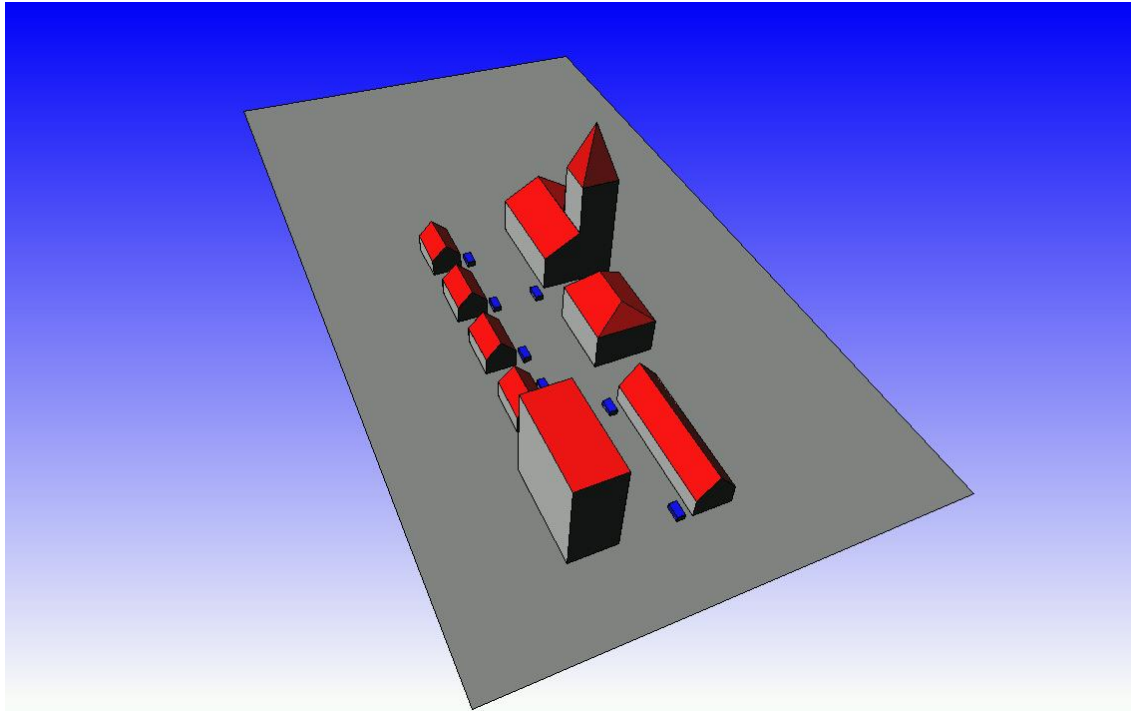


**Bild 4.18:** Erweiterung der BSP-Baum Hierarchie um topologische Zeiger, veranschaulicht am Beispiel der Geometrie aus dem Bild 4.15. Die topologischen Zeiger stellen Beziehungen zwischen Merkmalen der Menge in unterschiedlichen Dimensionen her und erlauben eine Analyse der geometrischen Eigenschaften des Objekts.

Kanten definieren zusammen mit dem Schnittpunkt die Begrenzung der Punktemenge. Durch das Ablegen der rot gezeichneten Zeiger zwischen den beiden Instanzen des Partitionierers  $x_0$  in den Baum bleibt die topologische Information erhalten. Bei mehreren Kanten und Schnittpunkten würden zusätzliche Zeiger auf diese verweisen und eine eindeutige Verbindung der Kanten aufzeigen. Durch den Einsatz einer doppelten Verkettung wird das Durchlaufen der Begrenzung in beiden Umlaufrichtungen ermöglicht. Der so beschriebene Pfad verbindet Knoten des Baumes innerhalb der gleichen Dimension.

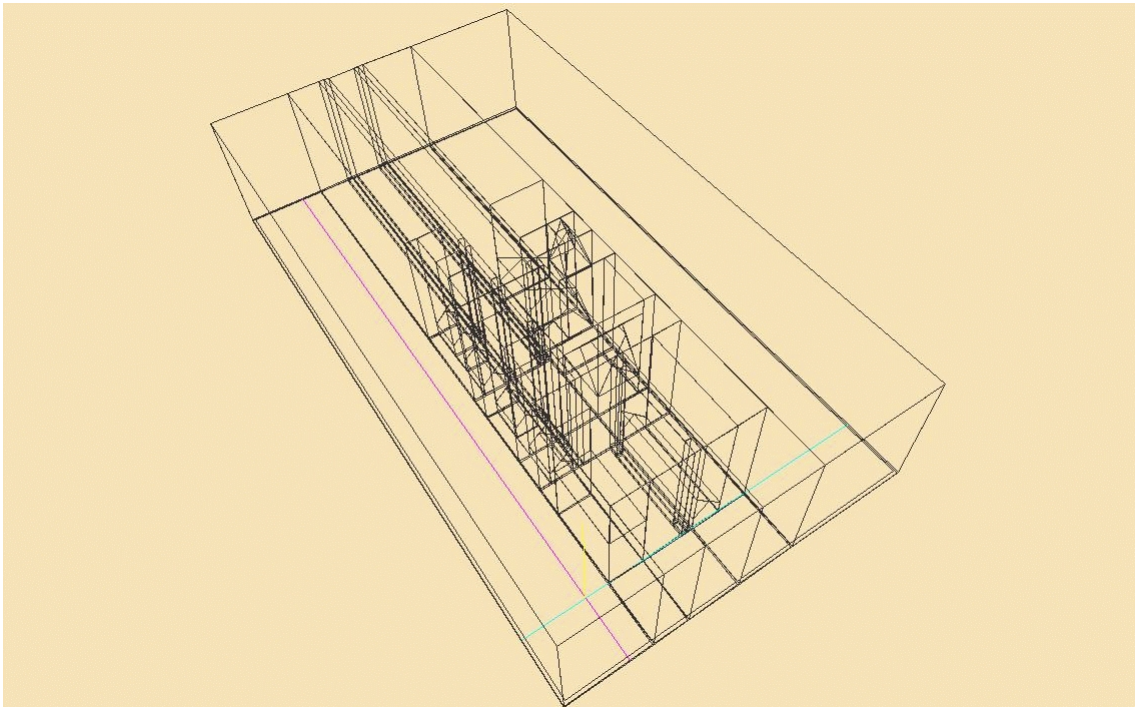
Weitere topologischen Merkmale werden durch die Nachbarschaftsbeziehungen zwischen Objekten in unterschiedlichen Dimensionen festgehalten. Dazu gehören die Beziehungen zwischen den  $N$ -dimensionalen Teilräumen der Blätter der Baumhierarchie und den  $(N - 1)$ -dimensionalen Teilräumen der Partitionierer. Diese Information ist im Bild durch die blau gezeichneten Zeiger dargestellt. Auf diese Weise wird die Verbindung eines höherdimensionalen Teilraumes zu der Menge der Punkte auf der Berandung festgehalten. Die Verbindung in die andere Richtung bleibt erhalten, wenn jedem Blatt des  $(N - 1)$ -dimensionalen Baumes Zeiger auf die anliegenden Blätter des Baumes der Dimension  $N$  zugewiesen werden. Nur zwei solche topologischen Zeiger sind als Beispiel im Bild 4.18 violett eingezeichnet. Die grünen Zeiger verbinden Knoten auf niedrigeren Ebenen der Hierarchie bzw. der Dimension mit ihren Elternknoten und ermöglichen somit die Traversierung des Baumes in umgekehrter Richtung vom Blatt zur Baumwurzel.

Die im TBSP Baum enthaltene topologische Information kann zu der Extraktion bestimmter Merkmale der Geometrie genutzt werden. So wird die polygonale Darstellung einer komplexen Grenzfläche der Menge durch die Verfolgung der topologischen Zeiger im Baum der Dimension  $N = 1$  beschrieben. Die Auswertung derselben Information in  $N = 2$  ergibt die Verbindung benachbarter Polygone zu der gemeinsamen Kante. Mit Hilfe der violett dargestellten Zeiger wird eine Partitionierung der Geometrie in disjunkte konvexe Zellen ermöglicht (Bild 4.20). Die Zellen werden durch die Blätter des dreidimensionalen BSP Baumes beschrieben, die Grenzflächen zwischen den Zellen werden durch die Blätter der zweidimensionalen Bäume gebildet. Die Darstellung der Geometrie als ein Graph,



**Bild 4.19:** Das Modell eines typischen Straßenszenarios, welches für die Untersuchung zur Extraktion konvexer disjunkter Zellen verwendet wurde. Das Modell wurde von [Schöberl \(1997\)](#) zur polarimetrischen Untersuchung der Wellenausbreitung entlang von Straßenzügen erstellt.

dessen Knoten durch homogene konvexe Teilräume gegeben sind und dessen Kanten die zwischen den Teilräumen liegenden gemeinsamen Grenzflächen definieren, bildet die Basis



**Bild 4.20:** Die Konversion des topologischen BSP Baumes der Geometrie in konvexe disjunkte Teilräume (Zellen), wie sie von [Schöberl \(1997\)](#) zur schnellen Schnittpunktsuche eingesetzt werden.

für eine alternative Methode zur Strahlverfolgung, die Ähnlichkeit mit den in [SCHÖBERL \(1997\)](#), [FUNKHOUSER et al. \(1998\)](#) und [ZHANG et al. \(2001\)](#) vorgestellten Verfahren besitzt und im Folgenden kurz erläutert wird.

In einem ersten Schritt muss die Zelle (das Blatt des dreidimensionalen Baumes), in der sich die Strahlenquelle befindet, gefunden werden. Dies wird durch eine Klassifizierung der Position des Senders mit der TBSP-Baum Hierarchie erreicht. Diese Operation wird für jede Quelle nur einmal zu Beginn der Berechnung durchgeführt.

Der von der Strahlenquelle generierte Strahl liegt in der Zelle und muss mit der Grenzfläche der Zelle geschnitten werden. Hierzu werden die im [Bild 4.18](#) blau dargestellten topologischen Zeiger eingesetzt, die eine Verbindung zwischen dem Teilraum und seiner Grenzfläche herstellen. Die Schnittpunkte des Strahls mit allen Ebenen der Grenzfläche werden berechnet. Aufgrund der Konvexität der Zellen wird der Punkt mit der kürzesten Distanz zum Sender als der nächste Schnittpunkt identifiziert und in die lokalen Koordinaten der Ebene transformiert. Durch eine Klassifizierung mit Hilfe des zugehörigen zweidimensionalen BSP Baumes der Ebene wird dem Punkt ein Blatt dieses Baumes zugewiesen<sup>4</sup>. Das Blatt enthält die Verbindung zu der Nachbarzelle, in die der gebrochene Teilstrahl eintritt. Der reflektierte Teilstrahl bleibt wegen der Konvexität des Teilraumes in der Zelle erhalten und kann mit der gleichen Methode weiterverfolgt werden. Die zuletzt getroffene Ebene muss jedoch von der Suche ausgenommen werden. Empfänger werden in der gleichen Art und Weise wie bei einem klassischen BSP Baum behandelt. Die punktförmigen Empfänger werden in den dreidimensionalen Baum einsortiert und stehen in entsprechenden Listen in den Blättern des multidimensionalen BSP Baumes zur Ver-

<sup>4</sup> Sonderfälle durch Punkte auf der Berandung der Polygone müssen gesondert behandelt werden und werden hier vernachlässigt.

fügung. Methoden zur Beschleunigung der Empfängerdetektion (Abschnitt 4.5) können unverändert angewendet werden.

#### 4.10 Zusammenfassung

In diesem Kapitel wurden die Grundlagen der räumlichen binären Partitionierungsbäume (BSP Bäume) erläutert und am Beispiel der Konstruktion eines zweidimensionalen modifizierten BSP Baumes veranschaulicht. Der BSP Baum bildete die Basis für einen Algorithmus zur schnellen Schnittpunktberechnung des linienförmigen Strahls mit einer komplexen polygonalen Geometrie. Er dient ebenfalls zur einer effizienten Ortung der getroffenen Empfänger. Zu einer weiteren Beschleunigung der Empfängerdetektion wurde das Einfügen von virtueller Ebenen vorgeschlagen und analysiert.

Als Kernfunktion der BSP-basierten Schnittpunktsuche wurde der 'Punkt in Polygon' Test identifiziert. Alternativ zu dem klassischen linearen Verfahren zur Klassifizierung eines Punktes relativ zu einem beliebigen Polygon wurde der Einsatz planarer binärer Partitionierungsbäume vorgeschlagen. Voraussetzung ist die Erweiterung des BSP Baumes um die mengentheoretischen Attribute *in* und *out*. Es wurde ein Algorithmus zur Durchführung von Mengenoperationen mit BSP Bäumen vorgestellt und der Baum zu einer multidimensionalen BSP-Baum Hierarchie in  $\mathbb{R}^n$  erweitert. Der 'Punkt in Polygon' Test im zweidimensionalen Raum wurde als ein Sonderfall der Suche nach der Zugehörigkeit des Punktes zu einer Teilmenge betrachtet. Messungen zeigen, dass der BSP-basierte Test bereits bei einfachen Polygonen schneller ist als das klassische Verfahren, jedoch erst bei Polygonen mit vielen Kanten einen deutlichen Geschwindigkeitsvorteil aufweist.

Abschließend wurde die Erweiterung der multidimensionalen BSP-Baum Hierarchie zu einem topologischen BSP-Baum diskutiert. Die Auswertung der topologischen Information, die während der Konstruktion des Baumes generiert wird, kann zur Extraktion von geometrischen Merkmalen genutzt werden. Als Beispiel wurde die Extraktion der Oberflächendarstellung oder die Analyse der Kantengeometrie angeführt. Die ebenfalls mögliche Zerlegung des Raumes in einen gerichteten Graph mit konvexen Zellen führt zu einem neuen Verfahren der Schnittpunktsuche.

---

## Optimierung binärer Partitionierungsbäume

In diesem Kapitel werden Methoden zur Konstruktion optimierter Partitionierungsbäume vorgestellt. Für die Optimierung werden neben verschiedenen heuristischen Verfahren auch die Simulated Annealing Methode und genetische Algorithmen eingesetzt. Die erreichte Beschleunigung wird mit Messungen der Simulationszeiten in realen und künstlich erzeugten Testszenerien belegt. Ein Vergleich der Optimierungsverfahren schließt das Kapitel ab.

### 5.1 Optimale Partitionierungsbäume

Der innere Aufbau eines BSP Baumes ist entscheidend für die Effizienz der späteren Simulation. Die maximale Baumhöhe, definiert als die maximale Anzahl der Knoten zwischen der Baumwurzel und einem Blatt des Baumes, bestimmt die Anzahl der Operationen, die für jeden Strahl während der Schnittpunktberechnungen mit der Geometrie durchgeführt werden müssen (Kapitel 4 Seite 37).

Die Struktur eines BSP Baumes wird im Wesentlichen bestimmt durch die Form der Objekte im Szenario und die Reihenfolge, in der diese Objekte in den Baum eingefügt werden. Im Fall des modifizierten BSP Baumes ist die Geometrie durch die Polygone der Wände definiert. Die Form und die Lage der Polygone zueinander bestimmen sowohl die Wahl der Partitionierungsebenen als auch die Anzahl der Polygonfragmente, die bei einer Konstruktion durch das Aufschneiden der Polygone an den Partitionierungsebenen entstehen. Die Reihenfolge, in der die Polygone in den Baum eingefügt werden, definiert die Aufteilung des Raumes in Teilräume und ist somit ebenfalls für die Baumhöhe verantwortlich.

Da die Geometrie des Szenario vorgegeben ist, kann nur noch die Reihenfolge der Polygone ausgenutzt werden, um die Struktur des Baumes zu optimieren. Die Anzahl der Vergleiche eines Strahls mit den Partitionierungsebenen auf dem Weg von der Wurzel des Baumes in die entsprechenden Blätter ist von vielen Faktoren abhängig, vor allem aber von den Strahlparametern. Um verschiedene Bäume unabhängig von den Strahlparametern miteinander vergleichen zu können, muss zunächst eine geeignete Metrik gefunden werden. Die maximale Anzahl verschiedener BSP Bäume, die für ein Szenario mit  $N$  Polygonen konstruiert werden können, ist durch die  $N!$  mögliche Permutationen der Polygone gegeben. Eine Konstruktion und Bewertung aller Möglichkeiten kommt daher nur für sehr kleine  $N$  in Frage. Bei  $N = 13$  beträgt die Zahl aller möglichen Darstellungen der Geometrie durch einen BSP Baum 6227020800, so dass dieser Fall derzeit die noch praktisch realisierbare Grenze für eine vollständige Suche nach dem optimalen Baum darstellen dürfte. In Wirklichkeit können Szenarien aus mehreren Millionen Polygonen bestehen, für die eine vollständige Suche vermutlich nie realisierbar wird.

Ähnliche Probleme werden in der Komplexitätstheorie als NP-hart bezeichnet ([ATALLAH, 1998](#)), da zur Zeit kein Verfahren existiert, welches das globale Optimum in einer Zeit mit polynomialer Abhängigkeit von  $N$  finden kann. Die Lösungsansätze für diese Problemklasse gliedern sich wie folgt in zwei Kategorien:



## I. Blinde Verfahren

Zu diesen Verfahren zählen vor allem die Heuristiken. Abhängig von den geometrischen Eigenschaften der Objekte (in diesem Fall der Polygone) aus denen der BSP Baum aufgebaut werden soll, wird mit Hilfe einer Heuristik die angenommene optimale Reihenfolge für die Konstruktion des Baumes erzeugt. Eine solche Heuristik kann z.B. die Reihenfolge anhand der Polygonfläche festlegen. Viele andere Heuristiken sind denkbar, einige von ihnen wurden untersucht und werden im Folgenden bewertet. Eine wesentliche Eigenschaft dieser Verfahren ist die fehlende Rückkopplung bei der Optimierung, d.h. es erfolgt keine Überprüfung oder Modifikation der Optimierung durch eine Konstruktion des Baumes mit der anschließenden Bestimmung einer Metrik. So konstruierte BSP Bäume sind im Rahmen der Heuristik *a priori* optimal. Durch die fehlende Rückkopplung und die aufwendige Konstruktion sowie die Vermessung des Baumes sind diese Methoden besonders schnell, führen jedoch zu schlechteren Ergebnissen.

## II. Adaptive Verfahren

Bei diesen Verfahren wird in der Regel eine Vielzahl von verschiedenen Bäumen konstruiert und mit einer Metrik bewertet. Hierbei wird oft versucht, die aus der Konstruktion der Bäume gewonnenen Erkenntnisse für die weitere Optimierung auszunutzen. Ein Abbruchkriterium beendet die Optimierungsschleife nach einem vorher festgelegten Kriterium. Dieses Kriterium kann durch die Konvergenzeigenschaften der Metrik und/oder die Anzahl der gewünschten Iterationen der Optimierungsschleife gegeben sein. Zu dieser Kategorie gehören Verfahren wie genetische Algorithmen und evolutionäre Strategien, Random Walk Prozesse und die Simulated Annealing Methode.

In den folgenden Abschnitten wird zuerst auf die Definition einer geeigneten Metrik und später auf die einzelnen Optimierungsverfahren eingegangen.

### 5.2 Definition der Metrik

Im Bereich der Computergraphik wurde eine Vielzahl von verschiedenen Kostenfunktionen als Metrik für Ray Tracing Hierarchien und räumliche Partitionierungstechniken untersucht (CASSEN et al., 1995; GOLDSMITH und SALMON, 1987; NAYLOR, 1993; SUBRAMANIAN, 1990; SUBRAMANIAN und FUSSELL, 1991). Diese Metriken bestimmen den Ray Tracing Aufwand eines Objekts, indem sie ein Wahrscheinlichkeitsmodell eines Schnittpunktes des Objekts mit einem beliebigen Strahl auswerten. Ein besonders einfaches Beispiel ist in GOLDSMITH und SALMON (1987) zu finden. Dort wird die Trefferwahrscheinlichkeit eines Objekts als Funktion der Oberfläche der objektbezogenen Bounding Box bestimmt.

Die Konstenfunktion eines BSP Baumes  $T$  kann nach NAYLOR (1993) und CASSEN et al. (1995) mit der folgenden Vorschrift berechnet werden:

$$E_{cost}[T] = \begin{cases} 0 & T \text{ ist ein Blatt,} \\ 1 + p^- E_{cost}[T^-] & \text{sonst .} \\ + p^+ E_{cost}[T^+] & \end{cases} \quad (5.1)$$

In dem vorliegenden Fall ist  $T$  der aktuelle Knoten des Baumes.  $T^+$  und  $T^-$  sind der positive und der negative Teilbaum.  $p^+$  und  $p^-$  definieren die Wahrscheinlichkeitsmodelle

für die gewünschte Operation, die mit Hilfe des BSP Baums durchgeführt werden soll. So ist z.B. die Wahrscheinlichkeit, dass sich ein Punkt in dem Teilbaum  $T^+$  befindet, unter der Voraussetzung dass der Punkt im Bereich von  $T$  liegt, als das Verhältnis der Größen beider Teilräume gegeben (CASSEN et al., 1995):

$$\begin{aligned} p^+ &= V(r^+)/Vol(r) + p_0 \\ p^- &= V(r^-)/Vol(r) + p_0 . \end{aligned} \quad (5.2)$$

In der obigen Formel ist mit  $V()$  das Volumen des entsprechenden Teilraumes bezeichnet,  $p_0$  definiert die Wahrscheinlichkeit, dass der Punkt sich in der Partitionierungsebene befindet.

Basierend auf dieser Metrik kann eine allgemeinere Kostenfunktion für beliebige Meta-Hierarchien angegeben werden (FRACH und FISCHER, 2002)

$$E_{cost}[M] = \begin{cases} 0 & M \text{ ist ein Basisobjekt,} \\ 1 + \sum_i p_i E_{cost}[M_i] & \text{sonst .} \end{cases} \quad (5.3)$$

Mit  $M$  ist das übergeordnete aggregierte Meta-Objekt bezeichnet.  $M_i$  sind die in diesem Meta-Objekt enthaltene untergeordnete Objekte.  $p_i$  ist das in (5.2) definierte Wahrscheinlichkeitsmodell.

Andere Kostenfunktionen können auf eine ähnliche Art und Weise definiert werden. Eine besonders einfache Metrik für die Komplexität eines beliebigen räumlichen Partitionierungsverfahrens ist jedoch durch die maximale Höhe des Partitionierungsbaumes gegeben. Balancierte Bäume besitzen die kleinst mögliche Höhe und erlauben somit das Durchqueren des Baumes von der Baumwurzel bis in die Blätter mit der niedrigsten Anzahl der Entscheidungen. Die formale Definition ist mit der folgenden Gleichung gegeben:

$$E_{cost}[T] = \begin{cases} 0 & T \text{ ist ein Blatt,} \\ 1 + \max(E_{cost}[T^j]), \quad j = +, - . & \end{cases} \quad (5.4)$$

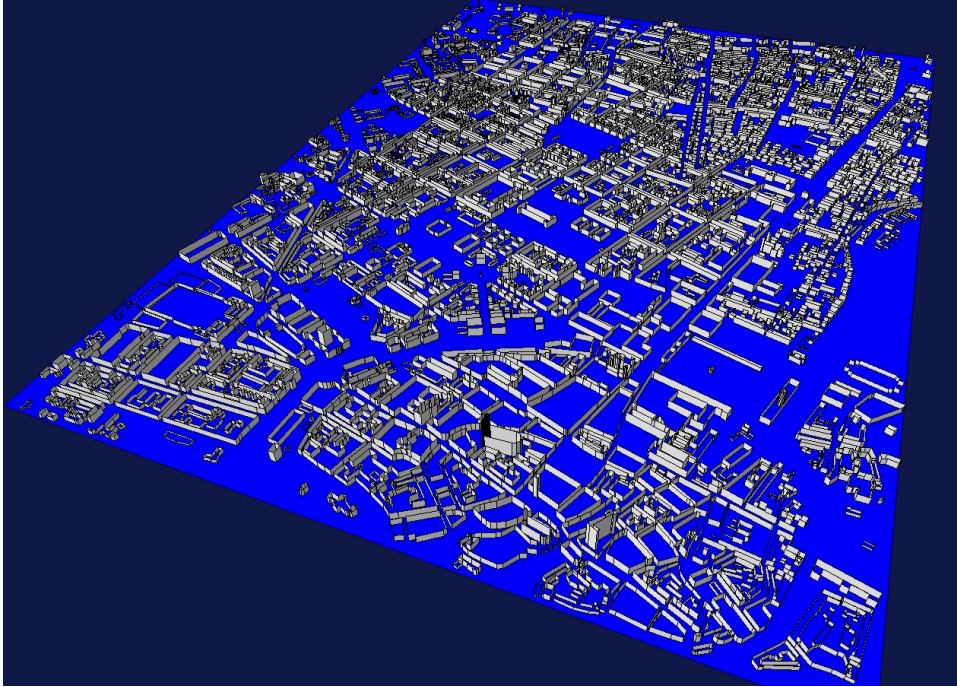
Diese einfache Metrik kann sehr leicht für beliebige Partitionierungs bäume realisiert werden. Sie erfordert lediglich die Messung der maximalen Entfernung eines Blattes von der Baumwurzel. Die Metrik kann bereits während der Konstruktion des Baumes bestimmt werden und stellt daher keinen Mehraufwand dar. Im weiteren Text wird diese Metrik für die Bewertung und die Optimierung räumlicher Partitionierungs bäume angewendet.

### 5.3 Geometrie des Vergleichsszenarios

Um die Effizienz der Optimierungstechniken vergleichen zu können sollte ein einheitliches, der Realität nahes Testszenario gewählt werden. Die in den folgenden Abschnitten vorgestellten Ergebnisse wurden durch die Anwendung der Methoden auf das im europäischen COST<sup>1</sup> 231 Projekt verwendete Modell der Innenstadt von München erzielt (CICHON, 1999). Eine dreidimensionale Ansicht des Modells ist im Bild 5.1 zu sehen. Der Datensatz besteht aus den Koordinaten der 17444 Flächen, die die vertikalen Wände der Gebäude darstellen. Die nur leicht ausgeprägte Topographie wurde im Modell nicht berücksichtigt. Das Szenario, für das auch die Daten von drei verschiedenen Messfahrten vorliegen, beschreibt ein Areal von  $2400 \times 3400 \text{ m}^2$  und wurde im Rahmen des COST 231 Projekts

<sup>1</sup>European Cooperation in the Field of Scientific and Technical Research

für den Vergleich verschiedener Techniken zur Funkfeldvorhersage eingesetzt. Das Szenario kann als ein typisches Modell für eine Geometrie aus dem Bereich des Mobilfunks angesehen werden.



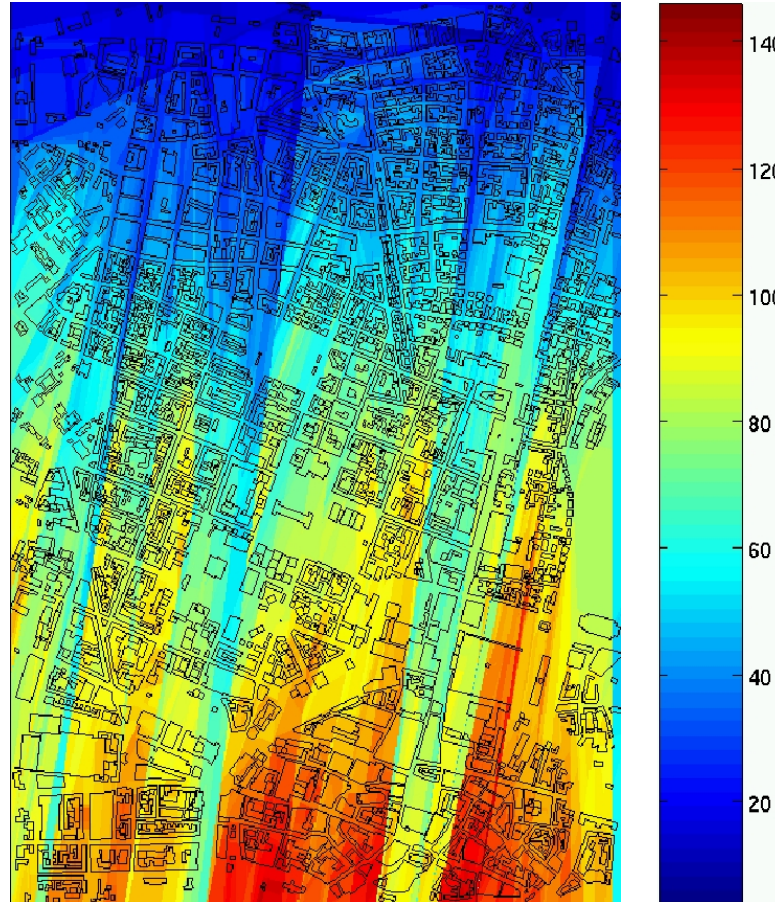
**Bild 5.1:** Das Modell des Vergleichsszenario, welches für die Bewertung der einzelnen Optimierungstechniken eingesetzt wurde.

Die Messung der Metrik wurde für jeden konstruierten Baum durchgeführt. Zusätzlich wurden weitere statistische Parameter wie die minimale und die mittlere Baumhöhe sowie deren Standardabweichung gemessen. Die Metrik kann anschaulich als Funktion der Betrachterposition dargestellt werden, indem die Position eines Punktes in den Partitionierungsbaum einsortiert und gleichzeitig die Höhe des Baumes für diese Position gemessen wird. Wird so die Baumhöhe auf einem Punkteraster abgetastet, so kann die lokale Struktur des BSP Baumes am Ort der Punkte in Form einer Übersichtskarte dargestellt werden. Solche Karten helfen die Problemgebiete zu lokalisieren, in den die Baumhöhe besonders hohe Werte annimmt. Eine solche Karte des nicht optimierten Baumes ist im Bild 5.2 zu sehen. Für die Positionen der Beobachterpunkte wurde ein  $5 \times 5 \text{ m}^2$  Raster gewählt, das sich in 1.6 m Höhe über dem Boden befand. Die Reihenfolge der Polygone, die bei der Konstruktion dieses Baumes benutzt wurde, ist durch die Reihenfolge der Polygone in der Modelldatei vorgegeben. Aufgrund dieser Reihenfolge wurde ein nur unzureichend balancierter Baum konstruiert, mit dessen Hilfe das Ray Tracing insbesondere im unteren Bereich der im Bild dargestellten Geometrie sehr viel mehr Zeit in Anspruch nehmen würde als im oberen Bereich. Das Ziel der Optimierung ist es, diese Unterschiede im gesamten Raum auszugleichen und somit die mittlere Zeit für das Ray Tracing zu minimieren.

#### 5.4 Optimierung mit Heuristiken

Der einfachste Art und Weise, eine geeignete Reihenfolge der Polygone für die Konstruktion des Baumes zu definieren, beruht auf der Definition einer Heuristik. Diese nutzt direkt





**Bild 5.2:** Graphische Darstellung der Baumhöhe des nicht optimierten BSP Baumes des Vergleichsszenario in Abhängigkeit der Beobachterposition. Die Beobachterpunkte wurden auf einem  $5 \times 5 \text{ m}^2$  Gitter gewählt, welches sich in 1.6 m Höhe über dem Boden befand.

messbare Eigenschaften der Objekte wie z.B. ihre Fläche oder ihr Volumen aus, um die Reihenfolge der Objekte für die Konstruktion des BSP Baumes festzulegen. Die Benutzung der Fläche eines Objektes als Maß für eine Heuristik wird dadurch begründet, dass Objekte mit einer großen Fläche auch größere Raumbereiche einnehmen und daher wahrscheinlich von vielen anderen Partitionierungsebenen geschnitten werden. Um eine Fragmentierung dieser großflächigen Polygone zu vermeiden, sollten diese zuerst in den Baum einsortiert werden und die Knoten des Baumes im oberen Bereich der Hierarchie definieren. Diese sog. *Maximum Area* (MA) Heuristik sortiert also die Polygone der Konstruktionsvorschrift entsprechend ihrer abnehmenden Fläche. Abwandlungen dieser Heuristik betrachten bei der Bestimmung der Konstruktionsreihenfolge die in eine vorgegebene Richtung projizierte Fläche der Polygone.

Eine weitere wichtige Eigenschaft der Polygone, die zur Bildung einer Heuristik hinzugezogen werden kann, ist ihre gegenseitige Lage im Raum. Die Anzahl der Polygone, die von der Ebene des betrachteten Polygons geschnitten werden, ist ein Maß für den Grad der, durch dieses Polygon verursachten, Fragmentierung. Durch den Vergleich der Fragmentierung kann eine weitere Heuristik definiert werden. Unter der Annahme, dass Polygone, die viele andere Polygone schneiden und somit zu einer hohen Fragmentie-

ung führen, am Ende der Konstruktionsreihenfolge stehen müssen, wird die *Minimum Intersections* (MI) Heuristik definiert. Diese Heuristik sortiert die Polygone mit einer geringen Fragmentierung an den Anfang der Konstruktionsreihenfolge ein. Voraussetzung dafür ist die Berechnung der Fragmentierung für jedes Polygon. Diese Forderung stellt eine Operation mit der Berechnungskomplexität  $\mathcal{O}(N^2)$  dar und beschränkt den praktischen Einsatz dieser Heuristik auf Modelle mit einer relativ kleinen Anzahl von Polygonen.

Eine Kombination verschiedener Heuristiken ist ebenfalls möglich. Bei der Kombination der MA mit der MI Heuristik werden die Polygone zuerst nach der MI Heuristik sortiert. Bei Polygonen mit einer identischen Anzahl der Schnittpunkte wird die MA Heuristik zur Bestimmung der Reihenfolge eingesetzt. Da bei der Erstellung der optimalen Reihenfolge mittels einer Heuristik keine Kenntnis der Struktur des Baumes vorliegt, können nur a priori gewonnene Aussagen über die erwartete Effizienz des Baumes getroffen werden. Die anschließende Verwendung einer Metrik ermöglicht jedoch einen objektiven Vergleich der resultierenden Bäume (siehe Abschnitt 5.8).

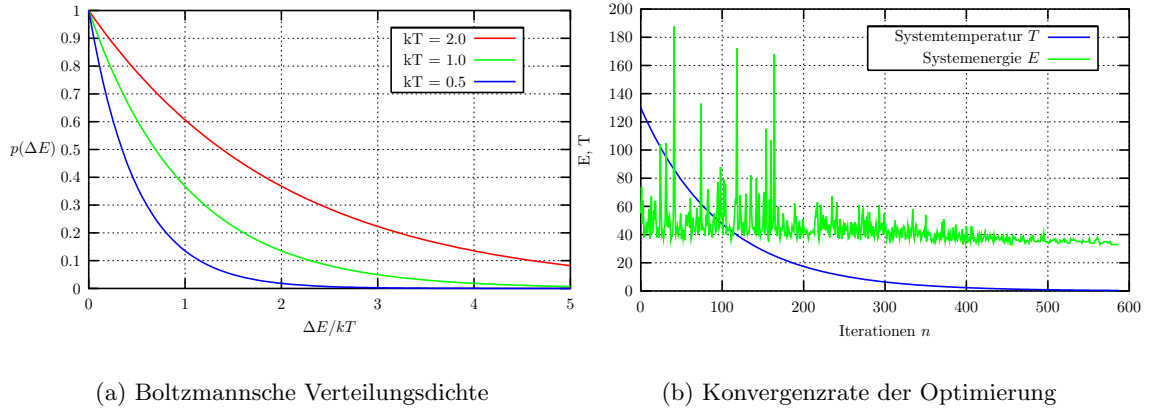
## 5.5 Optimierung mit der Simulated Annealing Methode

Die Simulated Annealing Methode wurde ursprünglich als eine Monte Carlo Methode zur Lösung mehrdimensionaler Zustandsgleichungen in der statistischen Thermodynamik entwickelt (METROPOLIS et al., 1953). Ihr Name leitet sich ab von der Simulation des Aushärtens einer Schmelze. Sie wurde im Laufe der Zeit verallgemeinert und zur Lösung anderer abstrakter Probleme wie z.B. das Problem des Handlungsreisenden in der Informatik eingesetzt. Die Weiterentwicklung der Methode betraf die Verbesserung der Konvergenzgeschwindigkeit (*Very Fast Simulated Re-Annealing*) und ihre Parallelisierung (INGBER, 1989), sowie die Erschließung neuer Einsatzgebiete (KIRKPATRICK et al., 1983; KIRKPATRICK und VECCHI, 1983).

Die klassische Simulated Annealing Methode betrachtet das Verhalten der Gesamtenergie  $E$  einer aus  $N$  Atomen bestehende Schmelze im thermischen Gleichgewicht bei der Temperatur  $T$ . Die Energie eines solchen Systems ist entsprechend der Boltzmannschen Verteilungsfunktion  $p(E)$  zwischen allen energetischen Zuständen  $E$  verteilt. Die Boltzmannsche Verteilungsfunktion ist gegeben durch die folgende Beziehung (PRESS et al., 1988):

$$p(E) \sim \exp\left(-\frac{E}{kT}\right) . \quad (5.5)$$

Diese Funktion besagt, dass sich das System mit einer endlichen (jedoch sehr kleinen) Wahrscheinlichkeit auch bei niedrigen Temperaturen  $T$  in einem energetisch hohen Zustand befinden kann (Bild 5.3a). Dadurch bedingt besteht eine endliche Wahrscheinlichkeit, dass das System das lokale Energieminimum zugunsten eines besseren globalen Minimums verlassen kann, indem es die dazwischenliegende Energiebarriere überwindet. Die Wahrscheinlichkeit, dass dies der Fall ist, wird über die Systemtemperatur  $T$  gesteuert. Indem die Temperatur des Systems mit der Zeit abnimmt, verringert sich die Wahrscheinlichkeit solcher Zustandsänderungen und das System wird in der letzten Konfiguration eingefroren. Die in der Schmelze frei beweglichen Atome werden während des Abkühlens auf bestimmte Positionen im Festkörper gebunden. Abhängig von dem tatsächlichen Verlauf des Abkühlprozesses  $T(t)$  entsprechen diese Positionen einem mehr oder weniger günstigen energetischen Zustand. Durch ein sehr langsames Abkühlen der Schmelze können sich Atome im Gitter besser umgruppieren und einen optimalen Energiezustand erreichen. Das so entstandene Gitter eines Einkristalls besitzt daher die minimale Gesamtenergie. Andererseits wird durch ein schnelles Abschrecken der Schmelze ein amorpher



(a) Boltzmannsche Verteilungsdichte

(b) Konvergenzrate der Optimierung

**Bild 5.3:** Die Boltzmannsche Wahrscheinlichkeitsverteilungsdichte in Abhängigkeit der bezogenen Energiedifferenz sowie der Verlauf der Systemenergie und der Temperatur während einer Optimierung mit der Simulated Annealing Methode.

Zustand mit einer hohen Gesamtenergie des Festkörpers erreicht, in dem die aktuelle Position der Atome zum Zeitpunkt des Abschreckens eingefroren wird. Zustände zwischen diesen beiden Extrema äußern sich in einem polykristallinen Aufbau des Festkörpers.

Die Simulated Annealing Methode bildet den Prozess der Umgruppierung der Atome im  $N$ -dimensionalen Zustandsraum simulativ nach. Zu dem diskreten Zeitpunkt  $n$  befindet sich das System im thermischen Gleichgewichtszustand bei der Temperatur  $T_n$  und der Gesamtenergie  $E_n$ . Zum Zeitpunkt  $n + 1$  geht das System mit der Übergangswahrscheinlichkeit  $p(\Delta E)$  in einen neuen Energiezustand  $E_{n+1}$  über. Die Energiedifferenz beider Zustände  $\Delta E = E_{n+1} - E_n$  wirkt sich also direkt auf die Wahrscheinlichkeit einer Zustandsänderung aus. Die Funktion  $p(\Delta E)$  ist gegeben durch die Form der Boltzmann-Verteilung:

$$p(\Delta E) = \begin{cases} 1 & \text{für } \Delta E < 0 \\ e^{-\frac{\Delta E}{kT}} & \text{sonst} \end{cases} \quad (5.6)$$

Für negative Energiedifferenzen  $\Delta E$  wird die Wahrscheinlichkeit einer Zustandsänderung gleich eins. In diesem Fall folgt also das System immer einer Zustandsänderung von dem aktuellen Zustand  $E_n$  in einen energetisch günstigeren Zustand  $E_{n+1}$ . Die entscheidende Bedeutung in der obigen Formel kommt dem zeitlichen Verlauf der Temperatur des Systems  $T_n$  zu. Bei einer zu schnellen Abkühlung hat das System nicht ausreichend Zeit, neue günstigere Zustände auszuprobieren und verbleibt in einem lokalen Minimum. Zu langsames Abkühlen führt andererseits zu langen Rechenzeiten. Der optimale Verlauf der Temperatur, der sog. *Annealing Schedule*, ist für Systeme mit Boltzmann-ähnlicher Verteilungsdichte der Zustände im Zustandsraum durch

$$T_n = \frac{T_0}{\ln n} \quad (5.7)$$

gegeben (INGBER, 1989). GEMAN und GEMAN (1984) konnten beweisen, dass für diesen Verlauf der Temperatur das System das globale Minimum der Gesamtenergie erreicht. Dieser Verlauf der Temperatur führt jedoch nur zu einer sehr langsamen Konvergenz des Verfahrens. Für die Wahl einer anderen Verteilungsdichte bei der Abtastung des Zustandsraumes ergeben sich unter Umständen auch schnellere Temperaturverläufe und Konvergenzraten

(*Fast Cauchy Annealing*). Die Einführung eines Aufheiz-/Abkühlzyklus im Temperaturverlauf kann das Verfahren weiter beschleunigen (INGBER, 1989).

Der Einsatz der Simulated Annealing Methode bei der Optimierung binärer Partitionierungsbäume setzt die eindeutige Definition eines Zustandsraumes voraus. Dieser Zustandsraum ist durch die Menge  $D$  aller  $N!$  Permutationen der  $N$  Polygone festgelegt. Somit muss jede Änderung des Zustands  $X_n \in D$  wiederum eine gültige Permutation  $X_{n+1} \in D$  zur Folge haben. Dies wird durch die Paarweise Vertauschung von Polygonen des Zustands  $X_n$  gewährleistet. Die Wahl der zu vertauschenden Polygone wird mit Hilfe einer Normalverteilung getroffen. Desweiteren wird eine eindeutige Definition der Systemenergie  $E_n$  benötigt. Diese ist durch die eingesetzte Metrik gegeben.

### Metropolis Algorithmus — Optimierungsschleife

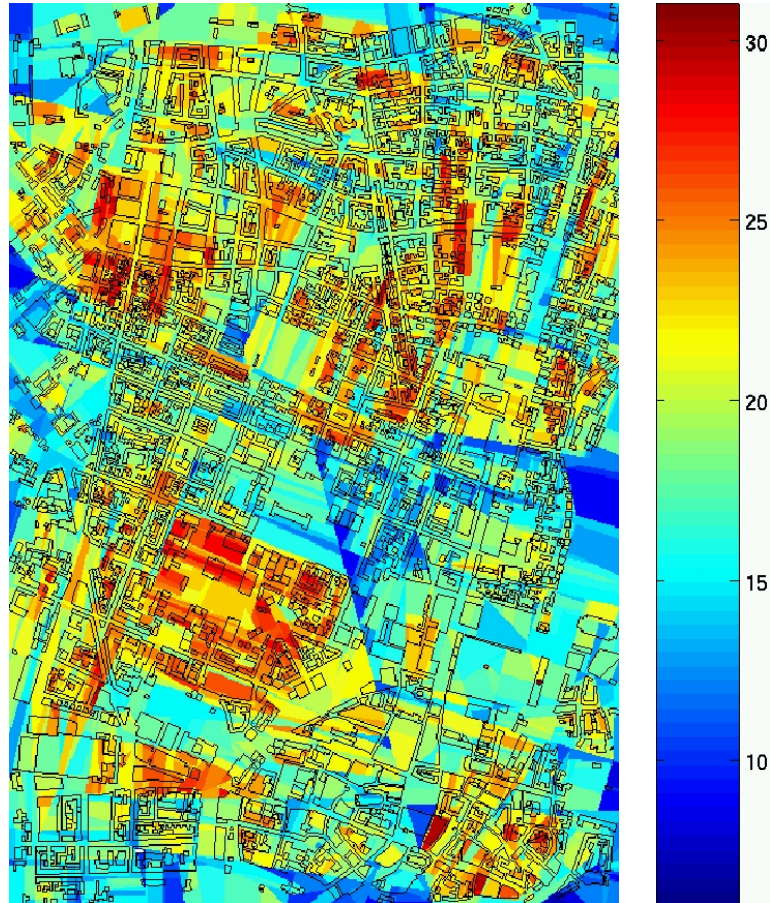
Zufällige Wahl eines Anfangszustands $X_0$ aus dem Zustandsraum $D$ . Bestimmung der zugehörigen Systemenergie $E_0$ durch die Auswertung der Metrik $M(X_0)$ . Wahl der Anfangstemperatur $T_0$ .	
Bestimmung eines neuen Zustands $X$ durch zufällige Modifikation des Zustands $X_n$ . Die Modifikation erfolgt entsprechend einer vorgegebenen Wahrscheinlichkeitsverteilungsfunktion durch zyklische Vertauschung mehrerer Polygone. Bestimmung der zugehörigen Systemenergie $E$ und der Energiedifferenz $\Delta E = E - E_n$ .	
Metropolis Kriterium $\text{rand}(0,1) < p(\Delta E)$	
J	N
Die Zustandsänderung wird akzeptiert. Der Zustand $X$ wird der neue Zustand $X_n$ und die Energie $E$ wird zu $E_n$ .	Der neue Zustand wird nicht akzeptiert und geht verloren.
Die Ausgleichsschleife läuft bis das thermische Gleichgewicht erreicht ist. Das thermische Gleichgewicht wird erreicht, wenn die Energiedifferenz $\Delta E$ $k$ -mal in Folge unter eine definierte Schranke $S$ fällt.	
Änderung der Systemtemperatur $T_n$ zu $T_{n+1}$ gemäß des vorgegebenen <i>Annealing Schedule</i> . Der Gleichgewichtszustand $X_n$ sowie die zugehörige Systemenergie $E_n$ werden zu dem neuen Systemzustand $X_{n+1}$ und Systemenergie $E_{n+1}$ .	
Die äussere Optimierungsschleife wird abgebrochen, wenn die Systemtemperatur eine vorgegebene Schranke $T < T_{min}$ unterschreitet.	
Der Zustand $X_n$ stellt jetzt den optimierten Zustand dar.	

**Bild 5.4:** Die Anwendung der von Metropolis et al. (1953) vorgestellten Simulated Annealing Methode auf die Optimierung der Konstruktion binärer Partitionierungsbäume.

Die Anwendung des von METROPOLIS et al. (1953) vorgestellten Simulated Annealing Algorithmus auf die Optimierung eines binären Partitionierungsbaumes ist im Diagramm 5.4 zu sehen. Zunächst wird ein zufälliger Systemzustand  $X_0 \in D$  erzeugt und seine Energie mit Hilfe der Metrik bestimmt. Um die Metrik auswerten zu können, muss der zu der Permutation  $X_0$  zugehörige Baum aufgebaut und vermessen werden. Die innere Schleife des Algorithmus hat die Aufgabe, das System bei konstanter Temperatur  $T_n$  in ein thermisches Gleichgewicht zu bringen. Hierzu wird der Systemzustand  $X_n$  durch die zyklische Vertauschung von Polygonen verändert und seine Metrik bestimmt. Die Energiedifferenz zu dem ursprünglichen Systemzustand bestimmt zusammen mit der aktuellen Systemtemperatur  $T_n$  die Akzeptanzwahrscheinlichkeit des neuen Zustands. Nach dem Metropolis Kriterium wird der neue Zustand immer akzeptiert, wenn der neue Zustand eine niedrigere Energie besitzt. Andererseits wird ein Zustand mit einer höheren



Energie nach der in der Gleichung (5.6) definierten Wahrscheinlichkeit angenommen. Das thermische Gleichgewicht wird als erreicht angenommen, wenn eine vorgegebene Anzahl sukzessiver Abtastungen des Zustandsraumes zu keiner weiteren Veränderung des Systemzustands führen.



**Bild 5.5:** Eine Übersichtskarte der Baumhöhe des optimierten BSP Baumes in Abhängigkeit der Beobachterposition. Im Vergleich zu der im Bild 5.2 dargestellten Karte konnte durch die Optimierung die maximale Baumhöhe auf 33 abgesenkt werden.

Die äußere Schleife des Algorithmus stellt die Systemtemperatur entsprechend dem *Annealing Schedule* ein und kühlt somit das System ab. Sie wird beendet, wenn die Systemtemperatur unter einen benutzerdefinierten Wert fällt. Die Implementierung des *Annealing Schedule* bestimmt die Konvergenzrate und die Rechengeschwindigkeit der Optimierung, wobei die Auswertung der Metrik in der inneren Ausgleichsschleife den Aufbau des Baumes erfordert und somit die laufzeitbestimmende Größe darstellt. Der Verlauf der Systemenergie und der Temperatur während einer Optimierung ist im Bild 5.3b dargestellt. Das Ergebnis der Optimierung des BSP Baumes für das Vergleichsszenario ist im Bild 5.5 zu sehen. Die maximale Baumhöhe konnte durch die Wahl der optimierten Reihenfolge der Polygone von 146 auf 33 abgesenkt werden. Gleichzeitig konnte die mittlere Baumhöhe von ursprünglich 75 auf 19 abgesenkt werden. Durch die Verkleinerung der mittleren Baumhöhe um den Faktor 3.5 ist eine Beschleunigung der Ray Tracing Simu-

lation um etwa den gleichen Faktor zu erwarten. Eine solche Beschleunigung konnte mit durchgeführten Simulationsmessungen tatsächlich nachgewiesen werden.

## 5.6 Genetische Algorithmen

Genetische Algorithmen (DAVIS, 1987; GOLDBERG, 1989) haben sich ebenfalls bei der Lösung komplexer Optimierungsprobleme bewährt. Ein genetischer Algorithmus bildet das Optimierungsproblem auf die evolutionäre Anpassung einer Population an ihre Umgebung ab. Die Population wird dabei durch ihren genetischen Pool repräsentiert, dessen Chromosome mit Hilfe einer Fitness-Funktion (Metrik) bezüglich ihrer Eignung bewertet werden. Durch Mutation, Kreuzung und Selektion passt sich der genetische Pool von einer Generation zur nächsten an die Rahmenbedingungen an. Genetische Algorithmen bilden somit den von Darwin beschriebenen Optimierungsprozess der Natur simulativ nach. Der prinzipielle Ablauf der genetischen Optimierung ist im Diagramm 5.6 dargestellt.

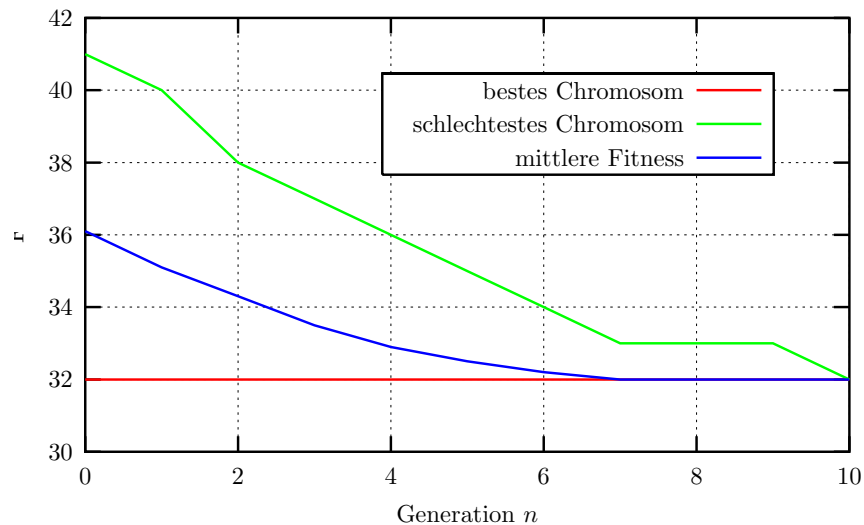
### Genetischer Algorithmus — Optimierungsschleife

Zufällige Wahl einer Anfangspopulation $P_0$ mit $M$ Chromosomen $X_k$ aus dem Zustandsraum $D$ . Bestimmung der zugehörigen Fitness-Funktion $F_k$ für jedes Chromosom durch die Auswertung der Metrik $M(X_k)$ .	
Sortieren der Chromosome der Population nach ihrer Fitness Funktion.	
<b>Selektion</b> der $K$ besten Chromosome für den Aufbau der Population $P_{n+1}$ .	
Aufbau der Population $P_{n+1}$ mit $M$ neuen Individuen durch die zufällige <b>Kreuzung</b> der $K$ besten Chromosome der vorhergehenden Population $P_n$ . Die Kreuzungswahrscheinlichkeit definiert die Menge des Erbmateriale, die zwischen beiden Eltern ausgetauscht wird, um die Chromosome der zwei Nachkommen zu erzeugen.	
Einfügen einer zufälligen Störung in den genetischen Pool durch die <b>Mutation</b> . Die Mutation verändert nach der vorgegebenen Mutationswahrscheinlichkeit einige Chromosome, indem sie zufällig gewählte Gene innerhalb der Chromosome vertauscht. Es werden keine weiteren Gene zwischen den Chromosomen ausgetauscht.	
Bewertung der neuen Population $P_{n+1}$ mit der Fitness Funktion. Um die Fitness Funktion auswerten zu können, muss der zu jedem Chromosom zugehörige Baum konstruiert und vermessen werden. Dies stellt die zeitaufwendigste Operation des Verfahrens dar.	
Sortieren der Chromosome der Population nach ihrer Fitness Funktion.	
Die Optimierungsschleife wird nach der vorgegebenen Anzahl Generationen abgebrochen. Zusätzliche Abbruchkriterien, die die Konvergenzeigenschaften der Fitness der Population betrachten sind ebenfalls möglich.	
Das beste Chromosom der letzten Population stellt den optimierten Zustand dar.	

**Bild 5.6:** Optimierung mit einem genetischen Algorithmus. Der Algorithmus erzeugt eine zufällige Population und bewertet diese mit der Fitness Funktion. Anschließend wird über die Selektion, Kreuzung und Mutation eine neue, besser angepasste Population erzeugt und bewertet. Die Optimierungsschleife wird nach einer vorgegebenen Anzahl Durchläufe abgebrochen.

Um einen genetischen Algorithmus einsetzen zu können, muss das Problem in der Form eines diskreten Zustandsraumes formuliert werden. Ein Zustand des diskreten Zustandsraumes wird durch eine Gensequenz, das Chromosom, beschrieben. Desweiteren muss eine eindeutige Fitness-Funktion existieren, die die Anpassung des durch das Chromosom beschriebenen Individuums an die statischen oder dynamischen Randbedingungen

beschreibt. Durch eine Selektion einer geringen Anzahl der am Besten angepassten Chromosome und der Bildung einer neuen Population durch die Kreuzung dieser Chromosome untereinander entsteht eine neue Generation. Dieser Prozess wird als das Generationsmodell eines genetischen Algorithmus bezeichnet. Die Mutation wirkt sich als eine zufällige Störung des genetischen Materials eines Chromosomes aus, indem es zwei beliebige Gene innerhalb eines und des selben Chromosomes vertauscht. Die Wahrscheinlichkeiten, mit denen eine Kreuzung oder eine Mutation innerhalb der Population auftritt, beschreiben die Fähigkeit zur Veränderung des genetischen Materials. Mit ihnen kann die Konvergenzrate und die Fähigkeit der Suche nach alternativen Lösungen gesteuert werden.

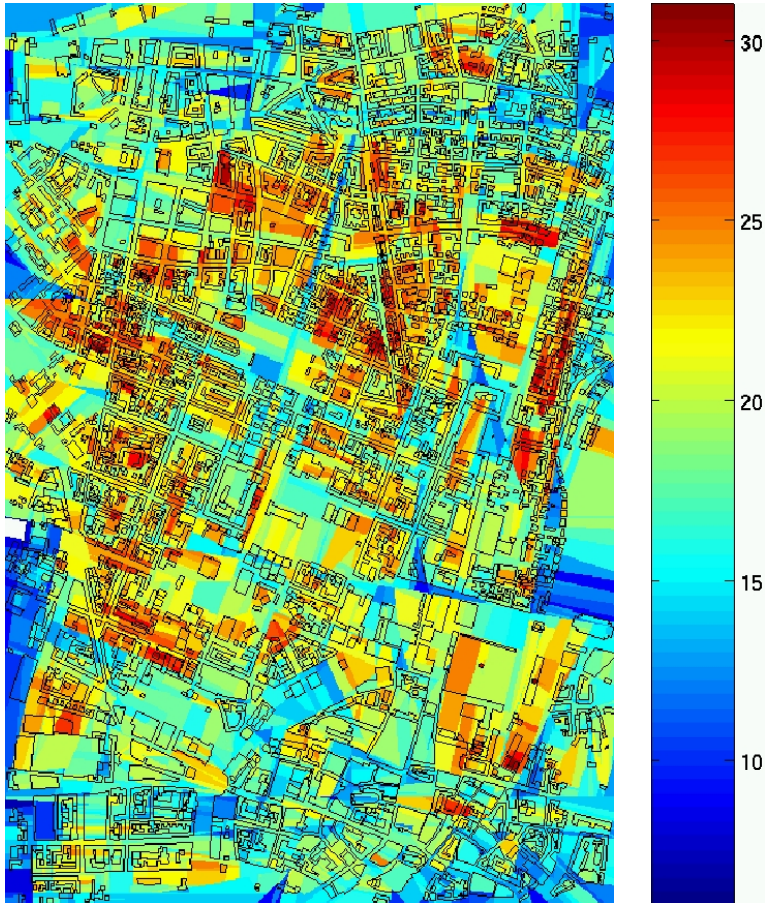


**Bild 5.7:** Konvergenzrate des genetischen Algorithmus. Dargestellt sind die Entwicklungen des besten und des schlechtesten Chromosomes sowie die mittlere Fitness des genetischen Pools.

Die Anwendung eines genetischen Algorithmus zur Optimierung binärer Partitionierungsbäume gestaltet sich besonders einfach (CASSEN et al., 1995). Die Reihenfolge der Polygone, die durch ihre Indizes eindeutig beschrieben sind, kann direkt als das Chromosom betrachtet werden. Die Polygone stellen hierbei die Gene dar. Die Metrik entspricht der Fitness-Funktion, die durch die Optimierung zu minimieren ist. Die Mutation entspricht der Vertauschung zweier zufällig gewählter Gene innerhalb eines Chromosomes und erzeugt somit immer einen gültigen Zustand aus dem Zustandsraum  $D$ . Die Kreuzung betrifft immer zwei Chromosome, die ihr genetisches Material austauschen und dadurch die Chromosome zweier Nachkommen der nächsten Generation erzeugen. Die Kreuzung kann nicht in klassischer Weise durch den Austausch beliebiger Gene realisiert werden, da dadurch Chromosome entstehen würden, die keine gültige Permutation der  $N$  Polygone darstellen und somit nicht im Zustandsraum  $D$  enthalten sind. Stattdessen müssen Gene immer paarweise ausgetauscht werden, so dass die Permutationsbedingung erhalten bleibt.

Das Konvergenzverhalten des genetischen Algorithmus ist im Bild 5.7 zu sehen. Aus der Konvergenz der Fitness Funktion des besten und des schlechtesten Chromosomes ergibt sich insgesamt die Optimierung der Population im Verlauf der zehn Generationen. Jede Population bestand aus 100 Chromosomen, aus den jeweils die 10 besten Chromosome ausgewählt wurden, um mit einer Kreuzungswahrscheinlichkeit von 0.6 und einer Mutationsrate von 0.1 die neue Population zu erzeugen.

Der Verlauf der Fitness-Funktion des optimierten Baumes ist im Bild 5.8 zu sehen. Die maximale Höhe des Baumes beträgt 32 und ist damit vergleichbar mit dem Ergebnis der Optimierung durch die Simulated Annealing Methode.



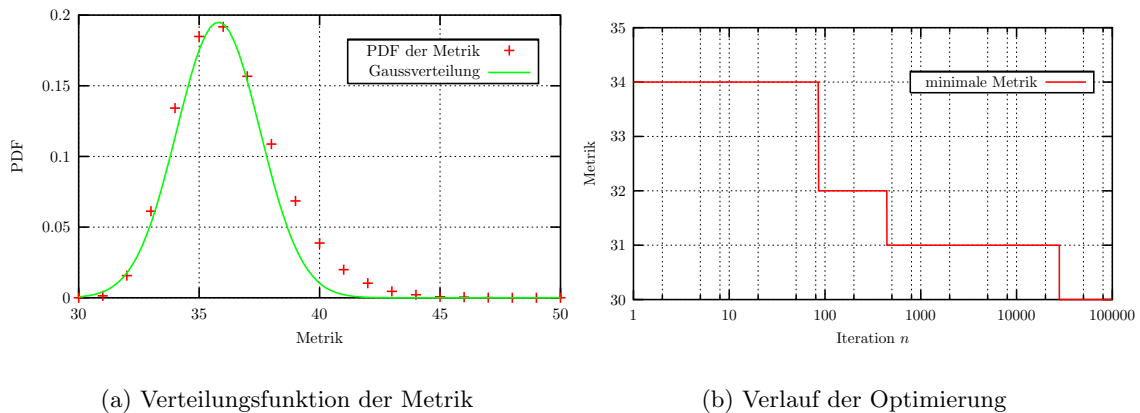
**Bild 5.8:** Optimierung des BSP Baumes mit dem genetischen Algorithmus. Die Höhe des Baumes ist farblich kodiert in Abhängigkeit der Beobachterposition aufgetragen. Verglichen mit dem durch Simulated Annealing optimierten BSP Baum (Bild 5.5) wird eine etwas gleichmäßigere Verteilung der Baumhöhe erreicht.

## 5.7 Optimierung durch zufällige Abtastung

Das Konvergenzverhalten der Metrik bei der Optimierung mit der Simulated Annealing Methode deutet darauf hin, dass die Metrik viele lokale Minima aufweist, die bereits sehr nahe am Optimierungsergebnis liegen. Der Verlauf deutet ebenfalls darauf hin, dass die Metrik keine konvexe Funktion im Zustandsraum bildet und die Simulated Annealing Methode das Ergebnis durch eine zufällige Abtastung des Zustandsraumes erreicht. Der Grund liegt darin, dass sich das Vertauschen von zwei Polygonen in der Konstruktionsreihenfolge auf den gesamten Baum auswirken kann. Somit ist eine lokale Veränderung der Eingangsgröße mit einer globalen Änderung der Metrik verbunden. Die Wahrscheinlichkeitsverteilung der Metrik wurde für 100000 zufällig erzeugte Zustände berechnet und ist im Bild 5.9a zu sehen.



Die Verteilungsfunktion der Metrik lässt sich in diesem Fall gut mit einer Gaussverteilung näherungsweise beschreiben. Diese Eigenschaft der Metrik wurde auch bei allen anderen geometrischen Modellen, die im Kapitel 8 behandelt werden, beobachtet (siehe auch Kapitel 8, Seite 126). Dadurch wird klar, dass bereits durch eine zufällige Wahl eines Zustandes mit einer hohen Wahrscheinlichkeit ein Baum mit einer guten Metrik gefunden werden kann. Dieser Verlauf der Verteilungsfunktion bildet die Grundlage für ein sehr einfaches Verfahren zur Optimierung des BSP Baumes. Das Verfahren wählt zufällige Reihenfolgen der Polygone und konstruiert den BSP Baum. Die Metrik des Baumes wird bestimmt und mit der bis dahin gefundenen minimalen Metrik verglichen. Ist die neue Metrik kleiner, so ersetzt diese das alte Minimum. Die zugehörige Reihenfolge der Polygone wird ebenfalls für eine spätere Rekonstruktion des Baumes abgelegt. Das Verfahren wird iterativ für eine voreingestellte Anzahl von Schritten durchgeführt. Diese Anzahl der Iterationen bestimmt, mit welcher Wahrscheinlichkeit ein minimaler Zustand gefunden werden kann. Der Verlauf der Optimierung ist im Bild 5.9b dargestellt. Unter Umständen ist eine große Anzahl von Iterationen nötig, um ein Minimum der Metrik zu finden.



**Bild 5.9:** Die Wahrscheinlichkeitsverteilungsfunktion der Metrik für das betrachtete Szenario. Der Verlauf deutet darauf hin, dass bereits durch eine zufällige Abtastung des Zustandsraumes ein näherungsweise optimaler Zustand gefunden werden kann.

## 5.8 Vergleich der Methoden

In der Tabelle 5.1 sind die Metriken der verschiedenen Optimierungsansätze für das im Bild 5.1 vorgestellte Szenario zusammengefasst. Aus der Tabelle ist ersichtlich, dass die Heuristiken kaum mit den adaptive Methoden vergleichbare Ergebnisse liefern. Durch die fehlende Rückkopplung der Heuristiken kann eine Anpassung der Konstruktionsreihenfolge des Baumes an die zugrundeliegende Geometrie nicht erreicht werden. Eine Ausnahme bildet die als 'zufällige Reihenfolge' bezeichnete Heuristik. In diesem Fall ist es möglich, wie bereits im letzten Abschnitt erläutert, durch eine zufällige Wahl einer Permutation eine sehr gute Metrik des BSP Baumes zu erreichen. Allerdings variiert das Ergebnis entsprechend der Verteilungsfunktion aus Bild 5.9a. Das beste Ergebnis erzielt die Optimierung durch eine zufällige Abtastung des Zustandsraumes, das Ergebnis wird jedoch erst nach der Konstruktion von mehr als 20000 verschiedenen BSP Bäumen erreicht.

Optimierung	Höhe des Baumes			
	Minimal	Maximal	Mittelwert	Std.-Abw.
keine Optimierung	1	146	68.27	28.02
MI Heuristik	1	115	79.66	17.05
MI/MA Heuristik	1	112	76.48	16.05
MA Heuristik	6	40	23.40	5.55
Zufällige Reihenfolge	7	35	20.08	3.81
Simulated Annealing Methode	5	31	20.54	4.39
Genetischer Algorithmus	7	30	18.80	3.55
Zufällige Abtastung	7	30	19.01	3.35

**Tabelle 5.1:** Vergleich der verschiedenen Optimierungstechniken, angewendet an das im Bild 5.1 dargestellte Szenario.

## 5.9 Weitere Optimierungsmöglichkeiten

Weitere Optimierungsmöglichkeiten betreffen sowohl die Schnittpunktberechnung mit Hilfe binärer Partitionierungsbäume als auch die Optimierungsstrategien der Bäume selbst. Eine Beschleunigung der Schnittpunktberechnung kann durch den Einsatz sogenannter *Rope Trees* um einige Prozent beschleunigt werden (HAVRAN et al., 1998). Ähnlich wie bei den topologischen BSP Bäumen (Kapitel 4 Seite 56) werden durch den *Rope Tree* die Nachbarkheitseigenschaften im BSP Baum in der Form einer zusätzlichen baumartigen Struktur festgehalten. Bei der Berechnung der Schnittpunkte eines reflektierten Teilstrahls kann die Position der Reflexionsstelle im BSP Baum ausgenutzt werden, um den betreffenden Teilbaum mit Hilfe des *Rope Tree* schneller zu finden. Dadurch wird eine Traversierung des ganzen Baumes von der Wurzel aus vermieden. Untersuchungen zur Effizienz dieses Verfahrens zeigen jedoch, dass nur ein geringfügiger Geschwindigkeitsgewinn im Bereich einiger Prozent bei einem verachtfachen Speicherbedarf zu erzielen ist (KŘIVÁNEK und BUBNIK, 2000).

Eine Verbesserung im Bereich der Optimierungsstrategien kann auf mehreren Wegen erreicht werden. Einerseits können die bestehenden Optimierungsalgorithmen weiter verbessert und neue Verfahren implementiert werden. Andererseits kann der Einfluss der Parameter auf das Ergebnis und die Konvergenzrate der Optimierung untersucht und der Parametersatz eingegrenzt werden.

Eine weitere Möglichkeit, das Ergebnis der Optimierung zu verbessern, liegt im gezielten Einfügen zusätzlicher virtueller Partitionierungsebenen in den BSP Baum. Als eine virtuelle Partitionierungsebene wird hierbei eine Ebene verstanden, die keine Polygone enthält. Somit kann durch den Einsatz virtueller Ebenen die Struktur des Baumes beeinflusst werden, ohne die Geometrie des Szenarios zu verändern<sup>2</sup>. Diese Möglichkeit wurde bei der Behandlung rotationssymmetrischer Körper untersucht. Aufgrund der Rotationssymmetrie entartet der gewöhnliche BSP Baum zu einer linearen Liste, da die Partitionierungsebenen zuerst nur durch die in der Geometrie befindlichen Polygone definiert sind. Diese Polygone sind durch den Algorithmus zur Konstruktion des Rotationskörpers alle in den selben Halbraum orientiert. Durch das Einfügen virtueller Ebenen, die durch die Eckpunkte des Rotationskörpers und die Rotationsachse definiert sind, wird die Anzahl der Freiheitsgrade für die Optimierung erhöht. Eine Reduktion der Baumhöhe um zwei Größenordnungen und mehr konnte auf diese Weise erreicht werden. Das automatische Einfügen von virtuellen Ebenen ist jedoch nur unter bestimmten Bedingungen möglich. Im

<sup>2</sup>Dies gilt nur unter der Annahme einer exakten Arithmetik.

obengenannten Fall wird dazu die Kenntniss der Rotationssymmetrie ausgenutzt. Bei unregelmäßigen Szenarien gestaltet sich eine automatische Festlegung der Ebenen schwierig. Eine manuelle Festlegung wäre zwar möglich, ist jedoch mit einem erheblichen Aufwand für den Benutzer verbunden und wurde daher im Rahmen dieser Arbeit nicht betrachtet.

Zuletzt soll auf die Bedeutung der Metrik bei der Optimierung eingegangen werden. Die Wahl der Metrik ist sehr von der Anwendung abhängig und kann entscheidend die Performance des Ray Tracing Verfahrens beeinflussen. Aus der Computergraphik sind andere Metriken bekannt, die sich besonders für Sichtbarkeitsprobleme und Rendering eignen (SUBRAMANIAN und FUSSELL, 1990). Die Wahl einer anderen als der hier eingesetzten Metrik könnte unter Umständen zu einer weiteren Steigerung der Effizienz führen. Ebenso können zusätzliche Bedingungen bei die Berechnung der Metrik berücksichtigt werden. So kann die Metrik für bestimmte Anwendungen die Position des Senders mitberücksichtigen. Durch eine Optimierung mit einer solchen Metrik wird die Baumhöhe in der Nähe des Senders, dort wo die Strahlendichte am höchsten ist, besonders klein sein. Die Baumhöhe würde dann mit steigender Entfernung zum Sender ansteigen. Der so optimierte Baum könnte für das Ray Tracing in der unmittelbaren Umgebung des Senders besonders schnell sein. In weiterer Entfernung wäre der Gewinn durch die größere Baumhöhe wieder kompensiert. Durch die Nebenbedingung wird die Anzahl der Freiheitsgrade des Optimierers eingeschränkt und es ist zu erwarten, dass der so konstruierte Baum im Mittel schlechter ist als ein Baum, der ohne Nebenbedingungen optimiert wurde. Desweiteren wäre so ein Baum nur unter der Einhaltung der gewählten Nebenbedingung effizient einsetzbar.

## 5.10 Zusammenfassung

Die Konstruktion des BSP Baumes erfordert die Festlegung der Reihenfolge, in der die Polygone in den Baum eingefügt werden. Durch die Wahl einer geeigneten Metrik kann die Güte des BSP Baumes bezüglich der Effizienz der Schnittpunktsuche bewertet werden. Die Definition einer Metrik ermöglicht die Bestimmung einer näherungsweise optimalen Reihenfolge der Polygone mit Hilfe nichtlinearer Optimierungsverfahren wie der Simulated Annealing Methode und den genetischen Algorithmen. Aufgrund der gaussförmigen Wahrscheinlichkeitsverteilung der Metrik kann ebenfalls eine einfache zufallsbasierte Abstufung des Zustandsraumes zur Konstruktion von näherungsweise optimalen BSP Bäumen eingesetzt werden. Die Voraussetzung dafür ist eine hinreichend hohe Zahl der Optimierungsschritte.

Durch die Wahl zusätzlicher virtueller Partitionierungsebenen kann in bestimmten Fällen die Effizienz des optimierten Baumes deutlich verbessert werden. In gleicher Weise konnte im letzten Kapitel durch das Einfügen zusätzlicher Ebenen die Empfängerdetektion beschleunigt werden.



---

## Adaptives Ray Tracing

Aufbauend auf der in den letzten Kapiteln behandelten schnellen Schnittpunktsuche mit optimierten BSP Bäumen wird in diesem Kapitel ein adaptives Ray Tracing Verfahren beschrieben. Das auf der *Cone Tracing* Methode basierte Verfahren passt die Strahlendichte lokal der zugrundeliegenden Geometrie an, um die Gesamtzahl der benötigten Strahlen minimal zu halten. Das Verfahren kann mit der *Spiegelungsmethode* kombiniert werden, um die Geschwindigkeit des Cone Tracing Verfahrens mit der Genauigkeit der Spiegelungsmethode zu erhalten. Eine neuartige Strahlendichtenormierung mit Hilfe markierter Strahlen wird für die schnelle Identifikation neuer Ausbreitungspfade eingesetzt. Die Eigenschaften der zufallsbasierten Strahlengenerierung werden mit zwei deterministischen Ansätzen verglichen und insbesondere bezüglich ihrer Speicherzugriffseffizienz bewertet. Die Realisierung einer Anpassung der Strahlendichte an die Geometrie wird vorgeschlagen und mit dem klassischen Fall einer konstanten Strahlendichte verglichen. Abschließend wird auf die Implementierung der Beugungskantendetektion und die Spiegelung der Beugungsquellen eingegangen sowie das eingesetzte Modell der Streuquellen beschrieben.

### 6.1 Das Cone Tracing Verfahren

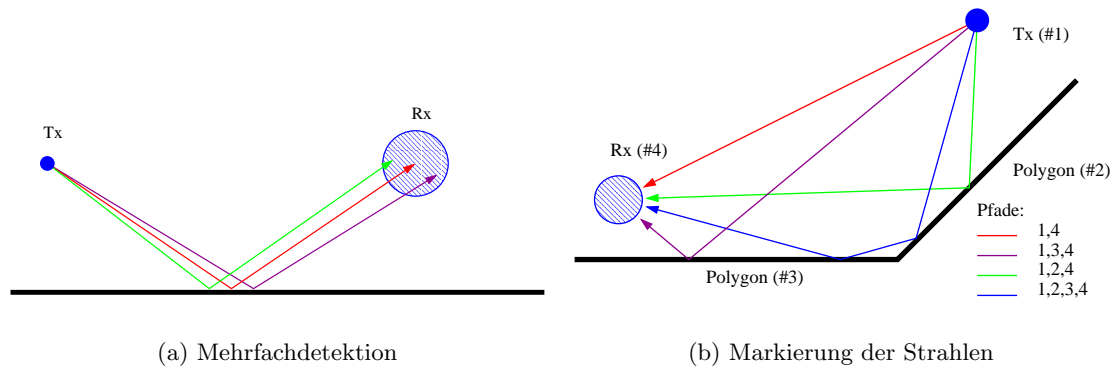
Die prinzipiellen Grundlagen des Cone Tracing Verfahrens wurden bereits im Kapitel 3 behandelt. Das Cone Tracing Verfahren greift auf die Methode der schnellen Schnittpunktsuche zurück, die im Kapitel 4 behandelt wurde. Der iterative Algorithmus berechnet die Ausbreitung eines Strahls durch das Szenario unter der Berücksichtigung einer vorgegebenen Anzahl von Reflexionen und Transmissionen. Durch das eingesetzte Wandmodell bedingt, treten transmittierte Strahlen unter dem gleichen Winkel aus der Wand heraus, unter dem sie in die Wand eingetreten sind. Der transmittierte Teilstrahl erfährt also keine Richtungsänderung und breitet sich stets geradlinig aus. Diese Eigenschaft kann vorteilhaft zur Beschleunigung des gesamten Verfahrens ausgenutzt werden, indem alle Transmissionen eines Teilstrahls vor den Reflexionen berechnet werden. Um die Transmissionen berechnen zu können, müssen alle von diesem Teilstrahl getroffenen Polygone durch eine Schnittpunktberechnung des Teilstrahls mit der Geometrie ermittelt werden. Dies wird durch eine einzige Abfrage des BSP Baumes erreicht<sup>1</sup>, bei der die entsprechende Anzahl der getroffenen Polygone gespeichert und an die innere Ray Tracing Schleife übergeben wird. Somit können alle Transmissionen eines Teilstrahls mit nur einer einzigen Schnittpunktberechnung bestimmt werden. Die Zahl der Abfragen des BSP Baumes wird dadurch erheblich reduziert.

---

<sup>1</sup>Im Gegensatz zu dem allgemeinen Verfahren, bei dem für jede Reflexion und für jede Transmission eine Schnittpunktberechnung mit dem BSP Baum durchgeführt werden muss. Andererseits können nur so Effekte berücksichtigt werden, bei denen beide Strahlen ihre Richtung ändern (so z.B. nach einer Brechung).

## 6.2 Identifikation der Ausbreitungspfade

Die Normierung der Strahlendichte bzw. die Identifikation bereits bekannter Ausbreitungswege stellt einen wichtigen Bestandteil eines deterministischen strahlenoptischen Verfahrens dar. Durch die endliche Ausdehnung der Detektionskugel bzw. die Überlappung konischer Strahlen kann ein Empfänger von mehreren Strahlen auf demselben Ausbreitungsweg getroffen werden (Bild 6.1a). Die Berücksichtigung aller Strahlen würde zu einer falschen Gewichtung des Ausbreitungsweges in der Summe aller Pfade bedeuten. Die Normierung der Strahlendichte hat die Aufgabe, die Mehrfachdetektion eines bereits bekannten Ausbreitungsweges zu verhindern. Sie bestimmt die Gewichtungsfaktoren für jeden Strahl in Abhängigkeit der Krümmungsradien der Wellenfronten (DIDASCALOU et al., 1999). Das gleiche Ergebnis wird erzielt, indem jeder Strahl mit allen, dem Empfänger bereits bekannten Ausbreitungspfaden verglichen wird (SCHÖBERL, 1997; VORLÄNDER, 1989). Neuere Verfahren nutzen disjunkte Strahlen, um eine Mehrfachdetektion eines Pfades durch die fehlende Überschneidung der Strahlen prinzipiell zu unterbinden (DRUMM und LAM, 1999; FUNKHOUSER et al., 1998; LOTT, 2002).



**Bild 6.1:** Aufgrund der endliche Ausdehnung der Detektionskugel bzw. der entsprechenden Überlappung der konischen Strahlen kann ein Empfänger mehrfach von Strahlen über den selben Ausbreitungsweg getroffen werden. Ein Aussortieren der Mehrfachtreffer ist nötig, um die richtige Gewichtung des Pfades in der Summe aller Ausbreitungswege zu erhalten.

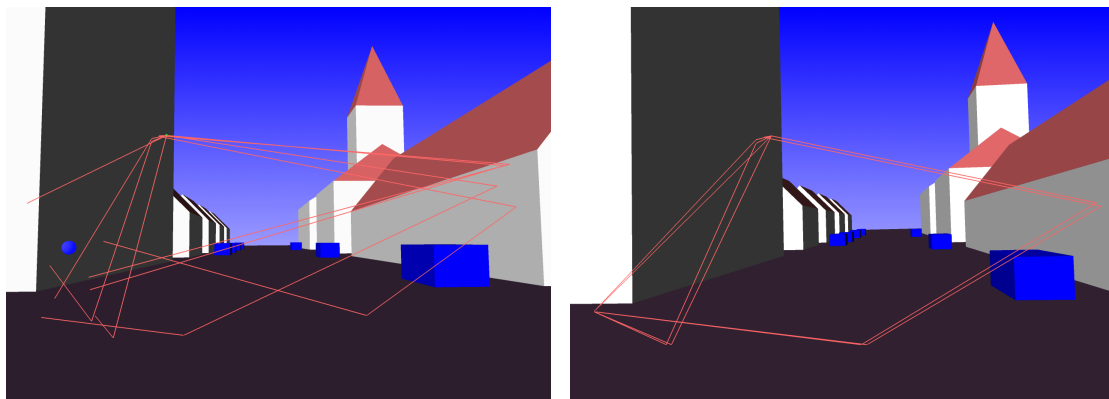
Das eingesetzte Verfahren zur Identifizierung der Ausbreitungswege gehört in die Kategorie der Ausschlussverfahren, die einen Vergleich des Ausbreitungsweges mit allen dem Empfänger bereits bekannten Pfaden erforderlich machen. Um diese Prüfung zu ermöglichen, muss in jedem Strahl eine Liste der getroffenen Objekte abgelegt werden. Diese Liste muss in jedem getroffenen Empfänger mit den Listen der bereits bekannten Ausbreitungswege verglichen werden. Der Vergleich wird bei einer Vielzahl der Interaktionen des Strahls mit der Geometrie sehr aufwendig, da für ein positives Ergebnis die Reihenfolge der getroffenen Objekte für alle Pfade verglichen werden muss.

Um den Aufwand zu verringern, werden in dieser Arbeit Strahlen mit einem Marker versehen, der im Empfänger durch eine Binärbaumsuche schnell mit den Markern der bekannten Ausbreitungswege verglichen werden kann. Der Marker muss für jeden Ausbreitungsweg mit einer hohen Wahrscheinlichkeit eindeutig sein und seine Berechnung darf keinen wesentlichen Mehraufwand für die Ray Tracing Schleife bedeuten. Weiter wäre es

wünschenswert, dass der Wert der Marker eine gleichmäßige Zufallsverteilung bildet, damit die Binärbäume in den Empfängern balanciert sind. Für die Berechnung des Markers wurde in dieser Arbeit die *Cyclic Redundancy Checksum* (CRC) mit einer Länge von 32 bit eingesetzt (PRESS et al., 1988). Die so berechneten Marker sind mit einer sehr hohen Wahrscheinlichkeit eindeutig und die Berechnung des CRC Markers kann sehr effizient mit Hilfe von Lookup-Tabellen realisiert werden. Ist eine absolute Eindeutigkeit der Identifizierung gefordert, so muss auf das klassische Verfahren des Listenvergleichs zurückgegriffen werden. Dieses Verfahren kann ebenfalls verbessert werden, indem die Objektkennungen in einer Liste mit binären Bäumen abgelegt werden. Die Verteilung der zufällig gewählten aber eindeutigen Objektkennungen sollte gleichmäßig sein, um eine Konstruktion zufällig balancierter Bäume zu erreichen.

### 6.3 Kombination mit der Spiegelungsmethode

Die Kombination der Strahlabschussmethode mit der Spiegelungsmethode wurde zuerst von VORLÄNDER (1989) zu einer effizienten Berechnung der Auralisierung in der Akustik eingesetzt. SCHÖBERL (1997) setzte diese Methode bei der polarimetrischen Untersuchung der Wellenausbreitung in pikozellularen Funknetzen ein. Die Kombinationsmethode setzt zuerst die Strahlabschussmethode ein, um in einem ersten Schritt einen möglichen Ausbreitungsweg von dem Sender zum Empfänger zu finden (Bild 6.2a). Dieser Ausbreitungsweg trifft nicht genau den Ort des Empfängers, der Empfänger befindet sich jedoch innerhalb des Strahlkonus. Wird ein neuer möglicher Ausbreitungsweg detektiert, so wird mit Hilfe der Spiegelung des Empfängers entlang des Ausbreitungsweges die neue Abschussrichtung des Strahls bestimmt, so dass der resultierende neue Strahl genau den Ort des Empfängers trifft (Bild 6.2b). Nach der Berechnung der neuen Abschussrichtung muss der korrigierte Strahl erneut durch das Szenario verfolgt werden, da durch die Richtungskorrektur der neue Ausbreitungsweg durch Objekte abgeschattet werden könnte.



(a) Ergebnis der Strahlabschussmethode

(b) Korrektur durch die Spiegelungsmethode

**Bild 6.2:** Zur Kombination der Strahlabschussmethode mit der Spiegelungsmethode. Im ersten Fall treffen Strahlen nicht genau die Mitte des Empfängers (links im Bild). Durch die anschließende Anwendung der Spiegelungsmethode werden die richtigen Pfade gefunden (Szenario aus Schöberl (1997)).

Die Kombinationsmethode hatte sich sehr gut bei der Berechnung von Reflexionen und

Transmissionen in der vereinfachten Geometrie bewährt. Unter diesen Voraussetzungen werden Pfade konstruiert, die eindeutig das Fermatsche Prinzip erfüllen. Bei der Anwendung der Spiegelung auf komplexen Geometrien mit gekrümmten Flächen sowie bei der Berechnung der Brechung und der Beugung von Strahlen führt das *Fermatsche Prinzip* zu einer nichtlinearen Funktion mit vielen lokalen Minima. Die Suche nach der Lösung des Problems ist dann mit einem erheblichen Aufwand verbunden. Die Spiegelung von Beugungsquellen wird im Abschnitt 6.6 behandelt.

## 6.4 Räumliche Abtastung

Die Aufgabe des Strahlengenerators im deterministischen Ray Tracing Modell besteht darin, den Raum nach möglichen Ausbreitungswegen abzutasten. Seine Effizienz ist durch die Anzahl der Strahlen bestimmt, die für das Auffinden aller relevanten Ausbreitungspfade benötigt werden. Es ist daher das Ziel einer Strahlengenerierung, das Wellenausbreitungsproblem mit der Berechnung der minimal möglichen Anzahl der Strahlen zu lösen.

### 6.4.1 Statistische Strahlengenerierung

Ein stochastischer Strahlengenerator verteilt die Strahlen nach einem zufälligen Muster mit einer vorgegebenen Verteilungsdichte über die Oberfläche der Einheitskugel. Die Verteilungsdichte kann entweder als konstant angenommen oder an die jeweilige Antennencharakteristik angepasst werden (SCHÖBERL, 1997). Diese Art des Strahlengenerators wird in stochastischen Ray Tracing Simulationen eingesetzt und ist somit ein Teil des physikalischen Modells. Dagegen ist im deterministischen Ray Tracing Modell vor allem die Strahlengenerierung mit einer konstanten Verteilung der Strahlendichte von Interesse, da in diesem Modell der Raum unabhängig von den Antenneneigenschaften analysiert werden soll.

Eine konstante Verteilungsdichte auf einer Sphäre kann mit Hilfe einer mit einem geeigneten Zufallszahlengenerator erzeugten Gleichverteilung  $\mathcal{U}(-1, +1)$  über dem Intervall  $[-1, +1]$  realisiert werden (VORLÄNDER, 1989) :

$$\begin{aligned}\phi &= \pi\mathcal{U}(-1, +1) \\ \theta &= \arccos(\mathcal{U}(-1, +1)) \quad .\end{aligned}\tag{6.1}$$

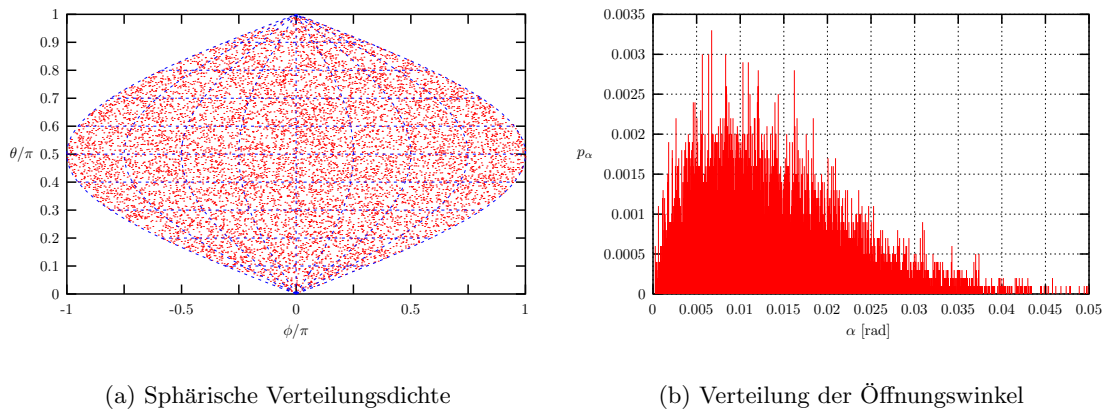
Die Winkel  $\theta$  und  $\phi$  geben die Richtung des Strahls im globalen sphärischen Koordinatensystem an. Die resultierende Verteilungsdichte ist für  $N = 10000$  Strahlen im Bild 6.3 zu sehen. Ebenfalls dargestellt ist die Verteilungsfunktion der Öffnungswinkel der Strahlkoni  $p_\alpha(\alpha, N)$ , die aus dem selben Datensatz berechnet wurde.

Der erforderliche Öffnungswinkel der konischen Strahlen wird durch den mittleren Winkelabstand von zwei benachbarten Strahlen bestimmt. Die im Bild 6.3b dargestellte Verteilungsfunktion kann mit einer Rayleigh Verteilung angenähert werden, die eine Bestimmung des Öffnungswinkels durch die Lösung des folgenden Integrals ermöglicht:

$$\int_0^{\alpha_0(N)} p_\alpha(\alpha, N) d\alpha = \gamma \quad .\tag{6.2}$$

Der Wert  $\gamma \in (0,1)$  gibt den prozentuellen Anteil der Strahlen, die einen kleineren Winkelabstand  $\alpha$  als die anvisierte Grenze  $\alpha_0(N)$  haben müssen.





**Bild 6.3:** Die Verteilungsdichte eines zufallsgesteuerten Strahlengenerators mit der zugehörigen Verteilung des Winkelabstands der Strahlen.

## 6.4.2 Deterministische Strahlengenerierung

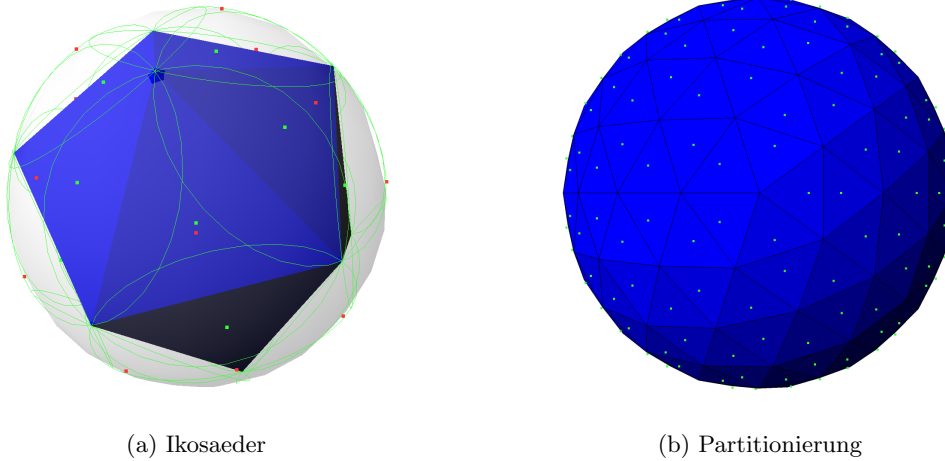
Bei der deterministischen Strahlengenerierung werden die Abschusswinkel der Strahlen in einer vorhersagbaren Weise bestimmt. Dadurch ist auch der Winkelabstand von zwei aufeinanderfolgenden Strahlen deterministisch bestimmt. Das Erreichen einer konstanten Verteilungsdichte der Strahlen auf der Kugeloberfläche gestaltet sich jedoch schwierig. Im Rahmen dieser Arbeit wurden zwei Ansätze zur deterministischen Strahlengenerierung untersucht. Der erste Ansatz stützt sich auf eine rekursive Triangulation eines Platonischen Körpers. Bei dem zweiten Ansatz wird das Problem auf die Minimierung der Energie von Punktladungen auf der Kugeloberfläche zurückgeführt und durch Punkte auf einer verallgemeinerten Spirale näherungsweise gelöst.

### 6.4.2.1 Rekursive Partitionierung Platonischer Körper

Der klassische Weg, eine näherungsweise konstante und deterministische Punkteverteilung auf einer Kugel zu erhalten, besteht in der rekursiven Verfeinerung eines *Platonischen Körpers* (WEISSTEIN, 2002). Als Platonische Körper werden alle regulären konvexen Polyeder bezeichnet, deren Oberfläche aus dem selben regelmäßigen Vieleck bestehen und in jeder Ecke des Körpers gleich viele dieser Vielecke zusammenstoßen. Ihr Name geht auf Platon zurück, der erstmals zeigen konnte, dass es nur fünf solche reguläre konvexe Polyeder gibt: Tetraeder, Hexaeder, Oktaeder, Dodekaeder und Ikosaeder.

Für eine rekursive Triangulation sind die Körper mit dreieckigen Flächen besonders gut geeignet (Tetraeder, Oktaeder und Ikosaeder). Durch die Halbierung ihrer Seiten kann jede dreieckige Fläche in vier neue Dreiecke aufgeteilt werden, deren Eckpunkte anschließend auf die Oberfläche der Einheitskugel verschoben werden. Die rekursive Fortsetzung einer solchen Verfeinerung ergibt als Ergebnis eine polygonale Approximation der Kugeloberfläche (Bild 6.4b). Die Wahl des Platonischen Körpers wirkt sich auf die Abweichung der Verteilungsdichte von der konstanten Verteilung aus. Im folgenden Text wird der Ikosaeder und der Oktaeder betrachtet und die so generierten Verteilungsdichten miteinander verglichen.

Für die Strahlengenerierung muss der Richtungsvektor des Strahls und der zugehörige Öffnungswinkel des Strahlkonus aus den Eckpunkten der Dreiecke berechnet werden.



**Bild 6.4:** Das Modell des Icosaeders (links) und der Partitionierung nach zwei Rekursionsschritten (rechts). Die so entstandene Approximation der Kugeloberfläche besteht aus 320 dreieckigen Flächen.

Jedem Dreieck wird ein Kreis umschrieben, durch dessen Mittelpunkt die Richtung des Strahls festgelegt wird. Der Radius des Kreises bestimmt den Öffnungswinkel des Strahlkonus (Bild 6.4a).

Die analytische Berechnung des Mittelpunktes und des Radius des Kreises geht von der impliziten Definition aller Punkte  $(x,y)$  eines Kreises durch die Eckpunkte  $(x_i,y_i)$   $i = 1, 2, 3$  (WEISSTEIN, 2002):

$$\begin{vmatrix} x^2 + y^2 & x & y & 1 \\ x_1^2 + y_1^2 & x_1 & y_1 & 1 \\ x_2^2 + y_2^2 & x_2 & y_2 & 1 \\ x_3^2 + y_3^2 & x_3 & y_3 & 1 \end{vmatrix} = 0 . \quad (6.3)$$

Nach dem Ausschreiben der Determinante

$$\begin{aligned} a &= + \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \\ b_x &= + \begin{vmatrix} x_1^2 + y_1^2 & y_1 & 1 \\ x_2^2 + y_2^2 & y_2 & 1 \\ x_3^2 + y_3^2 & y_3 & 1 \end{vmatrix} \\ b_y &= - \begin{vmatrix} x_1^2 + y_1^2 & x_1 & 1 \\ x_2^2 + y_2^2 & x_2 & 1 \\ x_3^2 + y_3^2 & x_3 & 1 \end{vmatrix} \\ c &= - \begin{vmatrix} x_1^2 + y_1^2 & x_1 & y_1 \\ x_2^2 + y_2^2 & x_2 & y_2 \\ x_3^2 + y_3^2 & x_3 & y_3 \end{vmatrix} \end{aligned} \quad (6.4)$$

kann die Kreisgleichung (6.3) als

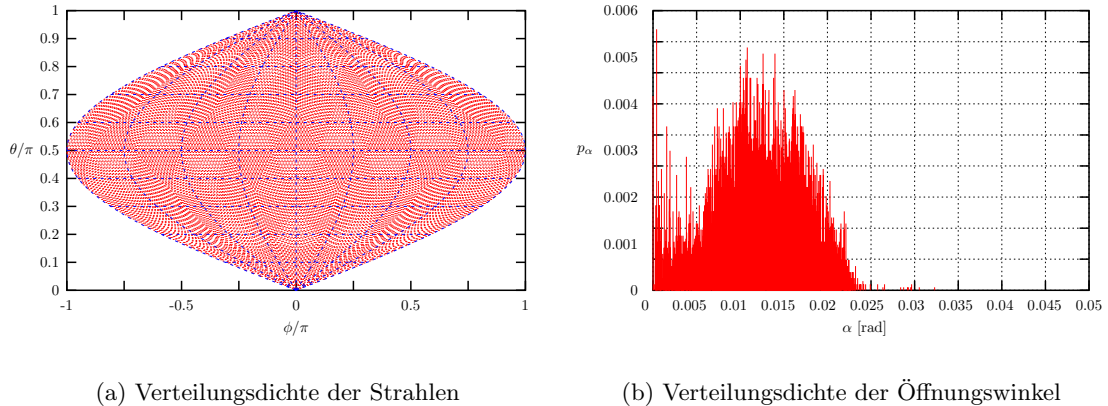
$$a \left( x + \frac{b_x}{2a} \right)^2 + a \left( y + \frac{b_y}{2a} \right)^2 - \frac{b_x^2}{4a} - \frac{b_y^2}{4a} + c = 0 . \quad (6.5)$$

geschrieben werden. Der Mittelpunkt und der Radius des Kreises sind gegeben durch:

$$(x_0, y_0) = \left( \frac{b_x}{2a}, \frac{b_y}{2a} \right), \quad r = \frac{\sqrt{b_x^2 + b_y^2 - 4ac}}{2a}. \quad (6.6)$$

Der Richtungsvektor und der Öffnungswinkel des Strahls ergeben sich aus der Transformation von dem lokalen Koordinatensystem der Ebene des jeweiligen Dreiecks in das globale Koordinatensystem des Szenario. Da nicht alle Dreiecke die gleichen Abmessungen haben, wird somit auch der Öffnungswinkel der Strahlen an die Abschussrichtung automatisch angepasst. Die Koni behalten dabei die kleinste überlappende Fläche bei.

Die Projektion der Strahlendichte des Ikosaedergenerators auf die Fläche der Einheitskugel ist in der Abbildung 6.5 zu sehen. Ebenso angegeben ist die Verteilung der Öffnungswinkel für diese Strahlendichte. Ein besonderer Nachteil aller Generatoren, die auf der Partitionierung Platonischer Körper beruhen besteht darin, dass mit ihnen nicht jede beliebige Strahlendichte eingestellt werden kann. Die resultierende Strahlendichte  $\kappa$



**Bild 6.5:** Die Verteilungsdichte der Strahlen eines auf der Partitionierung des Ikosaeders basierten Generators projiziert auf die Fläche der Einheitskugel. Ebenso angegeben ist die Verteilungsdichte des Öffnungswinkels der Koni. Der Ikosaeder wurde rekursiv in fünf Schritten in 20480 dreieckigen Flächen aufgeteilt.

ist abhängig von der gewählten Rekursionstiefe  $D$  der Triangulation. Sie entspricht der Anzahl der dreieckigen Flächen des partitionierten Ikosaeders und kann leicht mit der folgenden Gleichung bestimmt werden:

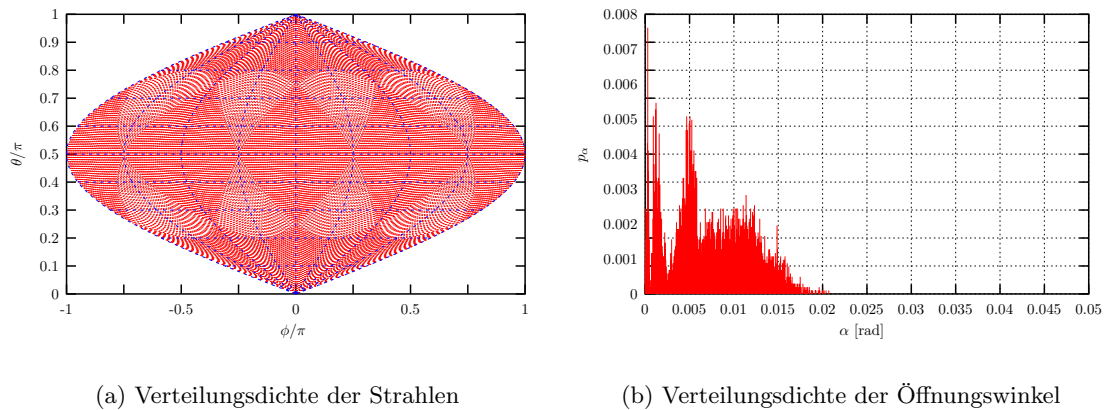
$$\kappa = \frac{F}{4\pi} 4^D. \quad (6.7)$$

$F$  stellt den Formfaktor des Platonischen Körpers dar. Für den Ikosaeder beträgt  $F = 20$ . Häufig ist es erwünscht, statt der Rekursionstiefe die gewünschte Strahlendichte zu definieren und somit indirekt die Rekursionstiefe einzuschränken. Die tatsächliche Strahlendichte liegt damit immer höher als die geplante Strahlendichte, womit auch eine entsprechend längere Simulationslaufzeit verbunden ist. Die für die Strahlendichte  $\kappa$  mindestens notwendige Rekursionstiefe ergibt sich aus:

$$D = \left\lceil \log_4 \left[ \frac{4\pi\kappa}{F} \right] \right\rceil \quad (6.8)$$

wobei  $\lceil x \rceil$  die obere Gaussklammer darstellt. Durch diese wird die Rekursionstiefe so eingestellt, dass die zu  $\kappa$  nächsthöhere Strahlendichte erreicht wird. Da besonders für hohe Rekursionstiefen die Abstände zwischen den diskreten Strahlendichten mit  $4^D$  ansteigen, werden aufgrund der Überabtastung die Rechenzeiten auch mit der vierten Potenz der Rekursionstiefen ansteigen. Dies ist der grundlegende Nachteil dieser Art der Strahlengenerierung.

Zum Vergleich sei noch das Ergebnis der auf einer Partitionierung des Oktaeders ( $F = 8$ ) basierten Strahlengenerierung im Bild 6.6 angegeben. Der ikosaederbasierte Generator erzeugt eine gleichmäßigere Verteilung der Strahlendichte sowie eine kleineren Überlappung der Strahlen und ist deswegen dem Oktaeder vorzuziehen. Bei dem oktaederbasierten Generator sind jedoch die Abstände der diskreten Strahlendichten um den Faktor 2.5 kleiner, was bei einer Anpassung an eine gewünschte Strahlendichte vorteilhafter ist.



**Bild 6.6:** Die Verteilungsdichte der Strahlen eines auf der Partitionierung des Oktaeders basierten Generators projiziert auf die Fläche der Einheitskugel. Ebenso angegeben ist die Verteilungsdichte des Öffnungswinkels der Koni. Der Oktaeder wurde rekursiv in fünf Schritten in 32768 dreieckigen Flächen aufgeteilt.

#### 6.4.2.2 Minimierung der Energie auf der Kugeloberfläche

Der Begriff der verallgemeinerten Spiralpunkte (*Generalized Spiral Points*) wurde zusammen mit dem zugehörigen Algorithmus zu ihrer Erzeugung 1994 von [RAKHMANOV et al. \(1994\)](#) und [ZHOU \(1995\)](#) eingeführt. In ihren Untersuchungen konnten sie zeigen, dass Punkte mit einer solchen Anordnung eine gute Näherungslösung des Problems der minimalen diskreten Energie auf der Oberfläche einer Kugel darstellen. Ausgangspunkt der Untersuchung ist die Bestimmung einer Konstellation von diskreten Punkten auf der Kugel derart, dass ihre diskrete Gesamtenergie im Potential  $V(r)$  minimal wird. Das Potential  $V(r)$  wurde mit einer  $r^\beta$  Entfernungsabhängigkeit angenommen, wobei  $r$  die Euklidische Distanz zwischen zwei Punkten ist und  $-2 < \beta < 2$ . Das Problem wurde für  $\beta = -1$  in der Form einer optimalen Position von gleichgeladenen Punktladungen auf einer Kugel von J. J. Thomson ([THOMSON, 1904](#)) formuliert. Diese Problemklasse tritt häufig in der Chemie (z.B. bei Fullerenen), der Physik und der Kristallographie auf ([RAKHMANOV et al., 1994](#)). Es existiert keine allgemeine und geschlossene Lösung des Problems für eine beliebige Anzahl von Partikel. Die beste bisher bekannte Strategie besteht aus der iterativen

Minimierung der Gesamtenergie mit der Newton-Methode. Diese beinhaltet jedoch eine sehr rechenaufwendige Lösung eines Gleichungssystems von der Ordnung  $\mathcal{O}(N^2)$  für  $N$  Partikel.

[RAKHMANOV et al. \(1994\)](#) konnte zeigen, dass die Position der Punkte auf der verallgemeinerten Spirale das Problem der Energieminimierung bis auf einen Restfehler von der Ordnung  $\mathcal{O}(N^{1-\beta/2})$  löst. Der positive Nebeneffekt dieser Anordnung besteht jedoch darin, dass die so gewonnenen Punkte in einer deterministischen Weise nahezu gleichmäßig in einem regulären hexagonalen Gitter auf der Kugeloberfläche verteilt sind. Mit Hilfe der folgenden Hilfsvariablen:

$$\begin{aligned} h_k &= 2 \frac{k-1}{N-1} - 1 \quad \text{für } k = 1 \dots N \\ \Delta\phi_k &= \frac{C}{\sqrt{N}} \frac{1}{\sqrt{1-h_k^2}} \end{aligned} \quad (6.9)$$

lässt sich die Menge der  $N$  Punkte auf der verallgemeinerten Spirale durch die folgende Iteration angeben, die die Lage der Punkte in sphärischen Koordinaten  $(\theta_k, \phi_k)$  definiert:

$$\begin{aligned} \theta_k &= \arccos(h_k) \\ \phi_k &= \begin{cases} 0 & \text{für } k = 1, N \\ (\phi_{k-1} + \Delta\phi_k) \pmod{2\pi} & \text{für } k = 2 \dots N-1 \end{cases} \end{aligned} \quad (6.10)$$

Die Konstante  $C$  wurde mit dem Wert 3.6 angegeben ([RAKHMANOV et al., 1994](#)). Für eine sich überlappende Partitionierung der Kugeloberfläche in  $N$  Teile wurde der maximal mögliche Durchmesser  $d_N$  für jeden Kreis von [VAN DER WAERDEN \(1952\)](#) hergeleitet:

$$d_N = \frac{4}{\sqrt{N}} \cdot \sqrt{\frac{2\pi}{\sqrt{27}}} (1 + \varepsilon_N) \quad \varepsilon_N > 0, \varepsilon_N \rightarrow 0 \quad \text{wenn } N \rightarrow \infty \quad (6.11)$$

Als Folge der obigen Formel kann eine Kugeloberfläche gleichmäßig mit  $N$  identischen sphärischen und sich überlappenden Kappen mit dem Durchmesser  $d_N$  abgedeckt werden ([RAKHMANOV et al., 1994](#)). Die obere Grenze für den Durchmesser bei einer flächenregulären (sich nicht überlappenden) Partitionierung wurde von [ZHOU \(1995\)](#) hergeleitet und beträgt:

$$d_N = \frac{7}{\sqrt{N}} \quad (6.12)$$

Der Öffnungswinkel der Strahlen kann mit

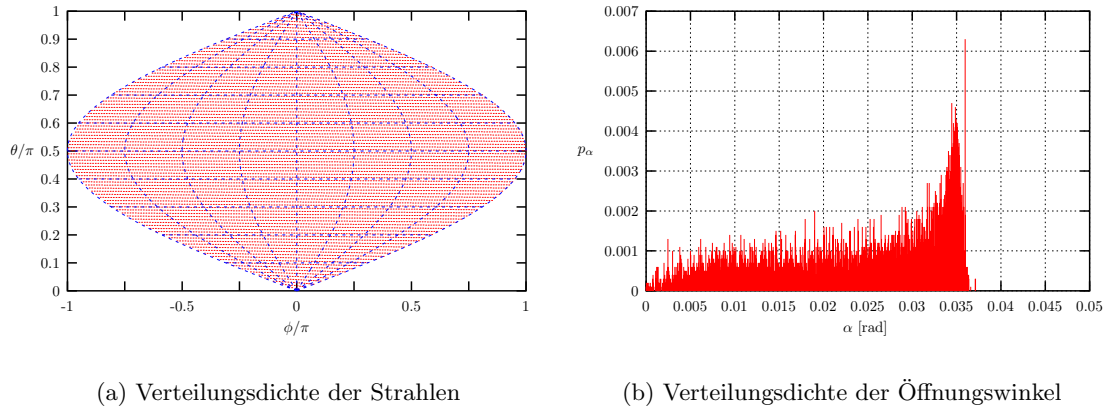
$$\alpha = \arcsin \left( \frac{2}{\sqrt{N}} \cdot \sqrt{\frac{2\pi}{\sqrt{27}}} (1 + \varepsilon_N) \right) \quad \text{mit } \varepsilon_N > 0.59144 \quad (6.13)$$

approximiert werden. Im praktischen Einsatz hatte sich für  $\varepsilon_N$  ein Wert von 0.8 bewährt. Ein anderer Weg, den Öffnungswinkel der Strahlen zu berechnen, besteht in der Auswertung des Winkelinkrements  $\Delta\phi_k$ . Dieser Wert gibt direkt die Winkelentfernung von zwei aufeinanderfolgenden Strahlen an, und kann somit als Öffnungswinkel des Strahls interpretiert werden:

$$\alpha_k = \arcsin(\Delta\phi_k) \quad \text{für } k = 1 \dots N \quad \text{und } |\Delta\phi_k| \leq 1, \quad (6.14)$$

mit dem Winkelinkrement  $\Delta\phi_k$  aus Gleichung (6.9).

Die resultierende Verteilungsdichte der Strahlen und Verteilung der Öffnungswinkel ist im Bild 6.7 für  $N = 10000$  zu sehen. Im Vergleich zu der Strahlengenerierung mittels Platonischer Körper kann mit diesem Strahlengenerator eine beliebige Strahlendichte realisiert werden. Die Berechnung der Öffnungswinkel gestaltet sich ebenfalls deutlich einfacher, jedoch ist die Überlappung benachbarter Strahlen auch entsprechend größer, was zu einer höheren Zahl von Mehrfachdetektionen desselben Ausbreitungsweges führt.



**Bild 6.7:** Die Strahlendichteverteilung eines deterministischen Strahlengenerators auf der Basis der Minimierung der Gesamtenergie auf der Kugeloberfläche.

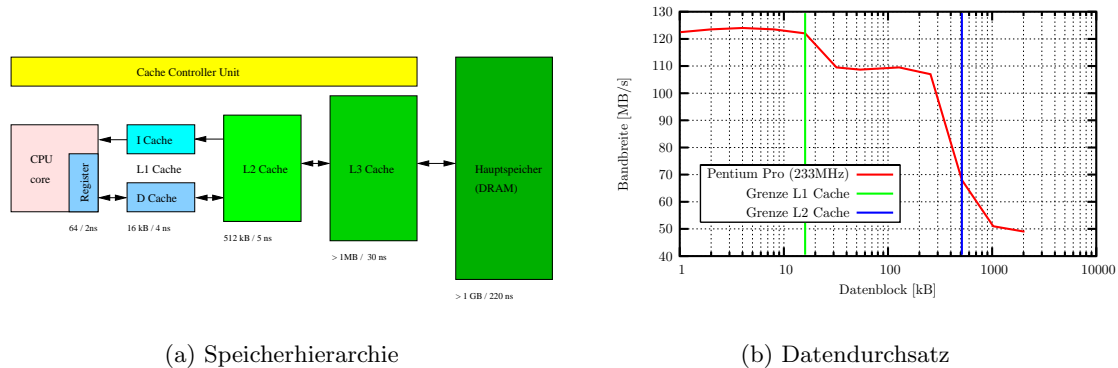
Die Umsetzung der Iteration (6.10) kann für große  $N$  zu Problemen im Zusammenhang mit der endlichen Wortbreite der Fließkommazahlen in einem Rechner und den damit verbundenen Rundungsfehlern bei der Addition des Winkelinkrements  $\Delta\phi_k$  führen (DOWD und SEVERANCE, 1998). Durch das Aufspalten der Summe  $\phi_k$  in Teilsummen für die verschiedenen Skalen kann das Auftreten der Rundungsfehler deutlich reduziert werden.

### 6.4.3 Speicherzugriffsmuster und Effizienz

Der Prozess der Strahlengenerierung hat auch indirekt Einfluss auf die Geschwindigkeit des Ray Tracing, da aktuelle Rechnerarchitekturen eine ausgeprägte Speicherhierarchie besitzen. Eine solche Hierarchie ist nötig, um den Prozessor ausreichend schnell mit Daten versorgen zu können. Die Datenzugriffsgeschwindigkeit nimmt mit steigender Speichergröße ab. Um also den Prozessor schnell mit den benötigten Daten versorgen zu können, werden zwischen den großen Hauptspeicher und den Prozessor eine Reihe schneller Zwischenspeicher (*Caches*) mit abnehmender Größe zwischengeschaltet. Der dem Prozessor am nächsten liegende Cache ist der L1 Cache. Dieser Speicher hat eine typische Größe von 16 - 64 kB und ist in den meisten Fällen in einen Instruktions- und Datenspeicher aufgeteilt, da sich die Zugriffsmuster auf beide Bereiche grundlegend unterscheiden. Der übergeordnete L2 Cache besitzt eine Größe von 128 - 512 kB, kann aber auch je nach Prozessor deutlich größer dimensioniert sein. Es wird sowohl für Daten als auch für Instruktionen verwendet und ist mit dem L1 Cache auf dem Prozessorchip integriert. Der letzte in der Hierarchie ist der L3 Cache, der einige MB groß ist und die Brücke zum dynamischen Hauptspeicher darstellt. Die Zugriffsgeschwindigkeiten verhalten sich umgekehrt proportional zu der Cachegröße. Der Zugriff wird von dem integrierten Cache Controller koordiniert. Greift der Prozessor auf eine Speicherstelle zu, so prüft der Cache Controller



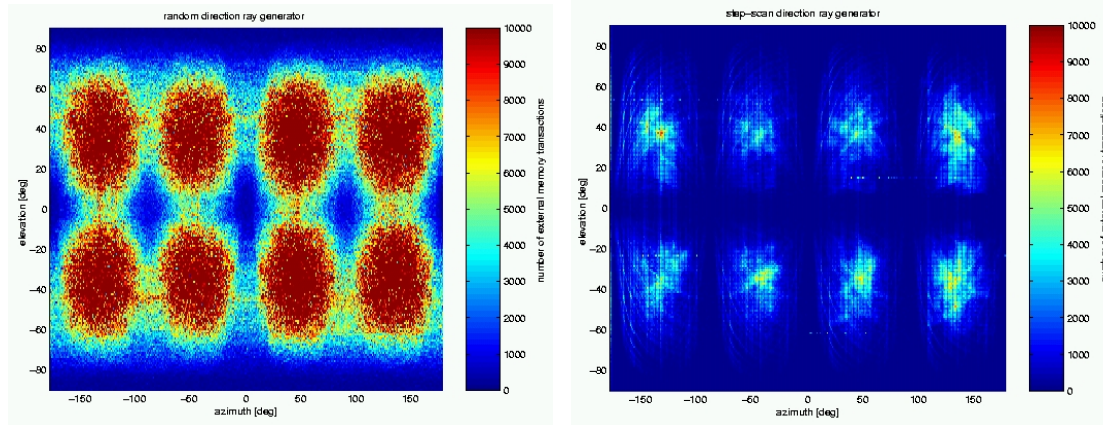
nach, ob sich die Daten bereits in einem der Caches befindet. Ist dies nicht der Fall, müssen die Daten aus dem langsamen Hauptspeicher geladen werden, es tritt ein *Cache Miss* auf. Da eine hohe Zahl solcher Ereignisse die Verarbeitung erheblich verlangsamen kann, ist es sehr wichtig, häufig benutzten Daten in den Caches nahe am Prozessor zu halten. Bild 6.8 stellt die Speicherhierarchie und den Datendurchsatz der Caches dar.



**Bild 6.8:** Der prinzipielle Aufbau und der gemessene Datendurchsatz einer Speicherhierarchie. Die Anzahl der eingesetzten Caches hängt von der anvisierten Anwendung der Prozessorfamilie ab. Im Serverbereich sind L3 Caches mit mehreren MB Größe üblich. Der dargestellte Datendurchsatz wurde auf einem handelsüblichen PC gemessen.

Durch die Generierung eines Strahls wird gewissermaßen die Adressierung der Datenstrukturen im Speicher vorgegeben. Werden zwei aufeinanderfolgende Strahlen in wesentlich unterschiedliche Richtungen gesendet, treffen sie sehr wahrscheinlich unterschiedliche Objekte, die an verschiedenen Stellen im Hauptspeicher abgelegt sind. Da die Zwischenspeicher nur eine begrenzte Größe besitzen, ist zu erwarten, dass in diesem Fall eine hohe Zahl von *Cache Misses* generiert wird. Umgekehrt werden Daten in den Caches gut ausgenutzt, wenn zwei aufeinanderfolgende Strahlen in etwa in die selbe Richtung gesendet werden, da sie mit hoher Wahrscheinlichkeit die selben Objekte treffen. Das Verhalten wird in den Bildern 6.9 deutlich. In diesem Fall wurde der Strahlengenerator in die Mitte eines kubischen Gitters positioniert. Die *Cache Miss Rate* wurde mit Hilfe der im Prozessor eingebauten Ereigniszählern bestimmt und ist in Abhängigkeit von dem Abschusswinkel der Strahlen dargestellt. Es wurde ein zufallsbasiertes und ein deterministisches Generatormodell eingesetzt, wobei das deterministische Modell die Bedingung der räumlichen Nähe der aufeinanderfolgenden Strahlen erfüllte.

Die im Bild 6.9 dargestellte hohe Anzahl der *cache misses* im Fall des zufallsbasierten Strahlengenerators bestätigt die Vermutung, dass dieses Generatorkonzept für die heutigen Rechnerarchitekturen weniger geeignet ist. Eine deterministische Strahlengenerierung zeigt eine deutlich bessere Ausnutzung der Speicherhierarchie. Da mit der zufallsbasierten Strahlengenerierung eine bessere Unterdrückung von *Aliasing-Effekten* erreicht wird, kann eine deterministische Strahlengenerierung um eine zufallsbasierte Komponente erweitert werden, um eine zufällige Variation der lokalen Strahlendichte zu bewirken. Bei einer kleinen Änderung der lokalen Strahlendichte sind keine negativen Auswirkungen bezüglich der effizienten Nutzung der Speicherhierarchie zu erwarten. Das Aliasing-Problem kann aber auch durch den Einsatz einer adaptiven deterministischen Strahlendichtegenerierung gelöst werden, welche die Strahlendichte in der Nähe von Diskontinuitäten der Geometrie künstlich erhöht wird. Dieses Verfahren wird im folgenden Abschnitt vorgestellt.



(a) Zufallsbasierte Strahlengenerierung

(b) Deterministische Strahlengenerierung

**Bild 6.9:** Die gemessene *cache miss* Rate für einen zufallsbasierten und einen deterministischen Strahlengenerator. Für die Geometrie wurde aus Gründen der Anschaulichkeit ein kubisches Gitter gewählt. Der Sender befand sich in der Mitte der Gitterzelle.

## 6.5 Adaptive Cone Tracing Verfahren

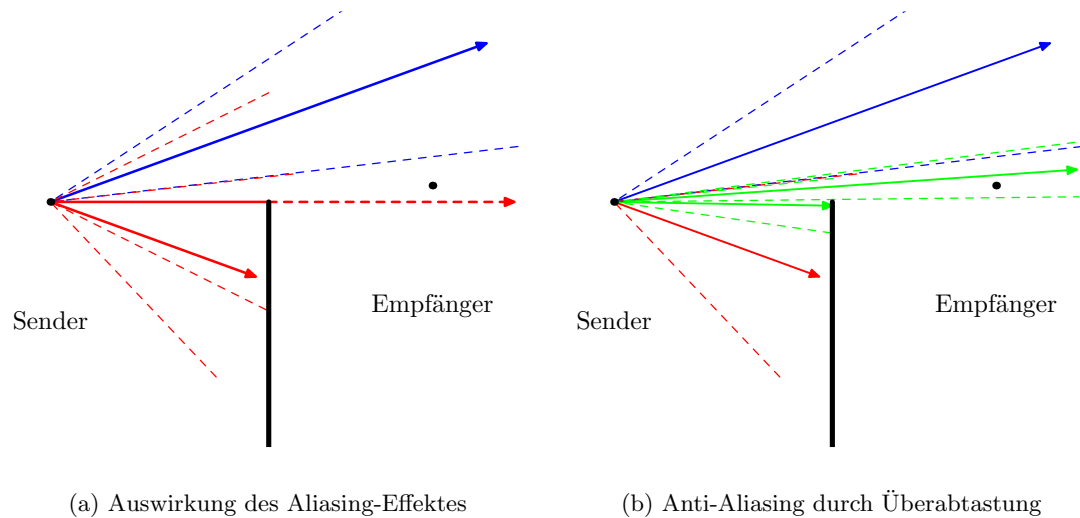
Eine adaptive Anpassung der Strahlendichte an die Geometrie des Szenarios ist in mehrfacher Hinsicht vorteilhaft. Sie hilft, die Gesamtzahl der benötigten Strahlen zu verringern, indem Gebiete ohne geometrische Strukturen mit nur wenigen Strahlen abgetastet werden. Dagegen wird die Strahlendichte in die Richtungen erhöht, in denen sich die Geometrie abrupt ändert. Dazu gehören zum Beispiel die Richtungen der Polygonkanten oder ihre Projektion aus der Sicht des Senders. Die durch eine Adaption erreichte Überabtastung des Raumes in der Nähe einer Kante führt nebenbei zu einer deutlichen Verringerung des Aliasing an der Kante. Ein anderer Nebeneffekt ist eine deutliche Reduktion der Mehrfachdetektion der Empfänger.

Aliasing entsteht dadurch, dass ein Empfänger, der sich hinter einer Polygonkante befindet aber durch die Kante nicht abgeschattet ist, aufgrund einer zu niedrig gewählten Strahlendichte nicht detektiert wird (Bild 6.10a). Der Suchstrahl, in dessen Kegel der Empfänger liegt, trifft auf das Polygon und wird in Abhängigkeit der Materialparameter nicht weiterverfolgt. Das durch Aliasing verursachte treppenförmige Muster, welches alle, durch klassisches Ray Tracing berechneten, Schattengrenzen hinter Kanten auszeichnet, kann besonders im Bereich der Visualisierung veranschaulicht werden. Anti-Aliasing Verfahren verringern die optisch störende Wirkung des Aliasing-Effektes, indem die Dichte der Strahlen in der Nähe einer Kante stark erhöht wird (Bild 6.10b).

Eine Möglichkeit, die Auswirkung des Aliasing zu verringern, besteht ebenso wie in der Computergraphik in der Erhöhung der Strahlendichte, mit der die Geometrie in der Nähe der Kante abgetastet wird. Bei den klassische Strahlengeneratoren aus Abschnitt 6.4.2 würde sich diese Erhöhung der Strahlendichte unabhängig von der Geometrie in alle Richtungen gleich auswirken. So würden in einem typischen Szenario, wie im Bild 5.1 dargestellt, die Bereiche ohne geometrische Strukturen mit der gleichen Strahlendichte abgetastet werden wie Bereiche mit vielen Polygonen. Das Ergebnis sind unnötig lange Simulationslaufzeiten.

Eine adaptive Anpassung der Strahlendichte erfordert eine Rückwirkung des Cone Tra-



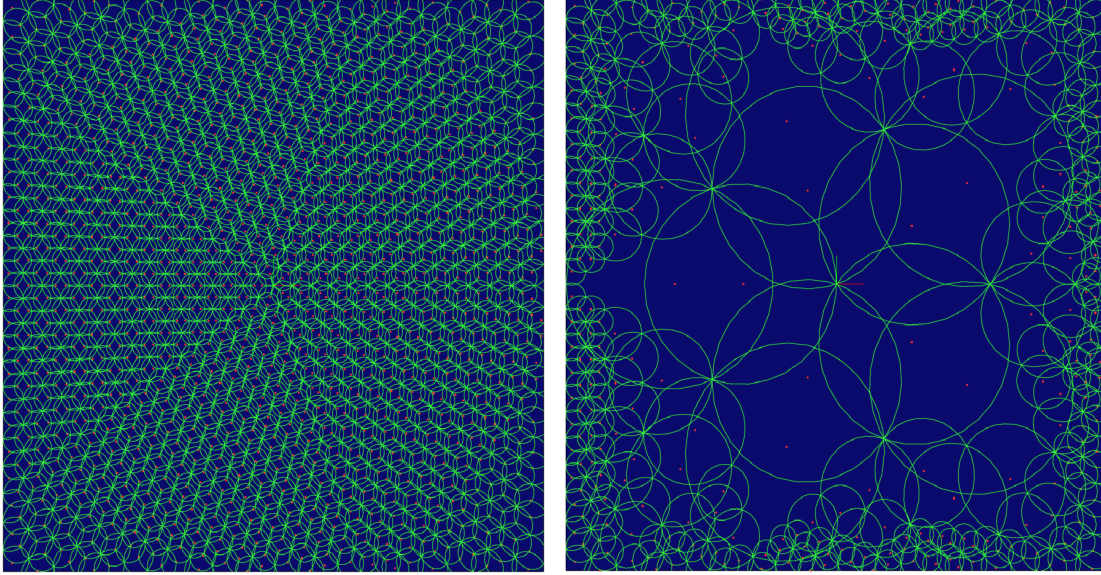


**Bild 6.10:** Das Aliasing entsteht durch die räumliche Unterabtastung der Geometrie. Eine Lösung des Aliasing-Problems besteht in der Erhöhung der Strahldichte in der Nähe der Kanten.

cing Algorithmus auf den Strahlengenerator. Da beide Komponenten in einem Simulator nur über die Datenstruktur der Strahlen kommunizieren, muss ein Weg gefunden werden, einen Rückfluss der Information von der Strahlverfolgung zum Strahlengenerator zu gewährleisten, ohne eine Auswirkung auf die Verwendbarkeit anderer Strahlengeneratoren zu haben. Eine Kopplung kann einfach realisiert werden, indem die Datenstruktur eines Strahls um eine Boolesche Variable erweitert wird, die von der Strahlverfolgung gesetzt wird, wenn der Strahl in Teilstrahlen aufgeteilt und somit die Strahldichte in die Richtung des Strahls erhöht werden soll. Der Strahlengenerator muss diese Variable vor der Generierung neuer Strahlen berücksichtigen und entsprechend eine Aufteilung in neue Strahlen vornehmen. Dieser Weg bietet dem Strahlengenerator ebenfalls eine einfache Möglichkeit, der Strahlverfolgung mitzuteilen, dass eine weitere Unterteilung nicht möglich ist, wenn die maximale, von dem Benutzer festgelegte Strahldichte erreicht worden ist.

Die Unterteilung eines Strahls in Teilstrahlen kann sehr einfach mit der Hilfe des Ikosaeder-Generators geschehen. Durch diesen Generator wird in jedem Triangulationsschritt ein Strahl in vier Teilstrahlen unterteilt und die lokale Strahldichte somit um den Faktor vier erhöht. Die untere sowie die obere Grenze der rekursiven Partitionierung wird aus den Benutzerangaben der gewünschten minimalen und maximalen Strahldichte nach Gleichung (6.8) berechnet.

Die Strahlverfolgung hat die Aufgabe zu entscheiden, ob eine Erhöhung der Strahldichte in Richtung des aktuellen Strahls erforderlich ist. Diese Entscheidung wird anhand der Lage des Strahls relativ zu den umgebenden Polygonen getroffen. Eine Erhöhung der Strahldichte wird immer dann vorgenommen, wenn sich ein Polygon nur teilweise innerhalb des Strahlkegels befindet. Es muss also die Lage der in der Umgebung des Strahlkegels befindlichen Polygonkanten untersucht werden. Dabei müssen 'echte' Polygonkanten von den Kanten, die durch eine Partitionierung des Polygons entstanden sind, unterschieden werden. Die Information über die lokale Geometrie der Kante wird durch eine Analyse der topologischen Information des BSP Baumes gewonnen.



(a) Konstante Strahlengenerierung

(b) Adaptive Strahlengenerierung

**Bild 6.11:** Durch eine adaptive Anpassung der Strahlendichte an die Geometrie kann die Gesamtzahl der Strahlen verringert werden. Die Konturen des Schnittes der konischen Strahlen mit dem Polygon sind eingezeichnet. Der Sender befand sich mittig über dem Polygon.

Die folgende Betrachtung muss für jeden Teilstrahl eines Strahls durchgeführt werden. In einem ersten Schritt muss festgestellt werden, welche Kanten sich überhaupt im Kegel des Partialstrahls befinden können. Diese Information wird durch eine Abfrage des BSP Baumes gewonnen, indem festgestellt wird, welche Partitionierungsebenen das Volumen des Teilstrahlkegels schneiden. Da eine ähnliche Berechnung bereits bei der Bestimmung der Schnittpunkte des Strahls mit dem BSP Baum durchgeführt werden muss, können beide Abfragen miteinander kombiniert werden. Dadurch wird eine erneute Abfrage des BSP Baumes vermieden. In einem zweiten Schritt muss die Lage der Kanten in den betreffenden Ebenen relativ zum Strahlkegel bestimmt werden. Dies geschieht durch die Berechnung der Schnittpunkte der Kantenlinie mit dem abgeschnittenen Kegel des Teilstrahls und eine anschließende aufwendige Klassifizierung aller möglichen Orientierungen des Liniensegments der Kante relativ zum Kegel.

Zur Berechnung des Schnittpunktes einer Linie mit einem Kegel wird die Position eines Punktes auf der Kantenlinie durch die folgende Gleichung definiert:

$$\vec{X}(t) = \vec{P} + t\vec{D} . \quad (6.15)$$

Die Kegeloberfläche ist durch die Position des Vertex  $\vec{V}$ , den Axialvektor  $\vec{A}$  und den Öffnungswinkel des Kegels  $\theta$  definiert:

$$\vec{A} \cdot \left( \frac{\vec{X} - \vec{V}}{|\vec{X} - \vec{V}|} \right) = \cos \theta . \quad (6.16)$$

Dabei wird angenommen, dass der Winkel  $\theta$  spitz ist ( $\theta \in [0, \pi/2]$ ) und der Kosinus des Winkels positive Werte annimmt (SCHNEIDER und EBERLY, 2002). Die obige Gleichung

wird quadriert und der Vektor  $\vec{X}$  wird durch die Parameterform der Geraden  $\vec{X}(t)$  aus Gleichung (6.15) ersetzt. Die Gleichung (6.16) beschreibt einen Doppelkegel. Um Mehrdeutigkeiten zu vermeiden, wird zu einer zusätzliche Einschränkung der Lösungsmenge die Annahme gemacht, dass sich ein gültiger Schnittpunkt nur im positiven Halbraum der durch die Gleichung  $\vec{A} \cdot (\vec{X} - \vec{V}) = 0$  definierten Ebene befindet. Die Beschränkung der Lösungsmenge auf den abgeschnittenen Kegel des Teilstrahls wird auf die gleichen Weise durchgeführt.

Die quadratische Gleichung (6.16) kann in der Form

$$(\vec{X} - \vec{V})^T M (\vec{X} - \vec{V}) = 0 \quad (6.17)$$

mit  $M = (\vec{A}\vec{A}^T - \cos^2 \theta I)$  geschrieben werden (SCHNEIDER und EBERLY, 2002). Die Lösung der quadratischen Gleichung besteht bei einem positiven Wert der Determinante aus zwei verschiedenen Schnittpunkten. Bei einem negativen Wert der Determinante sind beide Lösungen imaginär und es existiert kein Schnittpunkt. Für den Fall einer verschwindenden Determinante berührt die Linie die Kegeloberfläche tangential in einem Punkt.

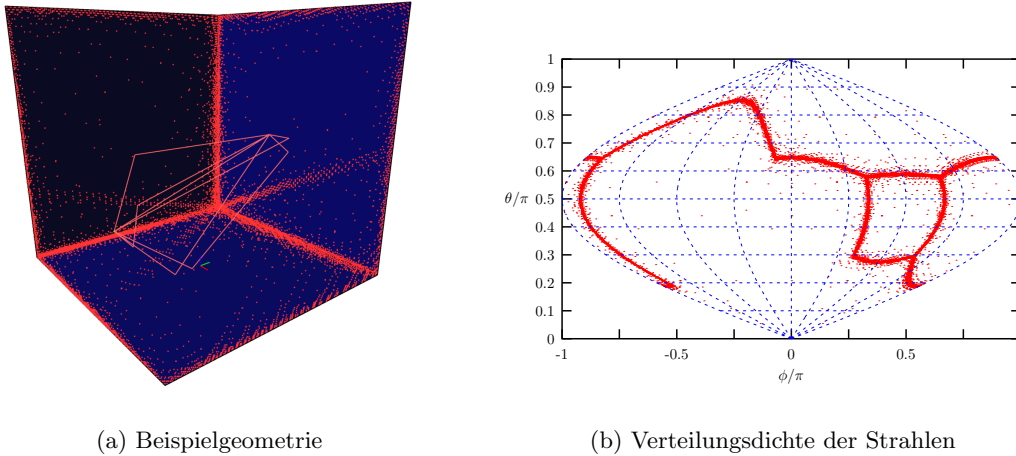
Nach einer Berechnung der Schnittpunkte der Kantenlinie mit dem Kegel des Strahls muss die Lösung auf den abgeschnittenen Kegel des Teilstrahls eingeschränkt werden. Dies geschieht durch eine Berechnung der Schnittpunkte der Linie mit den Ebenen, die die Oberseite und die Unterseite des abgeschnittenen Kegels definieren, und einem anschließenden Vergleich der Lösung mit den Schnittpunkten des Kegels. Zuletzt muss die Lösung mit der endlichen Ausdehnung der Kante verglichen und die Lage der Kante als 'außerhalb', 'teilweise außerhalb' oder 'innerhalb' des Kegels klassifiziert werden. Für die Entscheidung zur Erhöhung der Strahlendichte ist lediglich eine Klassifizierung 'getroffen' oder 'nicht getroffen' nötig. Da jedoch diese Funktion ebenfalls die Lage der möglichen Beugungsquellen auf der Kante berechnen soll, wird hier eine genauere Einordnung der Lage vorgezogen. Abschließend müssen, falls keine weitere Adaption der Strahlendichte erfolgen soll, die Positionen der Beugungsquellen auf der Kante berechnet werden.

Ein Vergleich der Abtastung mit einer konstanten und einer adaptierten Strahlendichte ist in den Bildern 6.11a und 6.11b dargestellt. Das quadratische Polygon wird von einer Quelle abgetastet, die mittig über dem Polygon positioniert ist. Zusätzlich zu den Auftreffpunkten der Strahlen auf dem Polygon sind die Schnitte der Strahlenkegel mit dem Polygon eingezeichnet. Die Strahlendichte wurde so eingestellt, dass der Ikosaeder-Generator die Geometrie mit insgesamt 20480 Strahlen abtastet. Diese Anzahl stellt ebenfalls die maximale Strahlendichte der adaptiven Generators im Bild 6.11b dar. Die minimale Strahlendichte wurde auf den niedrigsten möglichen Wert eingestellt. Dieser liegt bei dem Ikosaeder-Generator bei 20 Strahlen. Die Adaption der Strahlendichte an die Geometrie ist klar ersichtlich. Gleichzeitig wird die lückenlose Abdeckung des Raumes deutlich. Durch den Einsatz des adaptiven Generators konnte die Anzahl der generierten Strahlen von 20480 auf 1044 reduziert werden.

Ein anderes Beispiel ist im Bild 6.12 dargestellt. In diesem Fall ist auch die Projektion der Strahlendichte auf die Einheitskugel des Senders dargestellt. Die Lage der Kanten ist aus dieser Darstellung klar erkennbar.

## 6.6 Berechnung der Beugung

Die Beugung an Kanten wird ebenso wie die Streuung an rauen Oberflächen durch den Einsatz von sekundären Quellen realisiert, die nach einer Detektion des Effektes als neue Sender mit besonderen Eigenschaften in das Szenario eingebaut werden. Dadurch kann



(a) Beispielgeometrie

(b) Verteilungsdichte der Strahlen

**Bild 6.12:** Die Strahlendichteverteilung eines deterministischen adaptiven Strahlengenerators an einem Beispiel. Die variable Strahlendichte entspricht der Projektion der Abtastpunkte der Geometrie auf die Einheitskugel des Senders.

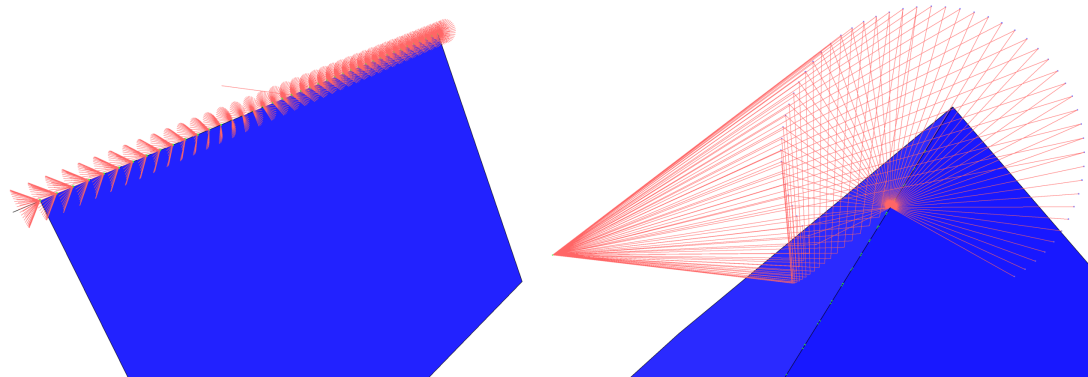
für die Berechnung beider Effekte die gleiche Cone Tracing Funktion eingesetzt werden, wie sie schon für die Berechnung der Reflexion und der Transmission verwendet wurde. Der Aufwand für die Implementierung und das Testen der Funktionalität wird dadurch deutlich geringer. Bei der Kombination der Beugung mit der Spiegelungsmethode müssen jedoch besondere Maßnahmen getroffen werden, damit die richtige Position der Beugung an der Kante gefunden wird.

Die Voraussetzung für die Installation der Sekundärquellen ist eine Detektion der Kanten, an den eine Beugung stattfinden kann. Die Suche nach den Kanten, die für eine Beugung in Frage kommen, kann im Vorfeld der Simulation mit Hilfe des topologischen Binärbaumes realisiert werden. Gleichzeitig kann die Kantengeometrie analysiert und die notwendigen Größen und Koordinatensysteme für eine spätere Verwendung im Baum abgelegt werden.

Einen anderen Weg bildet die Detektion der Kanten zur Laufzeit, die bereits von dem adaptiven Strahlengenerator zur Anpassung der Strahlendichte durchgeführt wird. Wie im Abschnitt 6.5 erwähnt, kann diese Kantendetektion mit einer Klassifizierung der Kante bezüglich der Beugung kombiniert werden. Eine anschließende Analyse der Kantengeometrie muss dann ebenfalls zur Laufzeit erfolgen. Wird die Analyse nicht mit Hilfe des topologischen BSP Baumes realisiert, so müssen alle an die Kante anliegenden Polygone auf eine andere Art und Weise gefunden werden. Diese Suche kann ebenfalls sehr schnell mit dem BSP Baum erfolgen. Dazu muss die Zugehörigkeit der Kante zu anderen Polygonen und deren Kanten abgefragt werden. Das Verfahren arbeitet wie folgt:

Der BSP Baum wird rekursiv mit einem Punkt der Kante durchquert. Es wird geprüft, ob sich der Punkt in der Partitionierungsebene des aktuellen Knotens befindet. Ist dies der Fall, so wird der Punkt mit allen Polygonen in dieser Ebene auf Zugehörigkeit geprüft und die positiven Ergebnisse in eine Liste eingetragen. Anschließend muss der Test mit beiden Teilbäumen des aktuellen Knotens durchgeführt werden, da beide Teilbäume relevante Polygone enthalten könnten. Falls der Punkt nicht in der Ebene liegt, so wird die Suche in dem Teilbaum fortgesetzt, der den Punkt enthält. Nach dem Durchlauf enthält die Liste alle Polygone, die den Punkt enthalten. Da diese Liste in der Regel nur einige wenige Polygone enthält, kann die Kantengeometrie schnell bestimmt werden.

Die Beugungsquellen werden, wie im Bild 6.13a dargestellt, auf der Kante platziert. Jede Beugungsquelle erzeugt einen Fächer von Strahlen unter dem entsprechenden Winkel zur Kante. Die Strahlen werden mit der Cone Tracing Methode verfolgt und können, soweit nicht anders gewünscht, weitere Beugungsquellen erzeugen. Erst nachdem ein Empfänger von einem Strahl über einen noch nicht bekannten Ausbreitungsweg getroffen wurde, und nach einer anschließenden Spiegelung und Überprüfung des neuen Ausbreitungspfades, wird mit der Auswertung der elektrischen Eigenschaften des Pfades begonnen. Dazu wird neben der Berechnung der elektrischen Parameter der Wände auch die Funktion zur Berechnung der Amplituden des von der Quelle abgestrahlten Feldes aufgerufen. Diese Funktion der Beugungsquelle initiiert ihrerseits die Auswertung der elektrischen Parameter der einfallenden Welle. Die Auswertung erfolgt in der Art einer Kettenreaktion bis zu der ersten Quelle des Feldes. Im Bild 6.13b sind die möglichen Ausbreitungswege zwischen einem Sender und einer kreisförmigen Anordnung der Empfänger dargestellt. Die drei Bereiche des Feldes (Abschnitt 2.5) sind erkennbar. Die genaue Position der Beugungsquelle wurde mit Hilfe der, im nächsten Abschnitt beschriebenen, Spiegelungsmethode bestimmt. Auf die Beiträge der anderen Kanten wurde der Übersichtlichkeit wegen verzichtet.



(a) Position der Beugungsquellen an der Kante

(b) Ausbreitung mit Reflexion und Beugung

**Bild 6.13:** Position der Sekundärquellen an der Beugungskante. Der Beitrag des gebeugten Feldes zum Gesamtfeld wird mit den im Abschnitt 2.5 beschriebenen Koeffizienten berechnet.

## 6.7 Spiegelung der Beugungsquellen

Die Vorteile der Kombination der Cone Tracing Methode mit der Spiegelungsmethode wurden bereits im Abschnitt 6.7 behandelt. Die Bestimmung der Spiegelquellen ist besonders bei planaren Geometrien einfach und effizient durchzuführen, da durch die planaren Grenzflächen die Position der Spiegelquelle eindeutig festgelegt ist. Bei krummlinigen Oberflächen oder bei Strahlen, die an Kanten gebeugt wurden, ist die Bestimmung des genauen Ortes der Spiegelquelle nicht mehr eindeutig und hängt in einer komplizierten Art und Weise mit den anderen getroffenen Objekten zusammen.

Ziel dieses Abschnitts ist die Bestimmung der genauen Position der Punkte auf den Beugungskanten, an denen der Strahl die Kante treffen muss, damit der Länge des Strahls minimal wird. Das Verfahren soll nur für Strahlen mit einer geringen Anzahl der Beugungsereignisse verwendet werden, da der hohe Rechenaufwand aufgrund des geringen



Beitrags der mehrfach gebeugten Strahlen zum Gesamtfeld nicht gerechtfertigt erscheint. Ebenfalls ist bei der Lösung der Minimierungsaufgabe mit Konvergenzproblemen bei einer steigenden Zahl der Unbekannten zu rechnen.

Bei der Formulierung des Problems ist zu berücksichtigen, dass zwischen zwei Beugungsquellen beliebig viele Transmissionen oder Reflexionen an planaren Wänden liegen können. Da die Beugungskanten eine beliebige Lage zueinander besitzen können, wirkt sich eine Verschiebung des Beugungspunktes an einer Kante auf die Lage der Punkte an allen anderen Kanten aus. Wird die Lage des Beugungspunktes im lokalen Koordinatensystem der Kante ausgedrückt, so lässt sich das Problem durch die Formulierung eines nichtlinearen Gleichungssystems mit  $N$  unbekannt Positionen der Punkte beschreiben. Im Folgenden werden die Gleichungen des Systems hergeleitet und eine iterative Lösung des Gleichungssystems vorgestellt. Dazu wird ein vereinfachtes geometrisches Modell betrachtet, das nur aus Beugungskanten ohne Reflexionen zwischen den Kanten besteht. Dies stellt keine Einschränkung der Lösung ein, da eine Reflexion zwischen zwei Beugungskanten durch die Spiegelung der ihr folgenden Kanten ersetzt werden kann.

### 6.7.1 Formulierung des Fermatschen Prinzips

Nach dem Fermatschen Prinzip breiten sich elektromagnetische Wellen entlang des Pfades mit der kürzesten Laufzeit aus (FEYNMAN, 1970). Im freien Raum entspricht dies der kürzesten geometrischen Entfernung der beiden Beobachterstandorte. Das dieser Arbeit zugrundeliegende geometrische Modell geht stets von einer Ausbreitung im freien Raum aus, so dass die geometrische Interpretation des Fermatschen Prinzips gültig bleibt. In diesem Fall lässt sich das Fermatsche Prinzip schreiben als

$$d = \sum_i d_i \stackrel{!}{=} \min \quad (6.18)$$

mit  $d$  der gesamten Länge des Pfades und den Längen  $d_i$  der den Pfad bildenden Teilstrahlen. Die Längen der Teilstrahlen sind durch die Position von jeweils zwei aufeinanderfolgenden Beugungspunkten gegeben:

$$d_i = \left\| \vec{D}_i \right\| = \sqrt{\vec{D}_i \cdot \vec{D}_i} \quad , \quad (6.19)$$

wobei der Vektor  $\vec{D}_i$  den Beugungspunkt  $i - 1$  mit dem darauf folgenden Beugungspunkt  $i$  verbindet. Der Differenzvektor lässt sich als eine lineare Funktion der Position der Beugungspunkte im lokalen Koordinatensystem ( $\vec{P}_i, \vec{E}_i$ ) der Kanten angeben:

$$\vec{D}_i = \begin{cases} \vec{P}_i + l_i \vec{E}_i - \vec{T} & \text{für } i = 1 \\ \vec{P}_i + l_i \vec{E}_i - \vec{P}_{i-1} - l_i \vec{E}_{i-1} & \text{für } 1 < i \leq N \\ \vec{R} - \vec{P}_{i-1} - l_i \vec{E}_{i-1} & \text{für } i = N + 1 \end{cases} \quad . \quad (6.20)$$

Die Vektoren  $\vec{T}$  und  $\vec{R}$  bezeichnen die Positionen des Senders und des Empfängers. Die Lage der Kanten im Raum wird durch einen Punkt  $\vec{P}_i$  auf der Kante und den Richtungsvektor der Kante  $\vec{E}_i$  definiert. Gesucht sind die unbekannt Positionen  $l_i$  der Beugungspunkte im lokalen Koordinatensystem der Kanten. Diese Unbekannt können zu dem unbekannt Vektor  $\vec{l}$  zusammengefasst werden.

Die Suche nach dem Minimum von  $d(\vec{l})$  entspricht der Suche nach den Nullstellen der ersten Ableitung von  $d(\vec{l})$  nach  $l_i$ , für die die zweite Ableitung der Funktion positiv wird:

$$\frac{\partial}{\partial l_i} d(\vec{l}) = f_i(\vec{l}) = 0 \quad \text{und} \quad \frac{\partial^2}{\partial l_i^2} d(\vec{l}) > 0 \quad \text{für } i = 1 \dots N \quad . \quad (6.21)$$

Die erste partielle Ableitung der Funktion  $d(\vec{l})$  kann einfach aus den Gleichungen (6.19) und (6.20) hergeleitet werden und ist in ihrer allgemeinen Form gegeben durch

$$\begin{aligned} f_i(l_{i-1}, l_i, l_{i+1}) &= \frac{\partial}{\partial l_i} d_i(l_{i-1}, l_i) + \frac{\partial}{\partial l_i} d_{i+1}(l_i, l_{i+1}) \\ &= \frac{\vec{E}_i \cdot \vec{D}_i(l_{i-1}, l_i)}{d_i(l_{i-1}, l_i)} - \frac{\vec{E}_i \cdot \vec{D}_{i+1}(l_i, l_{i+1})}{d_{i+1}(l_i, l_{i+1})} = 0 . \end{aligned} \quad (6.22)$$

Die erste ( $i = 1$ ) und der letzte ( $i = N$ ) Kante muss gesondert behandelt werden, da die Positionen des Senders und des Empfängers fest vorgegeben sind. Diese Positionen ersetzen in den Gleichungen die variable Position des Beugungspunktes auf der vorhergehenden bzw. der nachfolgenden Kante. Die Gleichung (6.22) ist eine andere Form der bekannten Beugungsbedingung  $\beta_i = \beta_o$ . Diese drückt aus, dass der Winkel  $\beta_i$  des einfallenden Feldes mit der Kante gleich sein muss dem Winkel  $\beta_o$ , den die abgehenden Strahlen mit der Kante bilden. Die Lösungsmenge aller Strahlen, die diese Bedingung erfüllen, bildet den Beugungskegel mit dem Öffnungswinkel  $\beta_o$ .

Die Anwendung der Gleichung (6.22) auf ein System mit  $N = 3$  Beugungskanten ergibt das folgende nichtlineare Gleichungssystem:

$$\begin{aligned} f_1(l_1, l_2) &= \frac{\partial}{\partial l_1} d_1(l_1) + \frac{\partial}{\partial l_1} d_2(l_1, l_2) = \frac{\vec{E}_1 \cdot \vec{D}_1(l_1)}{d_1(l_1)} - \frac{\vec{E}_1 \cdot \vec{D}_2(l_1, l_2)}{d_2(l_1, l_2)} = 0 \\ f_2(l_1, l_2, l_3) &= \frac{\partial}{\partial l_2} d_2(l_1, l_2) + \frac{\partial}{\partial l_2} d_3(l_2, l_3) = \frac{\vec{E}_2 \cdot \vec{D}_2(l_1, l_2)}{d_2(l_1, l_2)} - \frac{\vec{E}_2 \cdot \vec{D}_3(l_2, l_3)}{d_3(l_2, l_3)} = 0 \\ f_3(l_2, l_3) &= \frac{\partial}{\partial l_3} d_3(l_2, l_3) + \frac{\partial}{\partial l_3} d_4(l_3) = \frac{\vec{E}_3 \cdot \vec{D}_3(l_2, l_3)}{d_3(l_2, l_3)} - \frac{\vec{E}_3 \cdot \vec{D}_4(l_3)}{d_4(l_3)} = 0 . \end{aligned} \quad (6.23)$$

Im Folgenden wird ein iteratives Verfahren zur Lösung des nichtlinearen Gleichungssystems untersucht.

### 6.7.2 Die Newton-Raphson Iteration

Zur Lösung des nichtlinearen Gleichungssystems (6.22) kann eine Newton-Raphson Iteration (MARTINEZ, 1994; PRESS et al., 1988; SEBAH und GOURDON, 2001) über einem  $N$ -dimensionalen Parameterraum eingesetzt werden. Das Ergebnis der Iteration ist der  $N$ -dimensionale Lösungsvektor  $\vec{l}$ , der die Positionen der Beugungspunkte auf den Kanten angibt. Um auch eine gültige Lösung des Problems darzustellen, muss dieser Vektor ebenfalls die Nebenbedingung der Gleichung (6.21) erfüllen.

Die Newton-Iteration startet mit einem geeignet gewählten Startvektor  $\vec{l}_0$ . Die Lösung ergibt sich durch die Anwendung der Iterationsvorschrift

$$\vec{l}_{n+1} = \vec{l}_n - \mathbf{J}^{-1} \vec{f}(\vec{l}_n) , \quad (6.24)$$

in der der Vektor  $\vec{l}_n$  in den Lösungsvektor  $\vec{l}_{n+1}$  überführt wird.  $\mathbf{J}$  ist die Jacobi-Matrix, deren Elemente durch die  $N^2$  möglichen Partialableitungen der Funktion  $f_i$  nach  $l_j$  für  $i, j = 1 \dots N$  gegeben sind:

$$\mathbf{J} = \begin{pmatrix} \frac{\partial f_1}{\partial l_1} & \frac{\partial f_1}{\partial l_2} & \cdots & \frac{\partial f_1}{\partial l_n} \\ \frac{\partial f_2}{\partial l_1} & \frac{\partial f_2}{\partial l_2} & \cdots & \frac{\partial f_2}{\partial l_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial l_1} & \cdots & \cdots & \frac{\partial f_n}{\partial l_n} \end{pmatrix} . \quad (6.25)$$



Die Jacobi-Matrix des Problems ist eine tridiagonale, symmetrische, positiv definite  $N \times N$  Matrix der Partialableitungen der Funktion  $f_i$  aus der Gleichung (6.22). Die Matrix kann mit Hilfe der klassischen Gauss-Elimination invertiert werden.

$$\mathbf{J} = \begin{pmatrix} J_{1,1} & J_{1,2} & 0 & 0 & 0 & \dots & 0 \\ J_{2,1} & J_{2,2} & J_{2,3} & 0 & 0 & \dots & 0 \\ 0 & J_{3,2} & J_{3,3} & J_{3,4} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & J_{N-1,N} & J_{N,N} \end{pmatrix}. \quad (6.26)$$

Die Elemente auf der Hauptdiagonale der Matrix  $\mathbf{J}$  sind durch die Partialableitungen für  $i = j = 1 \dots N$  gegeben und haben die folgende Gestalt:

$$\begin{aligned} \mathbf{J}_{i,i} = \frac{\partial}{\partial l_i} f_i(l_{i-1}, l_i, l_{i+1}) &= \frac{1}{d_i(l_{i-1}, l_i)} - \frac{\left(\vec{E}_i \cdot \vec{D}_i(l_{i-1}, l_i)\right)^2}{d_i(l_{i-1}, l_i)^3} \\ &+ \frac{1}{d_{i+1}(l_i, l_{i+1})} - \frac{\left(\vec{E}_i \cdot \vec{D}_{i+1}(l_i, l_{i+1})\right)^2}{d_{i+1}(l_i, l_{i+1})^3}. \end{aligned} \quad (6.27)$$

Die Elemente auf den Nebendiagonalen sind durch die Bedingungen  $j = i - 1$

$$\begin{aligned} \mathbf{J}_{i,i-1} = \frac{\partial}{\partial l_{i-1}} f_i(l_{i-1}, l_i, l_{i+1}) &= \frac{\left(\vec{E}_{i-1} \cdot \vec{D}_i(l_{i-1}, l_i)\right) \left(\vec{E}_i \cdot \vec{D}_i(l_{i-1}, l_i)\right)}{d_i(l_{i-1}, l_i)^3} \\ &- \frac{\vec{E}_{i-1} \cdot \vec{E}_i}{d_i(l_{i-1}, l_i)}. \end{aligned} \quad (6.28)$$

und  $j = i + 1$

$$\begin{aligned} \mathbf{J}_{i,i+1} = \frac{\partial}{\partial l_{i+1}} f_i(l_{i-1}, l_i, l_{i+1}) &= \frac{\left(\vec{E}_i \cdot \vec{D}_{i+1}(l_i, l_{i+1})\right) \left(\vec{E}_{i+1} \cdot \vec{D}_{i+1}(l_i, l_{i+1})\right)}{d_{i+1}(l_i, l_{i+1})^3} \\ &- \frac{\vec{E}_i \cdot \vec{E}_{i+1}}{d_{i+1}(l_i, l_{i+1})}. \end{aligned} \quad (6.29)$$

definiert. Durch die Erhöhung des Index  $i$  in der Gleichung (6.28) um eins wird die Symmetrie der Matrix sichtbar.

$$\mathbf{J}_{i,j} = \mathbf{J}_{j,i}. \quad (6.30)$$

Die symmetrische tridiagonale Form der Matrix kann vorteilhaft bei einer effizienten Implementierung der Inversion ausgenutzt werden.

### 6.7.3 Präkonditionierung

Eine geeignete Präkonditionierung des Lösungsvektors  $\vec{l}_0$  ist für eine schnelle Konvergenz der Newton-Iteration wichtig. Durch eine Präkonditionierung wird versucht, den Startvektor  $\vec{l}_0$  in die Nähe des möglichen Lösungsvektors zu positionieren.

Die initialen Positionen der Beugungsquellen  $\vec{P}_i$ , die während der Cone Tracing Phase gefunden wurden, können als eine erste Näherung der richtigen Position der tatsächlichen Beugungsquelle auf der Kante angesehen werden. Der einfachste Fall einer Präkonditionierung besteht deshalb darin, den Startvektor  $\vec{l}_0$  zu Null zu setzen. Eine sehr viel

bessere Prakonditionierung wird durch die Losung der folgenden Naherungsgleichung fur das Quadrat der Lange  $d$  erreicht:

$$d^2 = \left( \sum_i d_i \right)^2 \approx \sum_i d_i^2 = \sum_i \vec{D}_i \cdot \vec{D}_i \stackrel{!}{=} \min . \quad (6.31)$$

In diesem Fall ist die Berechnung der partiellen Ableitungen besonders einfach und fuhrt zu einem System von  $N$  linearen Gleichungen  $\mathbf{A}\vec{l}_0 = \vec{y}$ , das auf eine klassische Art und Weise nach dem Startvektor  $\vec{l}_0$  aufgelost werden kann. Die Systemmatrix  $\mathbf{A}$  hat, ahnlich wie die Jacobi-Matrix (6.26), auch eine symmetrische tridiagonale Form. Ihre Elemente ergeben sich zu:

$$\frac{\partial}{\partial l_j} d^2 \approx \mathbf{A}_{i,j} = 2\vec{D}_i \cdot \frac{\partial}{\partial l_j} \vec{D}_i + 2\vec{D}_{i+1} \cdot \frac{\partial}{\partial l_j} \vec{D}_{i+1} = 0 . \quad (6.32)$$

Explizit geschrieben sind die Elemente durch den folgenden Ausdruck gegeben:

$$\mathbf{A}_{i,j} = \begin{cases} 2 & \text{fur } i = j \\ -\vec{E}_i \cdot \vec{E}_j & \text{fur } i = j \pm 1 \\ 0 & \text{sonst} \end{cases} . \quad (6.33)$$

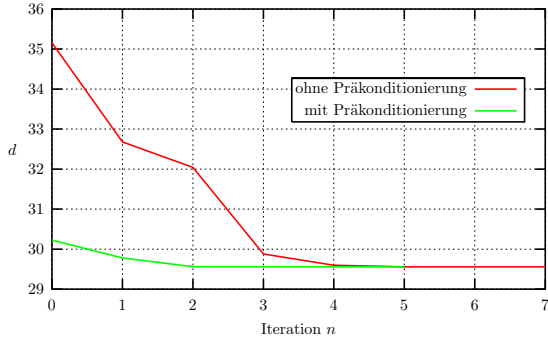
Desweiteren ergibt die zweite Ableitung der vereinfachten Gleichungen (6.31) den Wert 2. Somit stellt der Startvektor das Minimum der Naherung fur  $d^2$  dar. Es wird angenommen, dass der auf diese Weise berechnete Startvektor sich ebenfalls in der Naherung des Minimums von  $d$  befindet.

Die Auswirkung der Prakonditionierung auf die Konvergenz des Verfahrens kann dem Bild 6.14 entnommen werden. Wahrend sich bei der Wahl des Startvektors nach Gleichung (6.32) schnell eine Konvergenz der Iteration einstellt, kann bei einer fehlenden Prakonditionierung das Verfahren instabil werden. Fur eine zunehmende Zahl der Beugungskanten steigt die Zahl der lokalen Minima an und das einfache Newton-Verfahren kann nicht mehr eingesetzt werden. In einem solchen Fall muss auf andere Methoden zur Losung nichtlinearer Optimierungsprobleme bei der Bestimmung des kurzesten Pfades zuruckgegriffen werden. Eine alternative Losung mit der *very fast simulated re-annealing* Methode (INGBER, 1989) wird im folgenden Abschnitt vorgestellt. Die Iteration wird abgebrochen, wenn die anderung der Lange  $d$  des Pfades eine Toleranzschwelle unterschreitet oder wenn eine maximale Anzahl der Iterationsschritte uberschritten wird. Der Losungsvektor muss anschlieend noch mit den physikalischen Abmessungen der Kanten verglichen werden, da durch die Verschiebung der Beugungspunkte entlang der Kantenlinien einige Punkte den Bereich der Kante verlassen konnten.

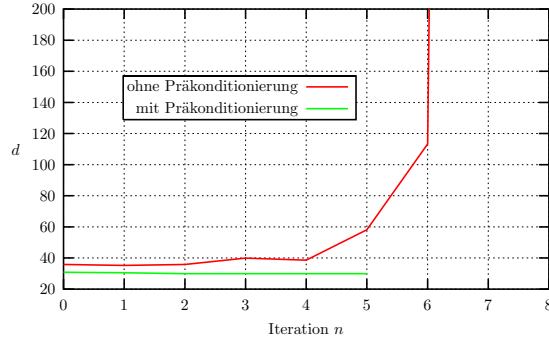
#### 6.7.4 Losung mit Simulated Annealing

Wie im letzten Abschnitt bereits erwahnt, kann die lineare Losung des Minimierungsproblems bei einer steigenden Anzahl der Beugungskanten nicht eingesetzt werden.

Die Gleichung (6.18) kann alternativ mit der Simulated Annealing Methode gelost werden. Die Positionen der Beugungspunkte auf den Kanten spannen den Parameterraum auf. Da die Kanten eine endliche Ausdehnung haben, ist der Raum der gultigen Parameter entsprechend begrenzt. Fur die Losung des Pfadintegrals wurde die adaptive Simulated Annealing Methode (ASA) nach INGBER (1995) eingesetzt. Das Ergebnis der Berechnung fur vier zufallig orientierte Kanten ist im Bild 6.15a dargestellt. Das Bild zeigt den Verlauf



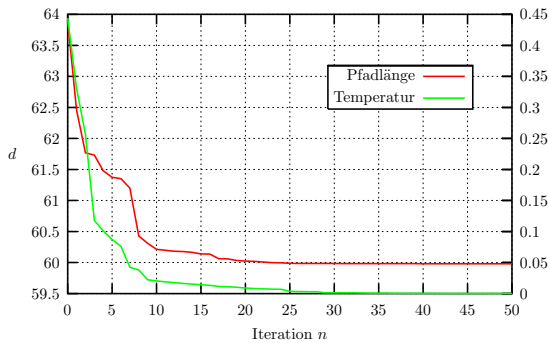
(a) Vergleich der Konvergenzraten



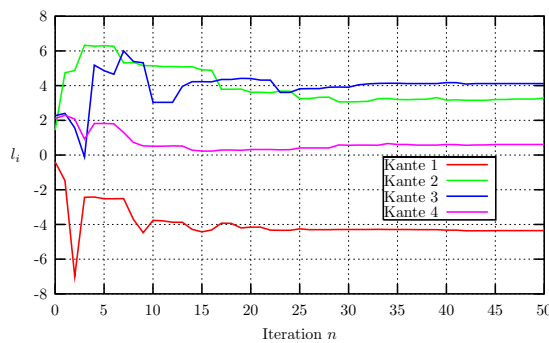
(b) Konvergenzproblem

**Bild 6.14:** Das Konvergenzverhalten der iterativen Lösung des Spiegelungsproblems. Eine Prädiktionierung des Lösungsvektors sorgt für eine schnellere Konvergenz und Stabilität der Iteration für eine niedrige Anzahl der Beugungskanten (hier 2).

der Temperatur und der Gesamtlänge des Pfades als Funktion der Anzahl der Annealing-Schritte. Die Verschiebung der Beugungspunkte auf den Kanten ist im Bild 6.15b zu sehen. Mit der klassischen Newton-Raphson Iteration konnte das Minimum für diese Anordnung der Kanten aufgrund der fehlenden Konvergenz nicht bestimmt werden.



(a) Konvergenz und Temperatur



(b) Konvergenz der Systemparameter

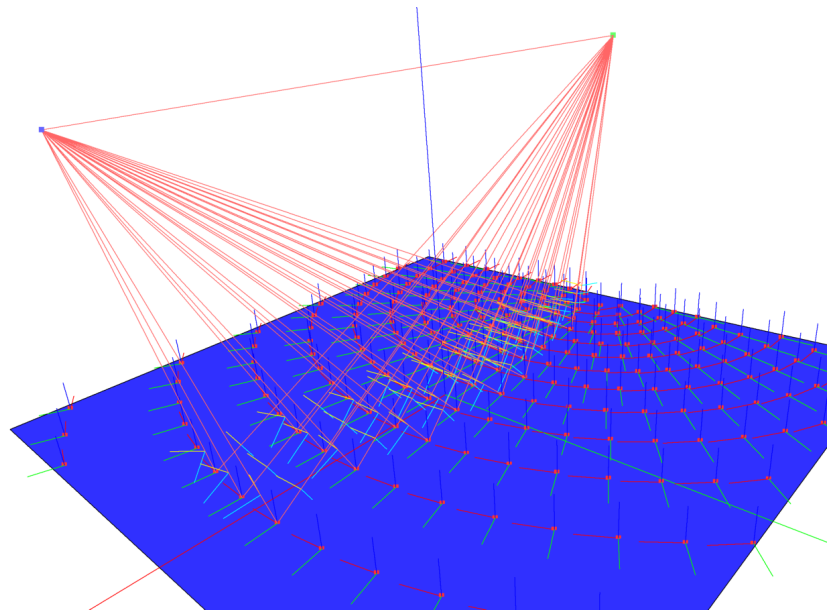
**Bild 6.15:** Lösung der Gleichung (6.18) für 4 Kanten mit der adaptiven Simulated Annealing Methode nach Ingber (1989).

## 6.8 Berechnung der Streuung

Die Streuung an rauen Oberflächen wird ähnlich wie die Beugung durch den Einsatz von Sekundärquellen modelliert. Aus Gründen der Effizienz wurde auf die Implementierung einer zweidimensionalen Berechnung der bistatischen Streukoeffizienten verzichtet. Stattdessen wurde das eindimensionale Modell nach Tsang et al. (2001) eingesetzt, das im Kapitel 2 Abschnitt 2.6 behandelt wurde. Nach diesem Modell wird an jedem Ort, an dem ein Strahl die raue Fläche trifft, eine eindimensionale Strahlenquelle nach Bild 2.8

auf Seite 20 installiert. Die von dieser Quelle in der Einfallsebene ausgesandten Strahlen können aufgrund ihrer konischen Form einen Empfänger in der Nähe der Einfallsebene treffen. Durch eine anschließende Spiegelung des gestreuten Strahls wird jedoch nicht die Position der Sekundärquelle auf der Fläche verändert, wie das bei einer spiegelnden Reflexion der Fall gewesen wäre, sondern nur die Abschussrichtung des Strahls bei der Sekundärquelle. Somit können Sekundärstrahlen aus der Einfallsebene ausgelenkt werden und durch die Vernachlässigung der Azimut-Abhängigkeit der Streukoeffizienten einen modellbedingten Fehler verursachen. Eine genaue Berechnung der Streuung würde jedoch den Einsatz eines aufwendigen zweidimensionalen Modells der Streukoeffizienten erforderlich machen.

Dieses Modell der Streuung führt zu der im Bild 6.16 dargestellten Verteilung der Streuzentren auf der gesamten Fläche des Polygons. Nur ein Teil der dargestellten Sekundärquellen, bestimmt vor allem durch den Öffnungswinkel der Strahlen, trägt jedoch zum Feld im Empfänger bei. Im Bild sind neben den gefundenen Strahlen ebenfalls die lokalen Koordinaten der Streuzentren sowie das globale Koordinatensystem eingezeichnet.



**Bild 6.16:** Realisierung der Streuung an rauen Oberflächen. Bei der Berechnung der Sekundärquellen wird die Spiegelungsmethode nicht angewendet. Die lokalen Koordinatensysteme der Strahlen sowie der Streuzentren sind mit eingezeichnet.

Bei der Anwendung des Modells der Streuung zusammen mit der adaptiven Strahledichtegenerierung ist zu beachten, dass die raue Oberfläche (bzw. das diese Oberfläche repräsentierende Polygon) mit der maximalen Strahledichte abgetastet werden muss. Dazu muss die Adaptionsbedingung in der Cone Tracing Funktion modifiziert werden. Die Modifikation kann jedoch sehr einfach realisiert werden, wenn die Funktion einen Zugriff auf die Information über die Oberflächenbeschaffenheit der Polygone erhält. Jedes Polygon wird um eine Variable mit dieser Information erweitert. Diese Variable enthält lediglich eine grundsätzliche Klassifizierung der Wandeigenschaften (transparent, reflektierend oder streuend), die ebenfalls zu einer Beschleunigung der Schnittpunktberechnung genutzt wird.

Dieses empirische Modell der Streuung stellt einen brauchbaren Kompromiss zwischen

der physikalischen Realität und des, für eine Nachbildung der Realität benötigten, Rechenaufwands dar.

## 6.9 Zusammenfassung

In diesem Kapitel wurde eine Kombination der Cone Tracing Methode mit der Spiegelungsmethode als Grundlage für einen adaptiven Ray Tracing Algorithmus vorgestellt. Es wurden stochastische und deterministische Strahlengeneratoren untersucht. Ihre Auswirkung auf eine effiziente Nutzung der Speicherhierarchie wurde gemessen und bewertet. An einem einfachen Beispiel konnte gezeigt werden, dass durch eine adaptive Strahlgenerierung auf der Basis der rekursiven Triangulierung des Ikosaeders ein erhebliches Einsparpotential bei der Gesamtzahl der, für die Abtastung benötigten, Strahlen besteht. Es wurden weiterhin die Grundzüge der Implementierung des Beugungs- und Streumodells behandelt. Dazu gehörte eine Methode der Beugungskantendetektion, die auf einer schnellen Schnittpunktbestimmung der Kanten und des Strahlkegels mit dem BSP Baum beruht und vorteilhaft mit dem Verfahren zur adaptiven Anpassung der Strahldichte kombiniert werden kann. Desweiteren wurden Methoden zur genauen Positionsbestimmung der Beugungsquellen auf den Kanten mit Hilfe der Spiegelungsmethode diskutiert.

---

## Paralleles Ray Tracing

Durch eine Parallelisierung des im letzten Kapitel vorgestellten Algorithmus kann der Anwendungsbereich der Methode auf größere Probleme erweitert werden. Diese Erweiterung wird sowohl bei der Größe des Szenarios als auch bei der maximalen Strahledichte erreicht. Eine Voraussetzung für die Parallelisierung ist die Aufteilung des Problems in voneinander unabhängige Teilprobleme, die parallel gelöst werden können. Diese Bedingung ist bei einem Ray Tracing Verfahren erfüllt, da die Strahlen unabhängig voneinander berechnet werden können. Somit stellt die Verfolgung eines Strahls die kleinstmögliche *Work Unit* (WU) dar. Durch die sehr große Strahlenmenge besitzt das Problem eine sehr feine Granularität, die eine effiziente Parallelverarbeitung, selbst auf sehr vielen Prozessoren, ermöglicht. Daraus resultiert das gute Skalierungsverhalten von Ray Tracing Verfahren.

In diesem Kapitel wird die Parallelisierung des Verfahrens für verschiedene parallele Rechnerarchitekturen behandelt. Nach einer kurzen Beschreibung der Besonderheiten der Architekturen werden an diese Architekturen angepasste Algorithmen entwickelt. Die Messungen der erreichten Beschleunigung werden mit der theoretischen Grenze, die durch das Amdahlsche Gesetz gegeben ist, verglichen. Die Auswirkung der Eigenschaften des Kommunikationsnetzwerks auf diese Grenze und auf die gemessenen Werte der Beschleunigung wird ebenfalls untersucht.

### 7.1 Parallele Rechnerarchitekturen

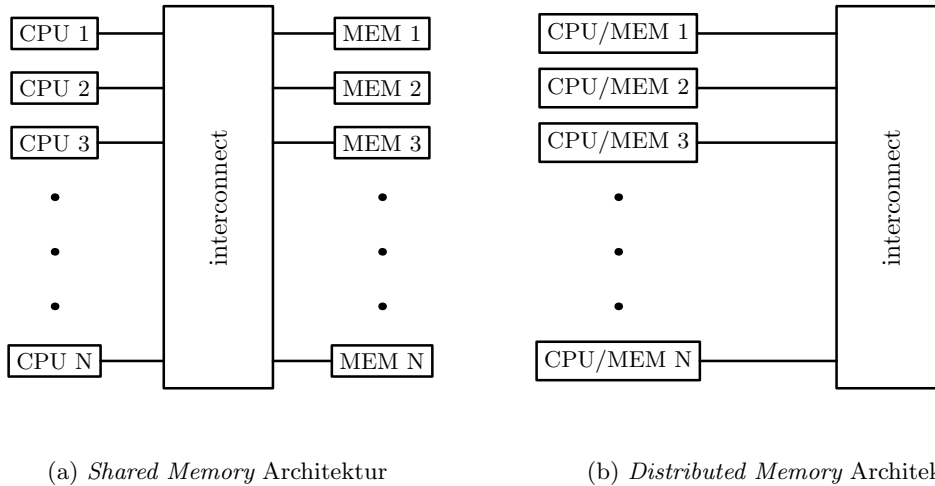
Parallele Rechnerarchitekturen lassen sich anhand der Verschaltung der Prozessoreinheiten und der Speichereinheiten durch ein Kommunikationsnetzwerk in zwei Klassen einteilen<sup>1</sup>. Die schematische Darstellung beider Architekturen ist im Bild 7.1 zu sehen. Aufgrund der unterschiedlichen Anordnung der Prozessoren und der Speicher relativ zu dem Kommunikationsnetzwerk ergeben sich deutlich verschiedene Anforderungen an das Kommunikationsnetzwerk sowie unterschiedliche Paradigmen der Entwicklung paralleler Programme. Beide Architekturen werden im Folgenden genauer betrachtet.

#### 7.1.1 Parallelrechner mit einem gemeinsamen Speicher

Parallelrechner mit der sogenannten *Shared Memory* Architektur bestehen aus Prozessoreinheiten, die über das Kommunikationsnetzwerk (*Interconnect*) auf den gemeinsamen Speicher zugreifen. Die Anzahl der Speichereinheiten kann dabei von der Anzahl der Prozessoreinheiten unterschiedlich sein. Durch die Lage des Kommunikationsnetzwerkes zwischen dem Prozessor und dem Speicher wird erreicht, dass jeder Prozessor auf alle Daten zugreifen und diese modifizieren kann. Dadurch wird die Programmierung dieser Rechnerklasse unter der Beachtung der erforderlichen Synchronisation der parallelen Funktionen (*Threads*) besonders einfach. Andererseits steigt jedoch der Aufwand für das

---

<sup>1</sup>Diese Einteilung bezieht sich vor allem auf universal einsetzbare Parallelrechner. Andere spezialisierte Architekturen wie z.B. *Systolic Arrays* oder *Single-Instruction Multiple-Data* (SIMD) Vektorprozessoren werden hier nicht betrachtet.



**Bild 7.1:** Vergleich der *Shared Memory* Architektur mit der *Distributed Memory* Architektur. Durch die unterschiedliche Anordnung der Prozessoren und der Speicher ergeben sich wesentliche Unterschiede bei der Programmierung.

Kommunikationsnetzwerk mit steigender Anzahl der Prozessoren stark an, da die Realisierung eines schnellen Kommunikationsnetzwerks entscheidend für die Speicherbandbreite und somit für eine gute Ausnutzung der zur Verfügung stehenden Prozessorleistung ist. Alle Prozessoreinheiten verfügen zudem über eigene Cache-Hierarchien, die unter der Ausnutzung lokaler Datenabhängigkeiten die Zahl der Zugriffe auf den gemeinsamen Speicher reduzieren.

Die am häufigsten anzutreffende Klasse eines *Shared Memory* Rechners stellen die *Symmetric Multi-Processor* (SMP) Systeme dar. Diese Rechnerklasse zeichnet sich dadurch aus, dass alle Prozessorelemente in gleicher Weise auf den gemeinsamen Speicher zugreifen können, und wird deshalb auch mit dem Begriff *Uniform Memory Access* (UMA) bezeichnet. Das Kommunikationsnetzwerk wird in der Regel im High-End Bereich als ein *Crossbar Switch* realisiert. Diese Art des Kommunikationsnetzwerks ist sehr schnell, da jedem Prozessor eine konstante Speicherbandbreite zur Verfügung gestellt wird. Unter der Voraussetzung, dass die Zahl der Prozessoreinheiten und der Speichereinheiten gleich ist, steigt jedoch der Realisierungsaufwand eines solchen Kommunikationsnetzwerkes quadratisch mit der Zahl der Prozessoren an und ist deswegen mit extrem hohen Kosten verbunden. Im Low-Cost Bereich werden häufig einfache Bussysteme eingesetzt, deren Realisierungsaufwand weitgehend unabhängig von der Anzahl der Prozessoren ist, die jedoch jedem Prozessor bestenfalls nur  $1/N$  der Speicherbandbreite bereitstellen<sup>2</sup>. In beiden Fällen limitieren entweder technologische/finanzielle Grenzen oder die schlechte Ausnutzung der zur Verfügung stehenden Prozessorleistung die Skalierbarkeit der SMP Architektur. Eine Übersicht der Eigenschaften einiger kommerziell verfügbaren UMA Systeme ist in der Tabelle 7.1 angegeben.

Vor allem die einfache Parallelisierung vorhandener Software auf Rechnern mit einem gemeinsamen Speicher begründete den Erfolg dieser Rechnerklasse. Es ist daher nicht verwunderlich, dass Hersteller an dem *Shared Memory* Paradigma festhalten und eine wei-

<sup>2</sup>Ausgestattet mit einer effizienten Cache-Hierarchie stellen Bussysteme bei kleinen Prozessorzahlen eine sehr kostengünstige Alternative zu den teuren Crossbar Switch-basierten Systemen.



System	Prozessor	Max. CPUs	Speicherbandbreite
SGI Power Challenge	MIPS-R10000	36	1.2 GB/sec (bus)
DEC 8400	Alpha-21164	14	1.8 GB/sec (bus)
Sun E6000	UltraSparc-2	30	2.5 GB/sec (bus)
Sun E10000	UltraSparc-2/3	64	13 GB/sec (crossbar)
HP Exemplar	PA-8000	16	15 GB/sec (crossbar)
Cray T90	Cray Vector	32	800 GB/sec (crossbar)

**Tabelle 7.1:** Daten einiger SMP Systeme (aus [Dowd und Severance \(1998\)](#)).

tere Steigerung der Leistung durch eine steigende Zahl der Prozessoren erreichen wollen. Da dem SMP Modell in der Realität technologische und wirtschaftliche Grenzen gesetzt sind, konnte eine weitere Leistungssteigerung nur durch den Verzicht auf die wichtige Eigenschaft des gleichmäßigen Speicherzugriffs aller Prozessoren erreicht werden. Die *Non-Uniform Memory Access* (NUMA) Architektur besteht aus Rechenknoten, die über einen schnellen lokalen Speicher verfügen und über ein Kommunikationsnetzwerk auf einen den globalen Speicher zugreifen können. Durch diese Architektur wird die Geschwindigkeit des Speicherzugriffs abhängig von der Lage der Daten im Speicher, die wichtige Eigenschaft des Zugriffs auf einen gemeinsamen Speicher bleibt jedoch erhalten. In heutigen Systemen bestehen die Rechenknoten selbst aus mehreren Prozessoren, die eine lokale SMP Einheit bilden. Bei der Programmierung dieser Rechnerklasse muss besonders auf die Position der Daten im Speicher des Rechners und eine gute Ausnutzung der Cache-Hierarchie durch Datenlokalität geachtet werden. Durch eine aufwendige Synchronisation der Cache-Speicher der Prozessoren, die auf gemeinsame Daten zugreifen, wird die Programmierung der Architektur weiter vereinfacht. Solche Systeme werden mit dem Begriff ccNUMA Architektur bezeichnet. Die Eigenschaften einiger solcher Systeme sind in der Tabelle 7.2 aufgeführt. Obwohl sich ccNUMA Systeme, aus der Sicht des Programmierers,

System	Prozessor	CPU/Kn.	Max. CPUs	Bandbreite
Data General AV-20000	Pentium	4	32	1 GB/sec
Sequent NUMAQ-2000	Pentium	4	32	1 GB/sec
SGI Origin-2000	MIPS-R10000	2	128	80 GB/sec
HP Exemplar X-Class	PA-8000	16	256	246 GB/sec

**Tabelle 7.2:** Daten einiger ccNUMA Systeme (aus [Dowd und Severance \(1998\)](#)).

abgesehen von den unterschiedlichen Zugriffsgeschwindigkeiten auf den Speicher, gleich wie ein SMP System verhalten, ist ihre Architektur eher mit der von Rechnern mit einem verteilten Speicher vergleichbar. In der Tat ist es durch eine entsprechende Emulation auf Betriebssystem-Ebene möglich, einen *Distributed Memory* Parallelrechner für den Benutzer wie eine ccNUMA Architektur aussehen zu lassen (*Virtual Shared Memory*).

### 7.1.2 Parallelrechner mit einem verteilten Speicher

Während bei einem *Shared Memory* Parallelrechner jeder Prozessor transparent auf den gesamten Speicher des Rechners zugreifen kann, besteht die *Distributed Memory* Architektur aus einer Vielzahl autonomer Rechenknoten mit einem lokalen Speicher, die über

ein Kommunikationsnetzwerk miteinander verbunden sind und Daten austauschen können. Die Rechenknoten werden für sich unabhängige Einprozessor- oder SMP Mehrprozessorsysteme eingesetzt. Diese Rechnerklasse wird nach der Art und der Leistung des Kommunikationsnetzwerks weiter unterteilt. Der prinzipielle Unterschied zu den *Shared Memory* Parallelrechnern liegt für den Programmierer darin, dass das Programm mit Hilfe entsprechender Werkzeuge selbst für den Austausch der benötigten Daten zwischen den Rechenknoten sorgen muss. Dieses *Message Passing* Paradigma setzt eine andere Denkweise bei der Entwicklung paralleler Programme voraus und führt dazu, dass bereits vorhandene Software sehr schwierig auf ein *Distributed Memory* System zu portieren ist und in den meisten Fällen neu geschrieben werden muss. Der große Vorteil dieser Rechnerarchitektur besteht darin, dass mit ihr eine wesentlich größere Anzahl Prozessoren zu einem Parallelrechner verschaltet und zur Lösung der Aufgabe eingesetzt werden kann. Für den Datenaustausch sowie die Synchronisation der parallel laufenden Prozesse stehen Werkzeuge zur Verfügung wie die ältere *Parallel Virtual Machine* (PVM) Umgebung oder der aktuelle Standard in der Form der *Message Passing Interface* (MPI) Bibliothek.

Als Beispiel für den Aufbau eines preiswerten Parallelrechners sei das Aufkommen der sog. *Beowulf Cluster* genannt (BECKER et al., 1995). Bei diesen Systemen bestehen die Rechenknoten aus gewöhnlichen PC's, die über ein hierarchisch aufgebautes sternförmiges Kommunikationsnetzwerk, meistens auf der Basis der preiswerten Ethernet Komponenten, miteinander kommunizieren. Nicht zuletzt durch das freie Betriebssystem *Linux* konnten so hochoptimierte leistungsfähige Parallelrechner aufgebaut werden, die sich in der TOP500 Weltrangliste der leistungsfähigsten Parallelrechner (MEUER et al., 2003) im vorderen Feld platzieren konnten. Systeme dieser Bauart bestehen in der Regel aus mehreren Hundert bis mehreren Tausend Rechenknoten mit jeweils mehreren Prozessoren. Die Daten einiger Rechner mit der *Distributed Memory* Architektur sind in der folgenden Tabelle 7.3 aufgetragen. Es bleibt anzumerken, dass die zur Zeit leistungsfähigsten Rechner die Merkmale

System	Prozessor	CPUs	Knoten	Peak	Speicher
Earth Simulator	NEC SX-5	5120	640	41 TFlop/s	10 TB
ASCI White	IBM RS/6000	8192	608	12.3 TFlop/s	6 TB
MRC Linux Cluster	Intel Xeon	1152	1152	11.2 TFlop/s	4.7 TB
SunFire E15k	UltraSparc3	672	20	4 TFlop/s	960 GB
Cray T3E	Alpha 21164A	2176	2176	3 TFlop/s	1TB

**Tabelle 7.3:** Parameter einiger Systeme mit verteiltem Speicher.

sowohl der *Shared Memory* als auch der *Distributed Memory* Architektur besitzen. Die Programmierung dieser hybriden Parallelrechner stellt eine besondere Herausforderung dar.

## 7.2 Amdahlsches Gesetz

Das nach Gene Amdahl benannte Gesetz (AMDAHL, 1967) ermöglicht eine prinzipielle Aussage über die maximal erreichbare Beschleunigung eines Programms durch die Parallelisierung. Das Programm bestehe aus einem sequentiellen Teil, der für seine Ausführung auf einem Prozessor die Laufzeit  $T_s$  benötigt. Den übrigen Anteil an der gesamten Laufzeit  $T$  des Programms trägt der parallelisierbare Teil mit der Laufzeit  $T_p$ . Die gesamte Laufzeit ist natürlich durch die Summe der Laufzeiten der beiden Programmteile gegeben:  $T = T_s + T_p$ . Unter der Annahme, dass eine Parallelisierung des Programms ohne einen

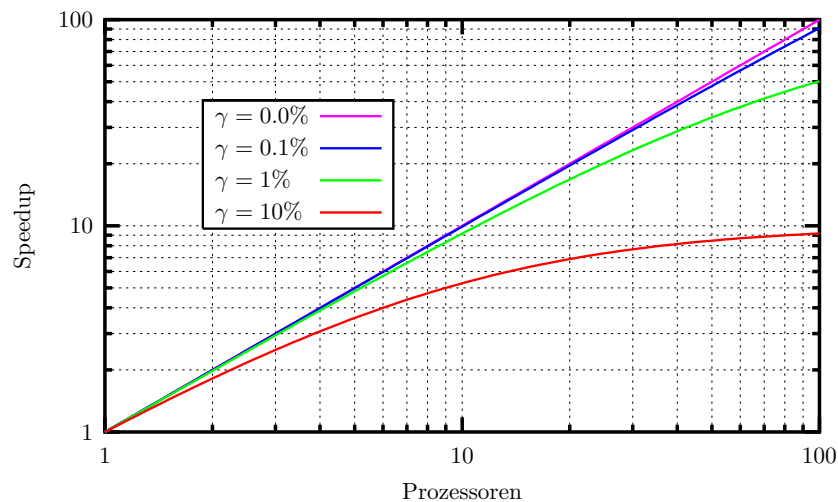
Mehraufwand zu realisieren ist und der parallelisierbare Teil des Problems eine unendlich feine Granularität besitzt und somit beliebig oft in parallele Teilaufgaben unterteilt werden kann, gilt:

$$T_N = T_s + \frac{T_p}{N} \quad (7.1)$$

mit  $N$  der Anzahl der Prozessoren. Die Beschleunigung der Berechnung ist definiert als das Verhältnis der Laufzeit des Problems auf einem Prozessor zu der Laufzeit des parallelen Programms auf  $N$  Prozessoren:

$$\beta(N) = \frac{T_0}{T_N} = \frac{T}{T_s + \frac{T_p}{N}} = \frac{N}{1 + \gamma(N - 1)} \quad (7.2)$$

mit  $\gamma = T_s/T$  dem Anteil des sequentiellen Programmteils an der gesamten Laufzeit. Das



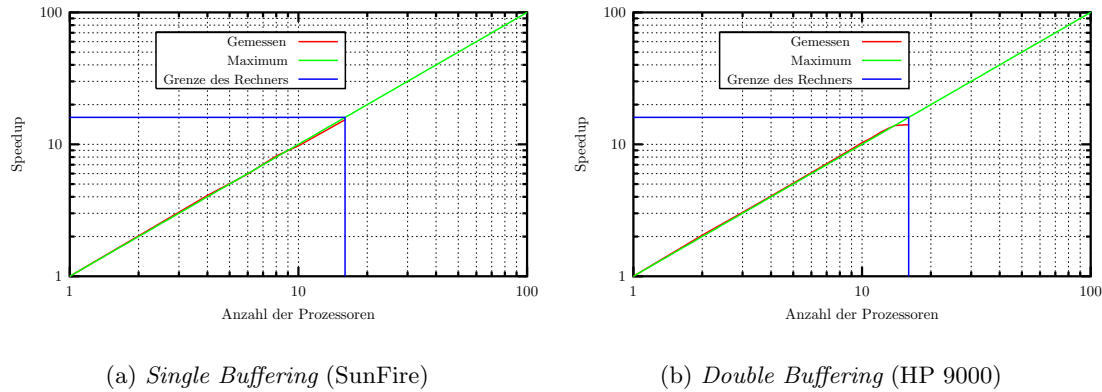
**Bild 7.2:** Das Amdahlsche Gesetz.

Amdahlsches Gesetz besagt, dass für ein Problem, das zu 75% parallel ist, eine maximale Beschleunigung der Berechnung um den Faktor 4 erreicht werden kann. Die wichtige Abhängigkeit der Beschleunigung von der Problemgröße wird dabei außer Acht gelassen, da das Gesetz von der Parallelisierung eines und des selben Problems ausgeht. Bei vielen Problemen wächst jedoch der sequentielle Anteil des Programms linear mit der Problemgröße während der parallele Teil des Problems quadratisch oder sogar kubisch wächst. Eine Beschreibung des Skalierungsverhaltens in Abhängigkeit der Problemgröße konnte durch eine Modifikation des Amdahlschen Gesetzes von [GUSTAFSON \(1988\)](#) gefunden werden.

### 7.3 Parallelisierung auf SMP Systemen

Eine Parallelisierung auf SMP Systemen mit wenigen Prozessoren kann sehr einfach realisiert werden, indem der Strahlengenerator die Strahlen in einen Puffer schreibt, der von einem *Worker Thread* abgearbeitet wird. Findet der *Worker Thread* einen leeren Strahlenpuffer vor, so ruft dieser die entsprechende Funktion des Strahlengenerators auf, um den Puffer zu füllen. Da die Strahlengenerierung deutlich schneller erfolgt als die Strahlverfolgung, entstehen durch diese einfache sog. *Master/Worker* Parallelisierung keine wesentlichen Wartezeiten der *Worker Threads*, die sich negativ auf das Skalierungsverhalten auswirken könnten. Die erforderliche Synchronisation der *Worker Threads* bei dem Zugriff auf den Strahlengenerator erfolgt durch eine sog. *Mutual Exclusive Access* (Mutex)

Variable. Die gemessene Beschleunigung (*Speedup*) dieser einfachen Parallelisierung ist in Bild 7.3a dargestellt. Aus dem Bild ist ersichtlich, dass eine nahezu lineare Beschleunigung durch dieses Verfahren erreicht werden kann. Ein etwas anderes Verfahren verwendet eine



**Bild 7.3:** Die gemessene Beschleunigung durch die Parallelisierung des Cone Tracing Verfahrens auf einem SMP System. Dargestellt sind die Messergebnisse beider Parallelisierungsansätze.

sog. *Double Buffering* Technik, um die Strahlverfolgung und die Strahlgenerierung zeitlich parallel durchführen zu können. Hierbei besitzt jeder *Worker Thread* zwei Strahlenpuffer, von denen jeweils der eine von der Strahlverfolgung abgearbeitet wird, während der andere Puffer parallel dazu von dem Strahlengenerator mit neuen Strahlen gefüllt wird. Diese Verfahren ist vorteilhafter auf Systemen mit vielen Prozessoren, bei denen sich die Zeit für eine Strahlengenerierung als die zentrale Komponente negativ auf die Geschwindigkeit der Simulation auswirken würde. Der negative Einfluss der zentralen Strahlengenerierung kann ebenfalls durch eine Vergrößerung des Strahlenpuffers verringert werden. Die Verwendung von lokalen Strahlenpuffern in jedem *Thread* ist ebenfalls vorteilhafter im Zusammenhang mit der Parallelisierung auf einer ccNUMA Architektur, da ein gemeinsamer Strahlenpuffer sich für alle *Threads* im langsamen globalen Speicher befinden würde. Dagegen befinden sich lokale Strahlenpuffer immer im schnellen lokalen Speicher des jeweiligen Rechenknotens, der den *Thread* ausführt. Die Synchronisation der *Double Buffering* Technik ist jedoch mit einem deutlich höheren Entwicklungsaufwand verbunden. Die Beschleunigung der Simulation durch diese Parallelisierung auf einem SMP System ist im Bild 7.3b dargestellt. Sie ist mit dem einfacheren *Single Buffering* Verfahren vergleichbar. Der Verlauf der gemessenen Beschleunigung beider Verfahren stimmt sehr gut mit den Erfahrungswerten anderer paralleler Ray Tracing Programme aus dem Bereich der Computer Graphik überein. Er bestätigt die Aussagen über die besondere Eignung von Ray Tracing Algorithmen für einen effizienten Einsatz der Parallelverarbeitung. Das Erreichen der Sättigung der im Bild 7.3b dargestellten Werte für > 15 Prozessoren ist durch die Konfiguration des Betriebssystems des eingesetzten Rechners bedingt.

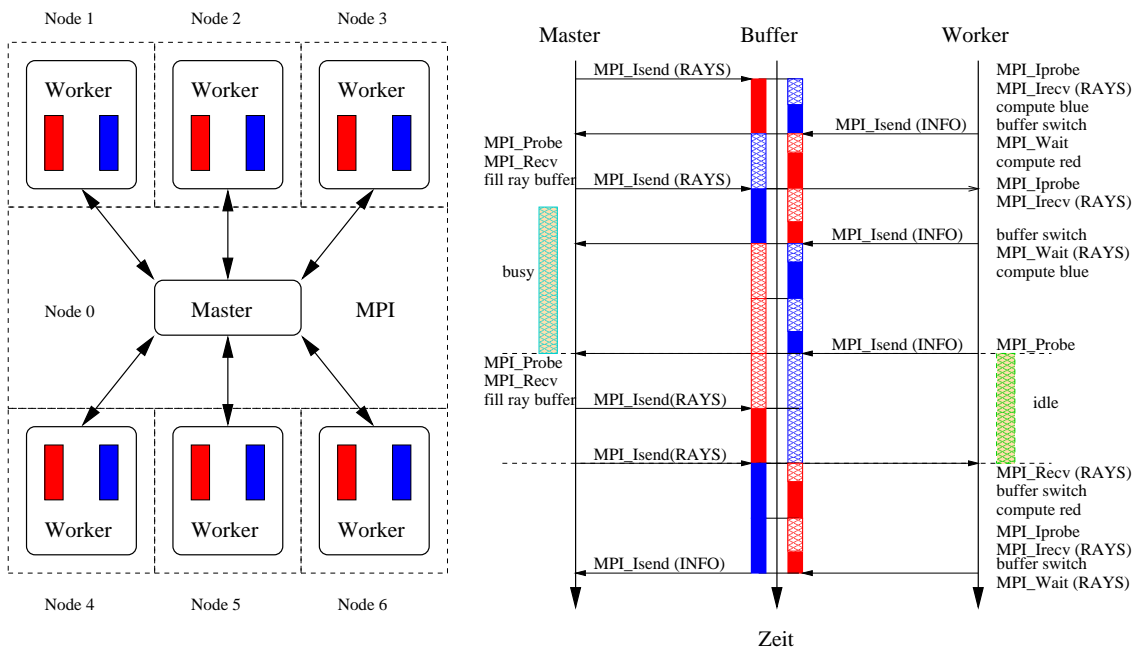
Im Verlauf der Strahlverfolgung werden neue Sekundärquellen für die Beugung und die Streuung erzeugt. Diese neuen Quellen müssen ebenfalls von den *Worker Threads* abgearbeitet werden. Da diese Quellen die Eigenschaften und Funktionen eines Senders besitzen, kann ihre Berechnung in der gleichen Art und Weise wie die der primären Quellen erfolgen. Nach ihrer Detektion während der Strahlverfolgung werden die neuen Quellen

auf unterschiedliche Stapel gelegt, damit nicht Sekundärquellen unterschiedlicher Ordnung verwechselt werden. Nachdem alle primären Quellen abgearbeitet wurden, wird die Liste aller Beugungquellen zur aktuellen Senderliste gemacht und abgearbeitet. Anschließend erfolgt die Abarbeitung der Liste der Streuquellen. In beiden Fällen können neue Beugungs- und Streuquellen entstehen. Diese müssen in neue Listen eingetragen und, entsprechend ihrer Ordnung, zu einem späteren Zeitpunkt abgearbeitet werden. Damit es zu keiner Vermischung der Quellen kommt, ist bei einem Listenwechsel eine *Barrier* Synchronisation aller *Threads* notwendig. Eine Alternative zu diesem Schema würde die Barrier Operation vermeiden, indem sie eine Hierarchie von *Threads* zur Abarbeitung der Sekundärquellen einsetzen würde. Somit könnten alle Sekundärquellen bereits unmittelbar nach ihrer Detektion in die Liste des untergeordneten *Thread* eingebunden werden und würden sogleich berechnet.

## 7.4 Parallelisierung auf Systemen mit verteiltem Speicher

Während die Parallelisierung für die shared memory Architektur vor allem aus der Synchronisation der Threads bestand, muss eine Parallelisierung für ein System mit verteiltem Speicher neben der Synchronisation die gesamte Kommunikation der Daten realisieren. Durch den Einsatz der MPI Bibliothek hält sich jedoch der Implementierungsaufwand in Grenzen.

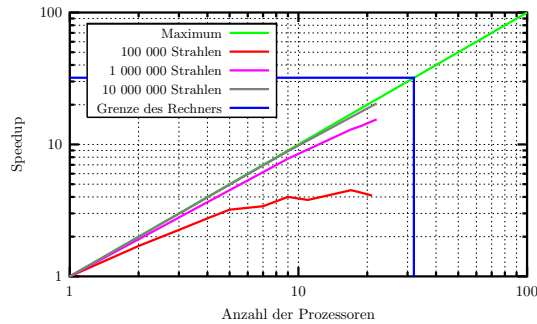
Die Parallelisierung für ein System mit verteiltem Speicher erfolgt ebenfalls nach dem *Master/Worker* Schema (Bild 7.4) zusammen mit der *Double Buffering* Technik, um die Kommunikation der Strahlenpuffer mit der Berechnung parallel durchführen zu können. Die Synchronisation der Strahlenpuffer mit der dafür benötigten Kommunikation über die MPI Schnittstellen ist ebenfalls im Bild 7.4 dargestellt.



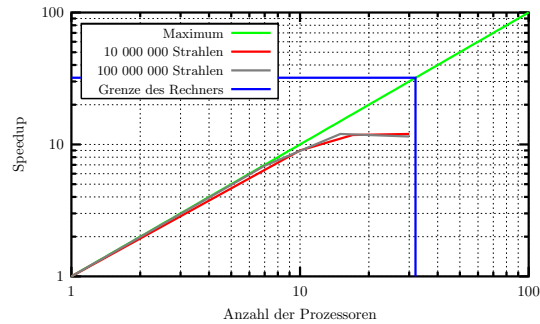
**Bild 7.4:** Die *Master/Worker* Parallelisierung mit der *Double Buffering* Technik.

Der Ablauf der Berechnung besteht aus drei Phasen. In der ersten sog. *Setup* Phase werden die Worker Prozesse auf den zugewiesenen Knoten des Parallelrechners gestartet und die Geometriedaten werden auf diese Worker verteilt. In der zweiten Phase wird

die parallele Berechnung durchgeführt. Hierbei generiert der Master Prozess auf Anfrage Strahlenpuffer und versendet diese an die entsprechenden Worker Prozesse. Die Worker Prozesse führen die Berechnung der Strahlen durch und senden alle detektierten Sekundärquellen an der Master Prozess zurück, der einen neuen Strahlenpuffer generiert. In der dritten und abschließenden Phase werden die Ergebnisse der Berechnung der einzelnen Worker von dem Master Prozess eingesammelt und kombiniert. Hierbei müssen auch alle mehrfach gefundenen Pfade aussortiert werden. Die resultierende Beschleunigung des Ray Tracing Verfahrens auf einem Beowulf Cluster ist im Bild 7.5a in Abhängigkeit der Problemgröße sehen. Die double buffering Technik ist aufgrund der langsamen Kommu-



(a) Beschleunigung (Siemens HPCLine)



(b) Begrenzung durch Netzwerkbandbreite (Siemens HPCLine)

**Bild 7.5:** Die Beschleunigung der Berechnung durch eine Parallelisierung auf einem System mit verteiltem Speicher.

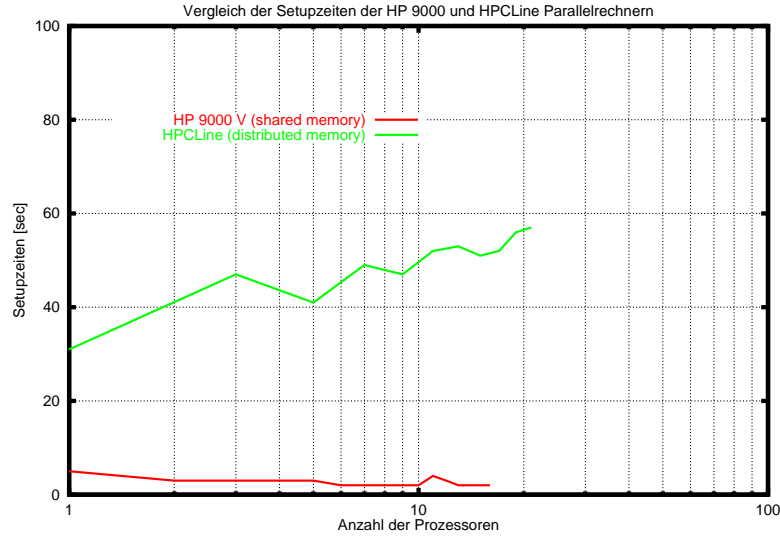
nikation zwischen den Knoten sehr wichtig für eine effiziente Ausnutzung der Rechenleistung und damit das Erreichen der maximalen Beschleunigung der Berechnung. Durch die double-buffering Technik wird gewissermaßen die Kommunikation hinter der Berechnung versteckt<sup>3</sup>. Dies geschieht jedoch nur unter der Voraussetzung, dass die Berechnung des Strahlenpuffers mehr Zeit in Anspruch nimmt als der Datenaustausch des anderen Puffers. Bei einer steigenden Zahl der Worker kommt der Master mit der Generierung und der Kommunikation neuer Strahlen nicht nach und es stellt sich eine charakteristische Sättigung der Speedup Funktion ein (Bild 7.5b). Eine Abhilfe schafft nur eine deutliche Verringerung der auszutauschenden Datenmenge oder eine dezentrale Parallelisierung ohne einen Master Prozess.

Die im Bild 7.5a dargestellte Abhängigkeit der Beschleunigung von der Problemgröße (Strahlendichte, Größe des Szenario) macht deutlich, dass eine Parallelisierung auf einem System mit verteiltem Speicher nur für besonders große Probleme sinnvoll ist. Der Grund für dieses Verhalten liegt in der bereits angesprochenen unterschiedlichen Skalierung des sequentiellen und des parallelisierbaren Anteils der Berechnung. Während der sequentielle Anteil durch die Setup Phase und das Einsammeln der Ergebnisse gegeben ist und bei einer konstanten Größe des Szenarios und gleichbleibender Anzahl der Empfänger unabhängig von der Strahlenanzahl ist, nimmt der parallelisierbare Anteil an der Berechnung mit der Strahlenanzahl stark zu. Bei einer sehr hohen Strahlendichte dominiert der parallele Anteil der Berechnung die gesamte Rechenzeit und sorgt für eine effiziente Ausnutzung der

<sup>3</sup>Viele Parallelrechner mit einem verteiltem Speicher besitzen von dem Prozessor unabhängige Kommunikationsprozessoren, die den Datenaustausch zwischen den Rechenknoten autonom durchführen.



Prozessorleistung. Bei einer abnehmenden Strahlendichte nimmt der Anteil der parallelen Berechnung an der gesamten Laufzeit ab und der sequentielle Anteil dominiert. Dieses Verhalten ist grundsätzlich auch bei einem SMP System vorhanden, aufgrund der sehr viel kürzeren Setup-Zeiten macht es sich jedoch nicht so stark bemerkbar (Bild 7.6).



**Bild 7.6:** Vergleich der Setup-Zeiten der parallelen Berechnung auf einem SMP System und auf einem System mit verteiltem Speicher.

Die Größe der übertragenen Nachrichten beeinflusst entscheidend den Durchsatz des Kommunikationsnetzwerks. Diese Auswirkung der Nachrichtenlänge auf die Kommunikationsgeschwindigkeit kann in erster Näherung mit der Pipeline-Theorie von [HOCKNEY und JESSHOPE \(1981\)](#) untersucht werden. Aus dieser Untersuchung folgt die Festlegung einer minimalen Nachrichtenlänge und somit auch einer minimalen Strahlenpuffergröße.

Die Zeit für die Übertragung einer Nachricht der Länge  $n$  Bytes über einen Kommunikationskanal mit der Zykluszeit  $\tau$  und der Setupzeit  $T_{setup} = s\tau$  beträgt

$$T = T_{setup} + n\tau = (s + n)\tau . \quad (7.3)$$

Die Übertragungszeit für eine Einheit ist

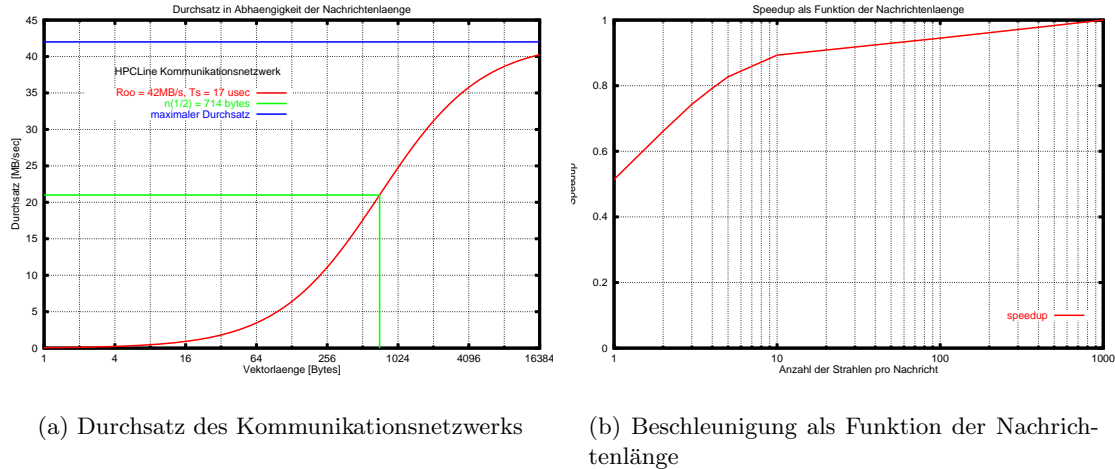
$$T_{unit} = \frac{T}{n} = \frac{s + n}{n}\tau . \quad (7.4)$$

Daraus ergibt sich für den Durchsatz des Kommunikationskanals

$$r = \frac{n}{s + n}\tau^{-1} = \frac{n}{s + n}r_{\infty} = \frac{r_{\infty}}{\frac{n_{1/2}}{n} + 1} . \quad (7.5)$$

Die Größen  $n_{1/2}$  und  $r_{\infty}$  sind die charakteristischen Werte eines Kommunikationsnetzwerks. Der Wert von  $n_{1/2}$  definiert die Länge einer Nachricht, bei der das Kommunikationsnetzwerk den halben maximalen Durchsatz erreicht, da  $s = n_{1/2}$  ist. Der Wert  $r_{\infty}$  gibt den maximalen Datendurchsatz des Kanals an, der für  $n \rightarrow \infty$  erreicht wird. Der Durchsatz des Netzwerks des verwendeten Beowulf Clusters mit  $n_{1/2} = 714$  Bytes und  $r_{\infty} = 42$  MBytes/sec ist im Bild 7.7a dargestellt. Die Nachrichtenlänge  $n_{1/2}$  entspricht einer Übertragung der Daten von 23 Strahlen. Im Bild 7.7b wurde der Einfluss der Größe des Strahlenpuffers und damit auch der Nachrichtenlänge auf die Effizienz der





(a) Durchsatz des Kommunikationsnetzwerks

(b) Beschleunigung als Funktion der Nachrichtenlänge

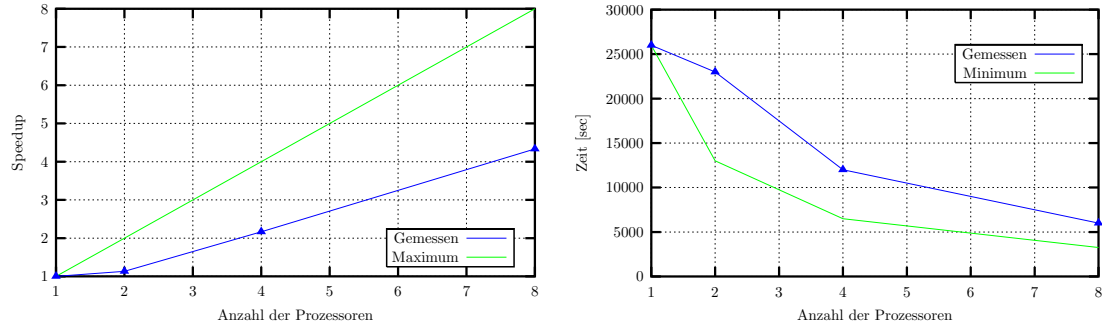
**Bild 7.7:** Auswirkungen der Nachrichtenlänge auf den Durchsatz des Kommunikationsnetzwerks und die erreichte Beschleunigung des Simulators.

Berechnung unter der Einbeziehung der double-buffering Technik gemessen. Es wurde die Anzahl der Strahlen, die in einer Nachricht auf einmal übertragen werden, variiert und die Auswirkungen auf die Dauer der Simulation untersucht. Es wird deutlich, dass mit einer abnehmenden Länge der Nachrichten ebenfalls die Effizienz der Parallelisierung abnimmt. Es ist daher gerade bei langsamen Kommunikationsnetzwerken mit relativ langen Setup-Zeiten (Ethernet) dringend nötig, möglichst lange Nachrichten zu übertragen, um den vollen Durchsatz des Kommunikationsnetzwerks zu erreichen.

## 7.5 Parallele Optimierung

Die Parallelverarbeitung kann ebenfalls zu einer Beschleunigung der der im Kapitel 5 behandelten Optimierung des BSP Baumes eingesetzt werden. Insbesondere die genetische Optimierung kann sehr leicht auf einem SMP Rechner parallelisiert werden. Die Auswertung der Fitness-Funktion stellt bei der genetischen Optimierung den zeitlich aufwendigsten Teil der Optimierung dar, da für jedes Chromosom im genetischen Pool der entsprechende BSP Baum konstruiert und vermessen werden muss. Da die Konstruktion der Bäume für verschiedene Chromosome unabhängig voneinander durchgeführt werden kann, kann diese ideal parallelisiert werden. Eine Parallelisierung auf Systemen mit einem verteiltem Speicher wäre ebenfalls leicht möglich, wurde jedoch im Rahmen dieser Arbeit nicht betrachtet. Das Ergebnis der Beschleunigung der Optimierung auf einem SMP Rechner ist im Bild 7.8 zu sehen.

Der Verlauf der im Bild 7.8 dargestellten Beschleunigung deutet auf ein Problem bei der Parallelisierung hin. Aufgrund der Unabhängigkeit der Teilprobleme ist ein deutlich besserer Speedup der Optimierung zu erwarten. Die Ursache der mäßigen Beschleunigung liegt hier im Betriebssystem des verwendeten Parallelrechners. Da bei der Konstruktion des BSP Baumes sehr häufig Speicher angefordert und wieder freigegeben wird, werden die entsprechenden Funktionen des Betriebssystems wiederholt aufgerufen. Es stellte sich heraus, dass diese Funktionen eine Synchronisierung aller laufender Threads zur Folge haben (eine sog. *Barrier* Operation). Durch diese Synchronisierung entstehen Wartezeiten, die sich negativ auf den Speedup auswirken. Eine mögliche Abhilfe besteht darin, dass jeder Thread eine interne Speicherverwaltung realisiert und den dafür benötigten Speicher



(a) Beschleunigung der Optimierung

(b) Rechenzeit der Optimierung

**Bild 7.8:** Verlauf der Beschleunigung und der Rechenzeit einer parallelen Optimierung des BSP Baumes mit einem genetischen Algorithmus auf einem SMP Rechner.

zuvor nur einmal anfordert.

## 7.6 Zusammenfassung

In diesem Kapitel wurde die Parallelisierung des Cone Tracing Verfahrens aus Kapitel 6 auf Systemen mit einem gemeinsamen Speicher und ebenso auf Systemen mit einem verteilten Speicher betrachtet. Nach einer Einführung in die Besonderheiten der verschiedenen Parallelrechner-Architekturen wurden die theoretischen Grenzen einer Parallelisierung anhand des Gesetzes von Amdahl erklärt. Die Parallelisierung auf symmetrischen Multiprozessoren wurde mit der single-buffering und der double-buffering Technik durchgeführt und die erreichte Beschleunigung der Simulation auf zwei verschiedenen SMP Rechnern dargestellt. Die Parallelisierung des Verfahrens auf einem Beowulf Cluster mit verteiltem Speicher mit Hilfe der Message Passing Interface Bibliothek wurde ebenfalls nach einem Master/Worker Schema mit der double-buffering Technik realisiert und der gemessene Speedup für drei verschiedene Problemgrößen wurde angegeben. Aus dem Vergleich der Setup-Zeiten der parallelen Berechnung auf einem SMP Rechner und dem Beowulf Cluster wird deutlich, dass eine parallele Simulation auf Rechnern mit einem verteilten Speicher erst bei sehr großen Problemen effizient durchgeführt werden kann. Abschließend wurde auch eine Parallelisierung der Optimierung des BSP Baumes mit einem parallelen genetischen Algorithmus auf einem symmetrischen Multiprozessor betrachtet. Hier stellte heraus, dass die Effizienz der Parallelisierung stark von einer effizienten Implementierung der Speicherallokation durch das Betriebssystem des Parallelrechners abhängig ist.



## Verifikation und Anwendungsbeispiele

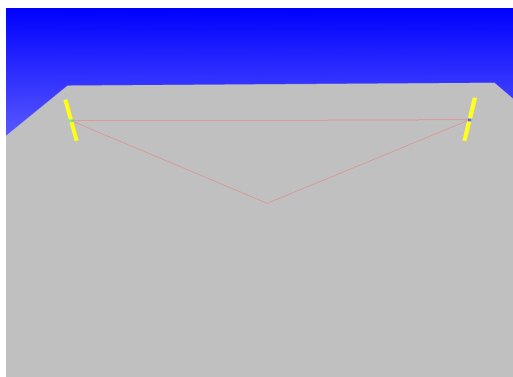
In diesem Kapitel wird zunächst das eingesetzte Wellenausbreitungsmodell an verschiedenen einfachen und komplexen Szenarien getestet und mit Messungen oder anderen, getesteten Simulationswerkzeugen verglichen. Anschließend wird das getestete Werkzeug bei der Lösung verschiedener Wellenausbreitungsprobleme eingesetzt und die so gewonnenen Ergebnisse vorgestellt.

### 8.1 Verifikation mit exakten Verfahren

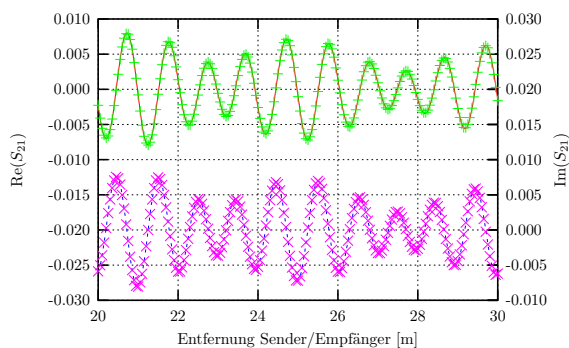
Zunächst sollen die Simulationsergebnisse für einige einfache Fälle mit Lösungen exakter Verfahren wie der Spiegelungsmethode verglichen werden.

#### 8.1.1 Spiegelung an einer leitenden Ebene

In einem ersten Test soll das Ray Tracing Verfahren mit dem einfachen zwei-Wege Ausbreitungsmodell verglichen werden. Die Übertragungsfunktion zwischen zwei  $\lambda/2$  Dipolen über einer ideal leitenden Ebene kann nach dem Spiegelungsprinzip bestimmt werden (BALANIS, 1997). Die genaue Stromverteilung auf der Oberfläche der Dipole wird mit der Momentenmethode (HARRINGTON, 1982) berechnet. Dieser Weg ermöglicht sowohl die Betrachtung und die einfache Modifikation beliebiger Dipolkonfigurationen als auch die Berücksichtigung des für die Speisung der Dipole notwendigen Luftspalts. Die schematische Darstellung der Anordnung ist im Bild 8.1a zu sehen. Für den Vergleich beider



(a) Schematische Anordnung



(b) Vergleich MoM/Ray Tracing

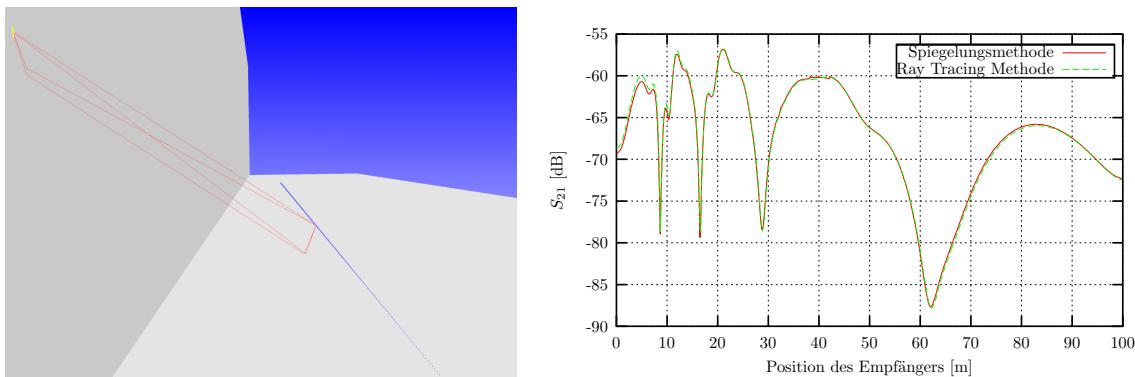
**Bild 8.1:** Verifikation des Ray Tracing Wellenausbreitungsmodells durch den Vergleich mit der Momentenmethode. Für die exakte Berechnung des komplexen Übertragungsfaktors unter realen Bedingungen wurde das Programm NEC (Burke und Poggio, 1981) eingesetzt.

Verfahren wurden zwei Dipole bei einer Frequenz von 300 MHz in 10 Meter Höhe über der

metallenen Ebene angeordnet. Die Entfernung wurde von 20 bis 30 Meter variiert. Das Ergebnis der Simulation mit dem auf der Momentenmethode basierten Simulator NEC (BURKE und POGGIO, 1981) und dem strahlenoptischen Ansatz ist im Bild 8.1 zu sehen.

### 8.1.2 Spiegelung an einer dielektrischen Ecke

Ein weiteres Testszenario besteht aus einer Anordnung von linienförmigen Quellen mit konstanter Belegung entlang einer dielektrischen Ecke. Die strenge numerische Lösung wurde durch den Einsatz einer Spiegelungsmethode gewonnen (REMBOLD, 1993), die anhand von Messungen in einem Tunnel validiert wurde (PREDIGER et al., 2000). Das Szenario und der Vergleich der Ray Tracing Simulation mit der angewandten Spiegelungsmethode ist im Bild 8.2 zu sehen. Der Sender befindet sich in 10 Meter Höhe 0.5 Meter vor der senkrechten Wand. Die Empfänger sind in einer Höhe von 1.6 Meter in einem Abstand von 10 Meter vor der senkrechten Wand angeordnet. Sowohl die Sendeantenne als auch die Empfangsantennen werden als vertikal angeordnete, linienförmige Quellen angenommen und besitzen somit eine omnidirektionale Charakteristik im Azimut und eine durch die konstante Stromverteilung entlang ihrer Achse gegebene  $\sin(x)/x$  Charakteristik in der Elevation bei einer vertikalen Polarisation. Die Halbwertsbreite in der Elevation beträgt 78 Grad bei 1.9 GHz. Die relative Dielektrizitätszahl der Wände wurde mit  $\epsilon_r = 10$  als verlustlos angenommen.



(a) Geometrie des Szenario

(b) Vergleich Spiegelungsmethode/Ray Tracing

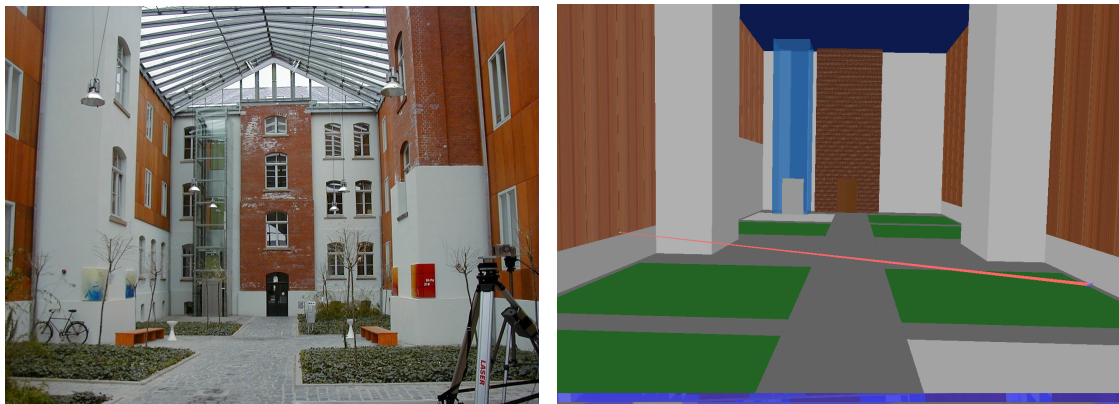
**Bild 8.2:** Verifikation des Wellenausbreitungsmodells durch den Vergleich mit der Spiegelungsmethode. Eine linienförmige Anordnung von Empfängerantennen entlang einer Ecke wird von einer Sendeantenne an der senkrechten Wand angestrahlt. Das Ergebnis wird durch die Überlagerung von vier verschiedenen Ausbreitungspfaden gewonnen.

## 8.2 Validierung durch Messungen

Für die weitere Überprüfung des Wellenausbreitungsmodells ist ein Vergleich der Simulation mit Messungen in komplexen realen Szenarien unerlässlich. Die erste Validierung kann durch einen Vergleich des vorhergesagten mit dem tatsächlich gemessenen Empfangspegel erreicht werden. Hierbei ist der notwendige Messaufbau verhältnismäßig einfach zu realisieren. In der Regel wird ein *Continuous Wave* (CW) Sender bei der gewünschten

Frequenz zusammen mit einer omnidirektionalen Antenne eingesetzt. Der Empfänger kann im einfachsten Fall aus einem kalibrierten Heterodynempfänger bei der gleichen Frequenz bestehen. Als Antenne kann abhängig von der Zielsetzung sowohl eine omnidirektionale Antenne als auch eine Antenne mit einer direktionalen Richtcharakteristik zum Einsatz kommen. In der Regel werden als Empfänger rechnergesteuerte Spektrumanalysatoren eingesetzt (SCHÖBERL, 1997). Ebenfalls nötig ist eine genaue Positionserfassung aller dynamischen Komponenten des Systems. Auf diese Weise kann eine Aussage über die zu erwartende Feldstärke am Ort des Empfängers getroffen werden, es können jedoch keine Aussagen über die zeitliche Verzögerungen der Ausbreitungspfade und die Richtungsabhängigkeit der Ausbreitungssituation gemacht werden. Die frequenzselektiven Eigenschaften des Raumes können durch eine synchrone Verstimmung des Generators und des Messempfängers gemessen werden.

Eine weitere deutliche Verbesserung der Aussagekraft einer Messung wird durch die Verwendung eines zeit- und richtungsauflösenden *Channel Sounder* (CS) erzielt. Durch die zeitliche Auflösung kann die Verzögerung der einzelnen Pfade aufgrund ihrer unterschiedlichen Laufzeit bestimmt werden. Die Richtungsauflösung bestimmt dagegen die Ankunftsrichtungen der Pfade und lokalisiert die Winkelposition der diskreten Spiegelquellen. Beide Informationen können für die Entwicklung neuer statistischer Ausbreitungsmodelle eingesetzt werden (DIETERT, 2001).

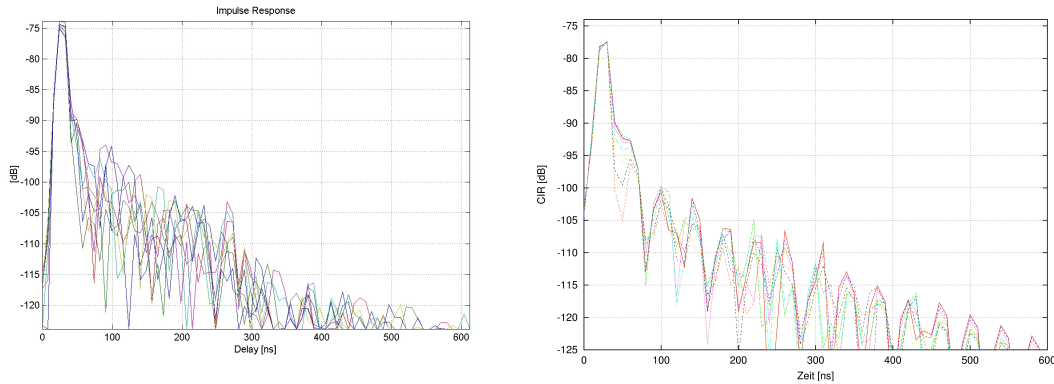


(a) Ansicht des Szenario

(b) Ansicht des 3D Modells

**Bild 8.3:** Das Szenario und das in der Simulation eingesetzte 3D Modell für die Verifikation mit CS Messungen.

Entscheidend für einen aussagekräftigen Vergleich der Simulation mit der Messung ist die detailtreue Modellierung des Szenarios (Bild 8.3). Zu diesem Zweck müssen sowohl die Geometrie als auch die Materialparameter des Raumes genau bekannt sein. Beide Bedingungen können jedoch aus Zeit- und Kostengründen selten eingehalten werden, so dass Einschränkungen bezüglich der Detailtreue in Kauf genommen werden müssen. In der Praxis ist daher mit mehr oder weniger großen Abweichungen der Ergebnisse des simulierten Modells von den gemessenen Werten zu rechnen. Ziel der Modellierung sollte es sein, nur die für die Wellenausbreitung relevanten Eigenschaften des Raumes korrekt nachzubilden um eine prinzipielle Vorhersage der Wellenausbreitung zu ermöglichen. Die Entscheidung, welche Strukturen im Szenario vernachlässigt werden können, kann nur durch Erfahrung und belegende Messungen getroffen werden.



**Bild 8.4:** Messdaten des RUSK ATM Channel Sounders und die simulierten Ergebnisse unter Einbeziehung des pulse-shaping Filters. Die Linien geben die Lage der diskreten Strahlen.

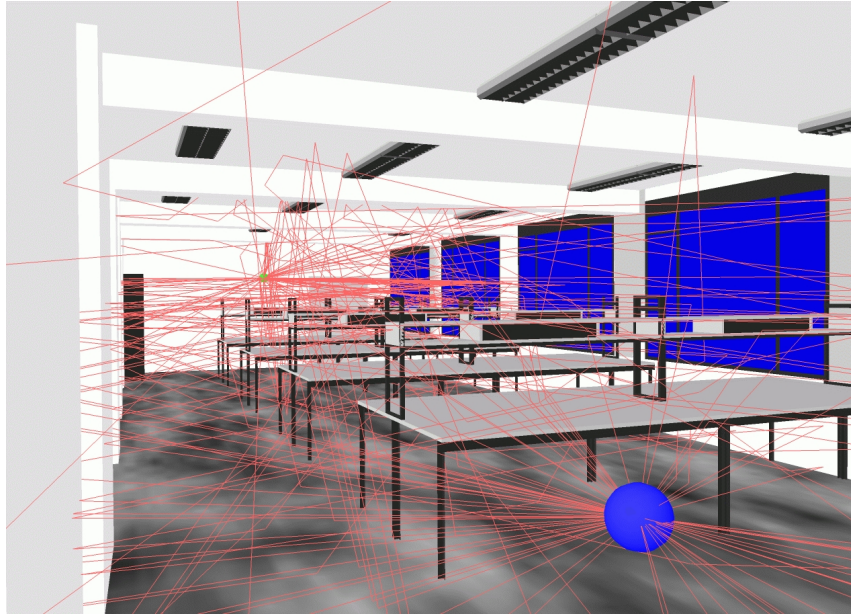
Im Bild 8.3 ist eine Ansicht und das 3D Modell des Testszenario auf dem Gelände der Technischen Universität Ilmenau dargestellt. In einer Messkampagne der Firma MEDAV, die in Zusammenarbeit mit der Universität Karlsruhe durchgeführt wurde, ist die Kanalimpulsantwort mit dem richtungsaufgelösten RUSK ATM Channel Sounder für viele verschiedene Sender- und Empfängerpositionen vermessen worden. Einige Daten der Messkampagne sind freundlicherweise von der Firma MEDAV für die Validierung des hier eingesetzten Ausbreitungsmodells zur Verfügung gestellt worden. Ein Vergleich der Messung mit der Simulation ist im Bild 8.4 dargestellt. Bei der Simulation wurde ebenfalls der Einfluss des pulse-shaping Filters einbezogen.

### 8.3 Simulation der Wellenausbreitung in Räumen

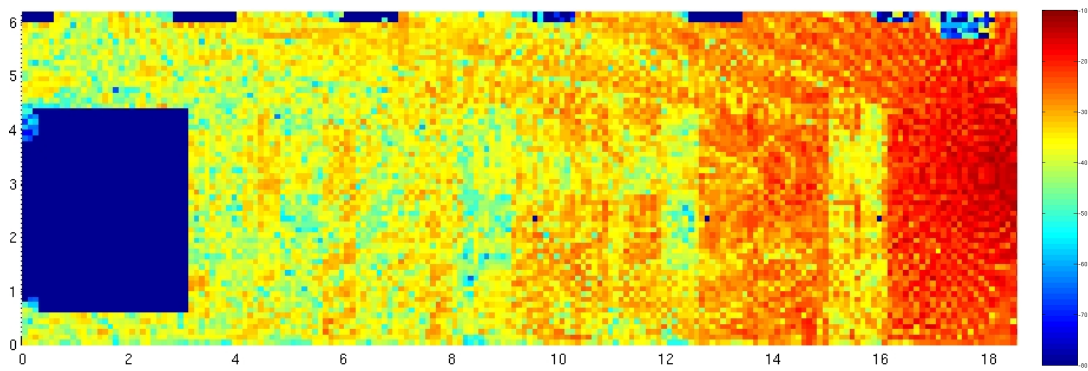
Eine wichtige Anwendung für deterministische Wellenausbreitungsmodelle bildet die Nahbereichskommunikation. Durch das Aufkommen der drahtlosen lokalen Datennetze und der Telefone nach dem *Digital Enhanced Cordless Telecommunication* (DECT) 1800 Standard wird eine Funknetzplanung innerhalb beschlossener Räume benötigt. Eine deterministische polarimetrische Untersuchung der Wellenausbreitung in solchen pikozellularen Netzen wurde von SCHÖBERL (1997) durchgeführt. Neuartige breitbandige Kommunikationsverfahren nach dem MIMO Prinzip nutzen die Mehrwegeausbreitung gezielt aus, um eine Erhöhung der Kommunikationsbandbreite zu erreichen. Für eine Bewertung solcher Mehrantennensysteme werden sowohl Messungen als auch Simulationen in einer pikozellularen Umgebung benötigt.

Eine solche Umgebung ist in Bild 8.5 dargestellt. Das verwendete Modell eines Labors des Instituts für Hochfrequenztechnik der RWTH Aachen ist im Rahmen einer Untersuchung zur Kanalkapazität eines am Institut entwickelten MIMO Systems modelliert worden. Das Modell beinhaltet eine detailgetreue Nachbildung des Mobiliars, verzichtet jedoch auf die Modellierung frei beweglicher Gegenstände wie der Stühle und eventuell im Raum befindlicher Geräte. Im Bild sind die gefundenen Ausbreitungspfade für eine Beispielaufstellung des Senders (grün) und des Empfängers (blau) eingezeichnet. Bei der Simulation wurden maximal 6 Reflexionen berücksichtigt.

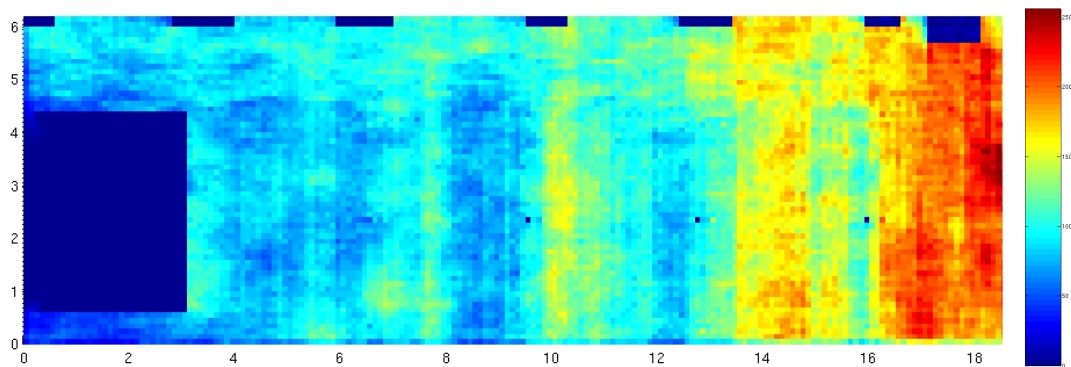




**Bild 8.5:** Das 3D Modell eines Labors des Instituts für Hochfrequenztechnik, das für die folgende Untersuchung eingesetzt wurde.



**Bild 8.6:** Die Übertragungsdämpfung zwischen einem isotropen Sender und einer flächendeckenden Anordnung isotroper Empfänger in 1.15 m Höhe.



**Bild 8.7:** Anzahl der detektierten Ausbreitungspfade in Abhängigkeit der Empfängerposition.

Die Wellenausbreitung innerhalb des Raumes wurde bei der anvisierten Mittenfrequenz des MIMO Systems von 10.5 GHz simuliert. Für die elektrischen Eigenschaften der Wände und der Gegenstände im Raum wurden typische Werte aus der Literatur eingesetzt. Da die Ausbreitungsbedingungen vor allem von der Vielzahl der reflektierten Pfade dominiert sind, ist nach einer Untersuchung von FISCHER (2002b) mit einer relativ schwachen Auswirkung der Variation der Materialparameter auf die Ausbreitungssituation zu rechnen. In weiten Bereichen kommt zu der Menge der reflektierten Strahlen auch eine starke *Line-Of-Sight* (LOS) Komponente hinzu. Der Verlauf der Übertragungsdämpfung zwischen einem isotropen Sender an der hinteren Wand des Raumes und einer flächendeckenden Verteilung von isotropen Empfängern ist im Bild 8.6 zu sehen. Sowohl der Sender als auch die Empfänger waren vertikal polarisiert.

Im Bild 8.7 ist die Anzahl der gefundenen Ausbreitungspfade als Funktion der Empfängerposition angegeben. Das Bild verdeutlicht den Ausmaß der Mehrwegeausbreitung in einem solchen Szenario. Die Vielzahl der Ausbreitungspfade in der Nähe des Senders ist auf Mehrfachreflexionen zwischen Boden und Decke sowie zwischen den beiden seitlichen Wänden zurückzuführen.

#### 8.4 Funknetzplanung für den Transrapid

Die Magnetschwebebahn Transrapid verfügt über eine permanente Datenverbindung zwischen dem Fahrzeug und Basisstationen, die entlang der Trasse positioniert sind. Die Datenverbindung wird mit Hilfe eines hochredundanten 38 GHz Richtfunksystems ermöglicht. Das Fahrzeug verfügt auf der ersten und auf der letzten Sektion über ein Antennensystem, das zusätzlich zu der Makrodiversität der beiden Antennengruppen auch die Mikrodiversität innerhalb der Gruppe für die Gewährleistung einer zuverlässigen Kommunikation ausnutzt (Bild 8.8). Zusätzlich wird zirkulare Polarisation ausgenutzt, um einfache Reflexionen zu unterdrücken. Die Datenverbindung wird mit einer *Continuous*

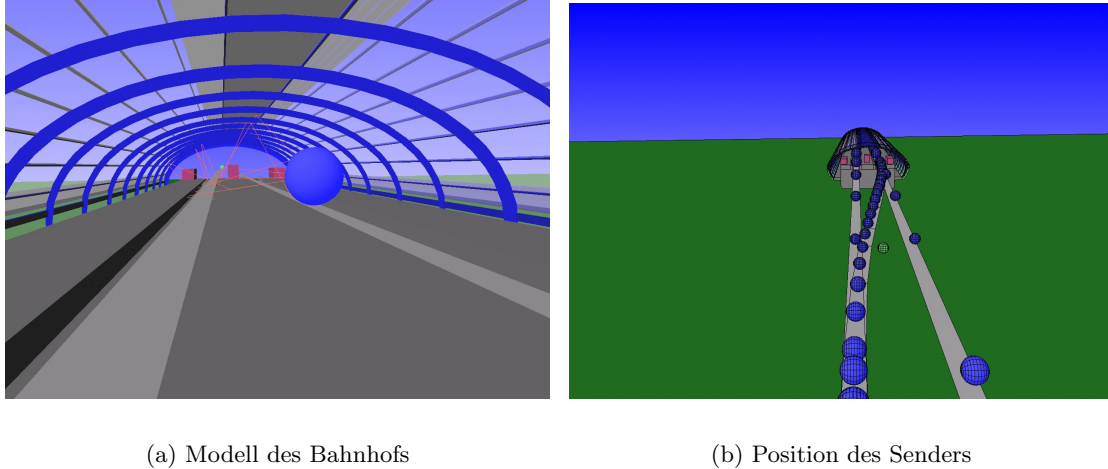


**Bild 8.8:** Die Magnetschwebebahn Transrapid auf der Versuchsstrecke in Lathen, Emsland (aus Transrapid (2002)).

*Phase Frequency Shift Keing* (CPFSK) Modulation bei einer Datenrate von 4Mbit/s realisiert. Am Institut für Hochfrequenztechnik der RWTH Aachen wurden Werkzeuge zur Funknetzplanung linearer Netzwerke entwickelt und unter anderem bei der Planung der Transrapid Trasse in Schanghai eingesetzt (FISCHER, 2002a). Während die Berechnung

der Positionen der Basisstationen entlang der Trasse automatisch durchgeführt wird (FISCHER, 2002b), stellt die Positionierung der Basisstationen in der Nähe oder innerhalb der Bahnhöfe einen iterativen Prozess dar, der manuell unter den gegebenen komplexen Grenzbedingungen durchgeführt werden muss.

Das in dieser Arbeit beschriebene Werkzeug wurde bei der Berechnung der Wellenausbreitung in den Bahnhöfen am Longyang Road und am Pudong Flughafen in Schanghai eingesetzt. Hierzu mussten die Bahnhöfe möglichst wirklichkeitsgetreu modelliert werden. Das Modell des Bahnhofs am Longyang Road ist im Bild 8.9 dargestellt. Die Planung ging

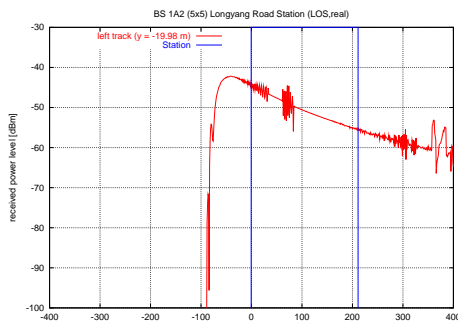


**Bild 8.9:** Ein 3D Modell des Bahnhofs Longyang Road Station in Schanghai. Das Dach wurde aus Gründen besserer Übersichtlichkeit in dieser Darstellung transparent gemacht. Die Positionen des Senders und der Empfänger sind mit grünen und blauen Kugeln markiert.

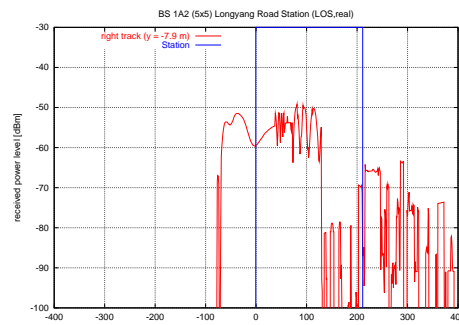
von einer Positionierung der Basisstationen auf beiden Seiten des Bahnhofs aus. Während bei den Basisstationen auf der rückwärtigen Seite des Bahnhofs die Positionierung neben der Fahrbahn keine Probleme hinsichtlich der Funkversorgung beinhaltete, stellte die Positionierung der Basisstation auf der anderen Seite des Bahnhofs ein Problem dar. Da sich in diesem Bereich auch die Weiche des Transrapid befindet, sind die Möglichkeiten bei der Platzierung der Basisstation eingeschränkt. Zusätzlich zu der eingeschränkten Bewegungsfreiheit kommt noch die Forderung erschwerend hinzu, dass beide Fahrbahnen im Bahnhofsbereich von dieser Basisstation versorgt werden müssen. Die Sichtlinie zwischen der Basisstation und einer der Fahrbahnen wird jedoch durch einen der beiden Fahrstühle auf der mittleren Plattform abgeschattet. Im Bild 8.10a ist die Abschattung im hinteren Bereich der rechten Fahrspur deutlich sichtbar. Bei der Berechnung der Wellenausbreitung im Bereich des Bahnhofs wurden nur Reflexionen an der Bahnhofsstruktur und am Boden berücksichtigt (FRACH und FISCHER, 2002). Eine Abschattung der rechten Fahrspur konnte durch eine Verschiebung der Basisstation von der Bahnhofshalle weg verhindert werden.

Aus den berechneten Ausbreitungswegen können nun weitere, für die Funktion des Systems relevante Parameter bestimmt werden. Zu diesen Parametern gehört der *Multipath Delay Spread* (MPDS) ( $\tau_{rms}$ ) und bei Einbeziehung eines Empfängermodells ebenfalls die Bitfehlerrate (FISCHER, 2002b). Die letztgenannte Größe ist im Bild 8.11 für beide Fahrbahnen aufgetragen. Bei der Berechnung der Bitfehlerrate wurde die Diversität des





(a) Linke Fahrspur



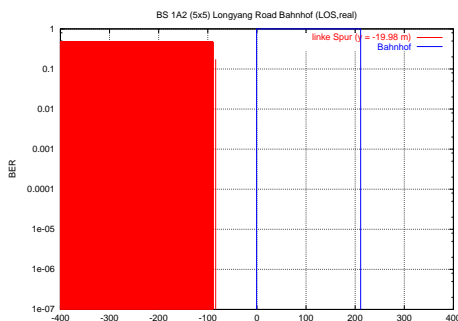
(b) Rechte Fahrspur

**Bild 8.10:** Verlauf der Empfangsleistung entlang beider Fahrbahnen vor dem Bahnhof und im Bahnhofsbereich. Der Bereich der Bahnhofshalle ist durch blaue Linien eingezeichnet.

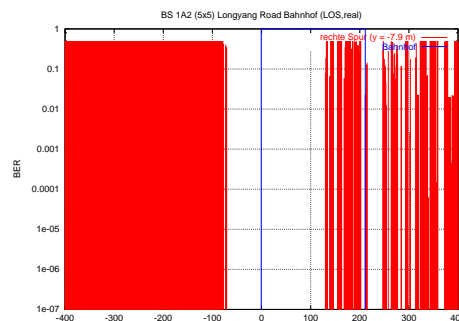
Systems nicht betrachtet. Aus diesem Grund steigt die Bitfehlerrate im Bereich der Abschattung auf der rechten Fahrspur stark an. Durch die Verschiebung der Basisstation konnte ebenfalls die hohe Bitfehlerrate in diesem Bereich eliminiert werden.

Der Bahnhof am Internationalen Flughafen Pudong besteht aus einer 180 Meter langen abgeschlossenen Halle mit einem anschließenden, 200 Meter langen Tunnel. Es ist in der Struktur vergleichbar mit einer U-Bahn Station. Durch die relative Enge des Raumes ist eine starke Mehrwegeausbreitung zu erwarten. Das Modell des Bahnhofsinnenen ist im Bild 8.12a zu sehen. Es wurden nur die für die Wellenausbreitung relevanten Objekte im Bahnhofsbereich modelliert. Die Einkaufs- und Büroräume in den benachbarten Etagen wurden bei der Berechnung der Wellenausbreitung nicht berücksichtigt.

Ebenso wie bei dem Bahnhof Longyang Road Station wird die Funkversorgung des Bahnhofs am Pudong International Airport von Basisstationen außerhalb der Bahnhofs

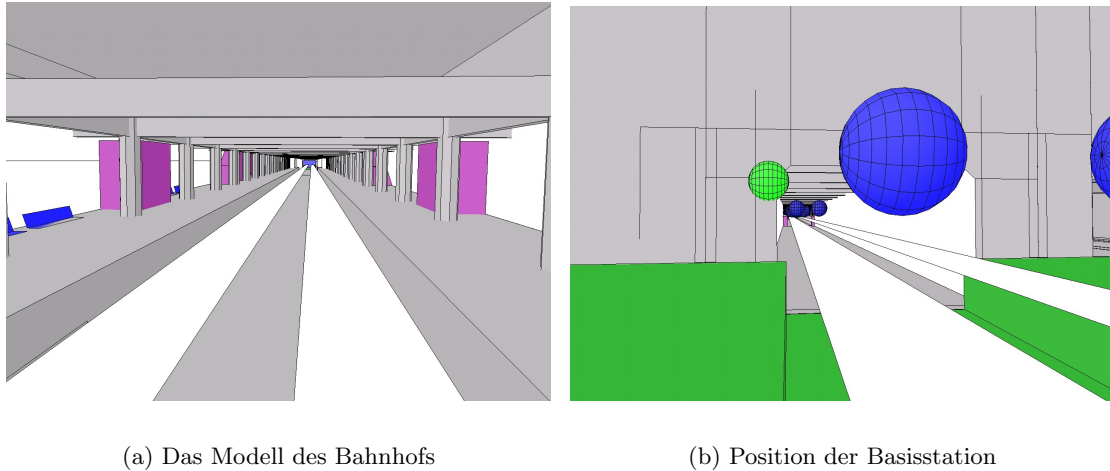


(a) BER Linke Fahrspur



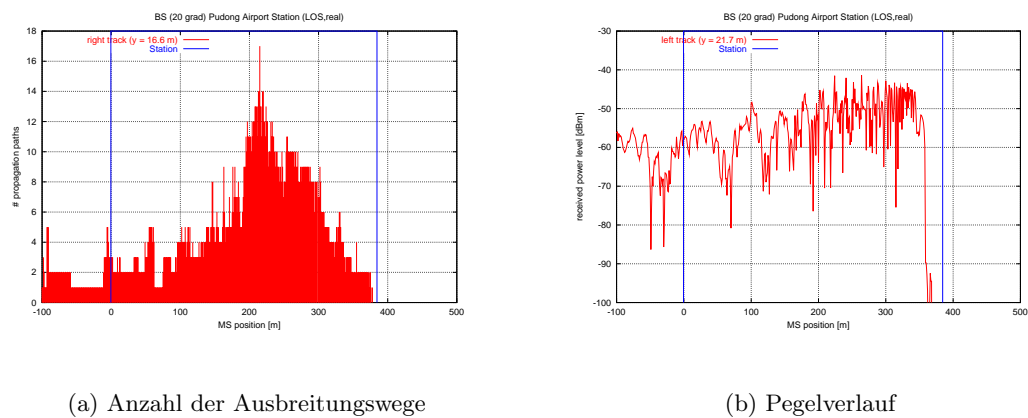
(b) BER Rechte Fahrspur

**Bild 8.11:** Verlauf der zu erwartenden Bitfehlerrate ohne Ausnutzung der Diversität als Funktion der Position der Mobilstation entlang beider Fahrbahnen.



**Bild 8.12:** Ein 3D Modell des Bahnhofs Pudong Station am Internationalen Flughafen Pudong in Schanghai. Der Bahnhof besteht aus einer Halle mit einem anschließenden Tunnel.

gewährleistet. Diese Anordnung wird bevorzugt eingesetzt, da sich das Fahrzeug in der unmittelbaren Umgebung der Basisstation aufgrund der Richtcharakteristik der Antennen in einer 'toten Zone' befindet, in der keine Kommunikation stattfinden kann. Der negative Einfluss dieser Funklöcher in der Umgebung der Basisstationen auf die Kommunikation mit dem Fahrzeug wird durch die Makrodiversität unterdrückt, dennoch ist ein schnelles Durchfahren dieser Bereiche von Vorteil. Die im Weiteren diskutierten Er-

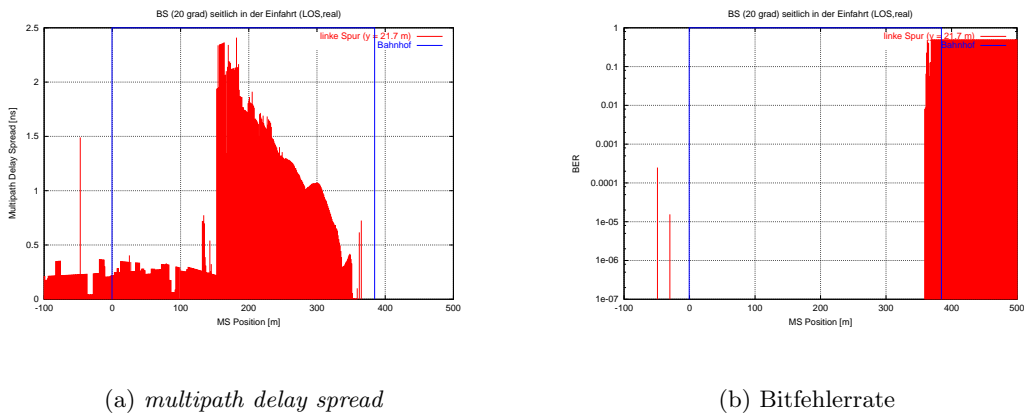


**Bild 8.13:** Die Anzahl der Ausbreitungswege und der Pegelverlauf für die linke Fahrspur. Die Bahnhofsgrenzen sind blau eingezeichnet.

gebnisse wurden durch die Simulation der Wellenausbreitung innerhalb des Bahnhofs für eine Position der Basisstation an der seitlichen Wand im hinteren Bereich des Tunnels gewonnen (Bild 8.12b). Diese Position stellte eine Möglichkeit unter einer Vielzahl anderer Alternativen dar, die ebenfalls im Rahmen des Projekts untersucht wurden. Der Verlauf der empfangenen Leistung über der Fahrbahn im Bereich des Bahnhofs und des Tunnels

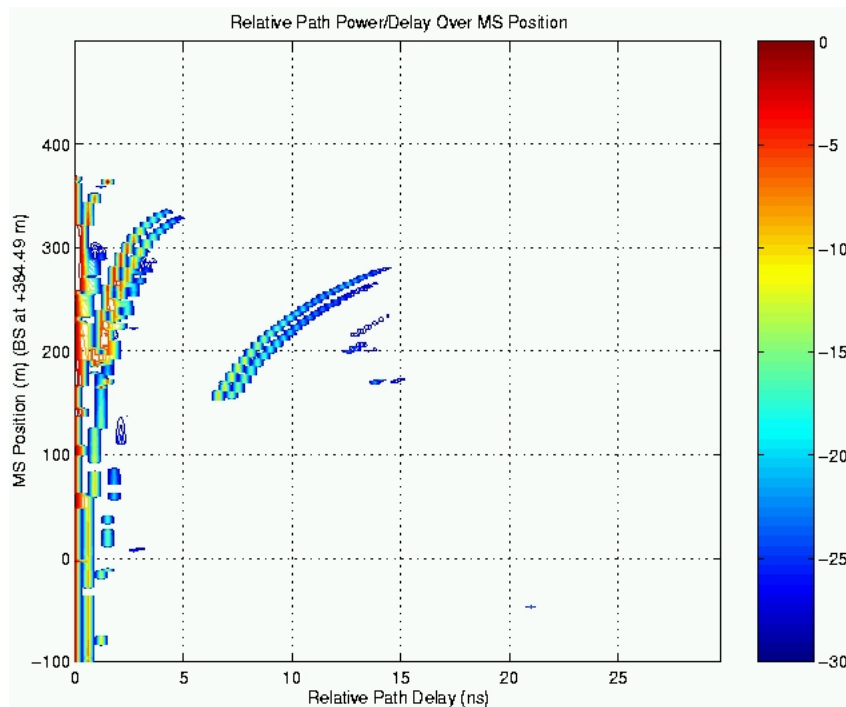
ist im Bild 8.13b dargestellt. Der Pegel zeigt einen ausgeprägten schnellen Schwund, der durch die extreme Mehrwegeausbreitung innerhalb des Tunnelbereichs verursacht wird. Diese Mehrwegeausbreitung wird an der Anzahl der detektierten Ausbreitungswege verdeutlicht, die ebenfalls im Bild 8.13a dargestellt ist.

Die frequenzselektiven Eigenschaften des Kanals werden durch die Kanalimpulsantwort und die daraus resultierende statistische Größe des MPDS. Diese Eigenschaften bestimmen entscheidend die Bandbreite des Kommunikationssystems (GENG und WIESBECK, 1998). Der Verlauf des MPDS entlang der Fahrspur wurde ebenfalls ausgewertet und ist im Bild 8.14a dargestellt. Es stellt sich ein für die Tunnelumgebung typischer Verlauf der Größe ein. Da sich durch die enge Umgebung innerhalb des Tunnels die Umwege der reflektierten Strahlen relativ wenig voneinander unterscheiden, nimmt der Spreizfaktor relativ kleine Werte an. Die resultierende Bitfehlerrate ohne Betrachtung des Einflusses der Diversität ist im Bild 8.14b zu sehen. Trotz der starken Mehrwegeausbreitung wird die geforderte Bitfehlerrate im Bahnhofsinneren erreicht. Der starke Anstieg der Bitfehlerrate in der Umgebung der Basisstation bei 380 Meter ist auf die 'tote Zone' in der Umgebung der Basisstation zurückzuführen. Im rückwärtigen Bereich der Basisstation ist aufgrund der Charakteristik der Antennen keine Kommunikation möglich und daher wird dort die maximale Bitfehlerrate erreicht.

(a) *multipath delay spread*(b) *Bitfehlerrate*

**Bild 8.14:** Der Verlauf des MPDS und der Bitfehlerrate über der linken Spur im Bahnhofsbereich.

Eine besonders anschauliche Darstellung der Ausbreitungsverhältnisse in der Form einer Kanalimpulsantwort ist im Bild 8.15 dargestellt. In diesem Bild ist auf der Y Achse die Position des Fahrzeugs aufgetragen. Die Position der Basisstation beträgt in diesem Bild 380 Meter. Der Bereich des Bahnhofs erstreckt sich von 0 bis etwa 180 Meter, wobei der restliche Bereich durch den Tunnel eingenommen wird. Die Antenne der Basisstation ist in diesem Diagramm nach unten in Richtung kleinerer Y Werte orientiert. Auf der X Achse ist die Differenz der Zeitverzögerung des Ausbreitungspfades relativ zu der LOS Komponente aufgetragen. Die farblich kodierten Werte auf der Z Achse dokumentieren die zusätzliche Dämpfung des Ausbreitungspfades relativ zu dem LOS Pfad. Wie dem Bild zu entnehmen ist, wird die LOS Bedingung durch den Bahnhof und den Tunnel hindurch gewährleistet.



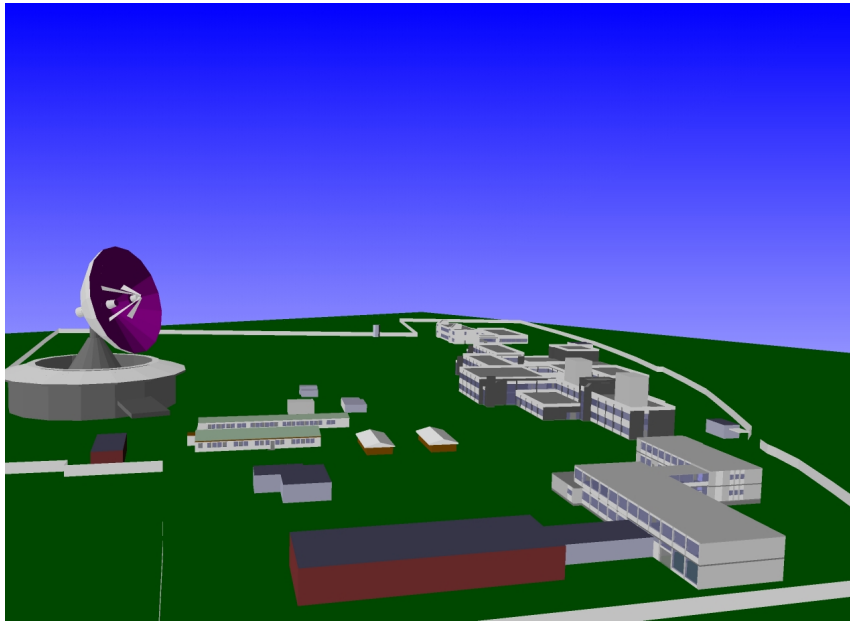
**Bild 8.15:** Die berechnete Kanalimpulsantwort in Abhängigkeit der Position des Fahrzeugs.

## 8.5 Berechnung der Streustrahlung eines Radars

Eine weitere Anwendung des in dieser Arbeit entwickelten Ray Tracing Verfahrens bestand in der Berechnung der Feldstärkeverteilung in der Nähe eines Radarsystems. Die durch eine Nebenkeule des Erregersystems induzierte elektrische Feldstärke sollte dabei sowohl im gesamten Areal als auch in den einzelnen Gebäuden für verschiedene Positionen des Radars untersucht werden. Das 3D Modell des Szenario ist im Bild 8.16 zu sehen. Zu einer besseren Übersichtlichkeit wurde das Radom in dem Bild nicht dargestellt. Das dreidimensionale Modell der Geometrie des Szenarios wurde in mehreren aufeinanderfolgenden Schritten erstellt. In einem ersten Schritt wurden zwei Luftbildaufnahmen des Landesvermessungsamtes NRW, die aus zwei verschiedenen gut dokumentierten Perspektiven gemacht wurden, durch eine semiautomatische Luftbilddauswertung in ein dreidimensionales Modell des Geländes umgewandelt. In diesem Modell war die Lage der Gebäude zueinander mit einer Genauigkeit von etwa 20 Centimeter festgelegt, die äußeren Gebäudestrukturen waren jedoch durch verschiedene Gebäudeprimitive dargestellt. In einem zweiten Schritt wurden die Gebäudeprimitive durch detaillierte dreidimensionale Modelle ersetzt. Im letzten Schritt wurden den Objekten Materialparameter zugewiesen. Die verwendeten Modelle der Gebäude wurden von dem Auftraggeber mit Hilfe des Programms AutoCAD erstellt und anschließend in ein 3D Format umgewandelt. Diese automatische Prozedur berücksichtigte die physikalische Ausdehnung der Wände und machte den Einsatz des im Kapitel 2 vorgestellten einfachen Wandmodells unmöglich. Aus diesem Grund wurde das ebenfalls im Kapitel 2 alternative Modell für ausgedehnte Wandstrukturen entwickelt, getestet und in den Simulator integriert. Die Voraussetzung der gleichartigen Orientierung der Polygone in den Außenraum war durch die automatische Generierung des 3D Modells erfüllt.

Das Cassegrain-System wurde nach vorhandenen Plänen modelliert. Die Untersuchung der Streustrahlung des Radarsystems sollte im L-Band durchgeführt werden. Die nur leicht





**Bild 8.16:** Das 3D Modell des Szenarios.

ausgeprägte Topographie des Geländes sollte bei der Untersuchung vernachlässigt werden. Für eine Berechnung der Feldstärke auf dem Gelände wurden Empfänger in einem  $1 \times 1\text{m}^2$  Raster in 1.5m Höhe über dem Boden angeordnet. Es wurden Berechnungen für 32400 verschiedene Radarpositionen<sup>1</sup> durchgeführt und die gefundenen Ausbreitungspfade anhand der bekannten Richtcharakteristik des Erregers sowie der Sendeleistung in komplexe Feldstärkevektoren umgerechnet. Da sich der Erreger des Radarsystems nicht im Drehzentrum des Radars befindet und bei der Wellenausbreitung die Rotation des gesamten Spiegelsystems (und hier insbesondere die Abschattung durch den Subreflektor) berücksichtigt werden sollte, musste für jede Winkeleinstellung eine vollständige Neuberechnung der Ausbreitungssituation erfolgen.

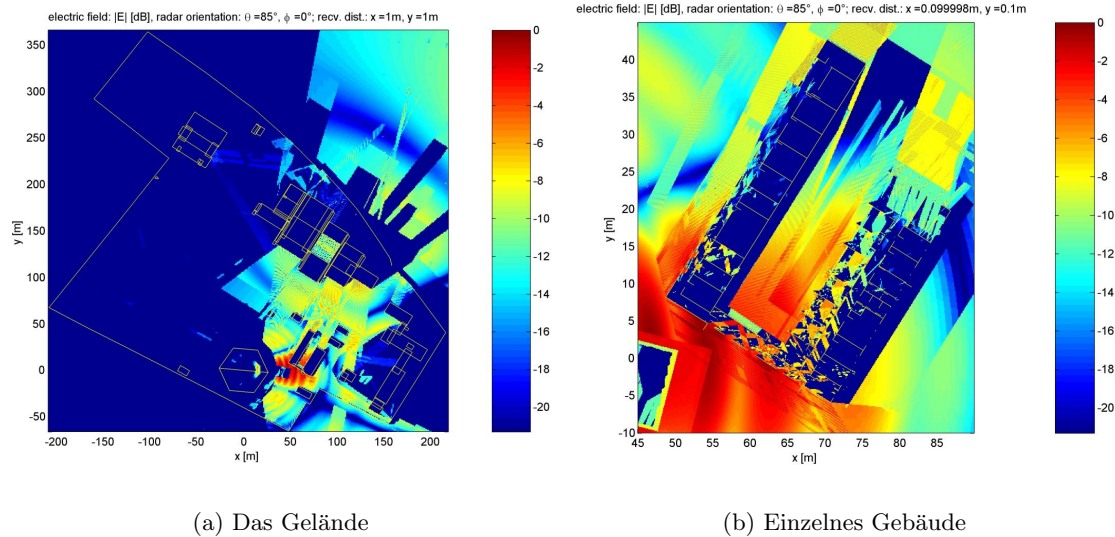
Das Ergebnis der Berechnung für eine Winkelposition ist im Bild 8.17a dargestellt. Die Einzelbilder der Ergebnisse wurden anschließend in Bildsequenzen umgewandelt, welche anschaulich die Entwicklung der Feldstärke auf dem Gelände während einer Drehung des Radars darstellten.

Die Berechnung der Wellenausbreitung auf dem Gelände wurde mit einem SunFire System des Rechenzentrums der RWTH Aachen durchgeführt. Für die Berechnung wurden 50 UltraSparc Prozessoren eingesetzt, die das Problem nach etwa zwei Tagen gelöst haben.

Für die Berechnung der Feldstärke innerhalb der Gebäude wurden die Empfänger in einem  $10 \times 10\text{cm}^2$  Raster in 1.5m Höhe über dem Boden angeordnet und die Simulation erneut für alle relevanten Positionen des Radars durchgeführt. Die Schrittweite war nachwievor 1 Grad in Azimut und Elevation, es wurden aber nur die Winkelbereiche betrachtet, in denen das Gebäude einen relevanten Beitrag des Feldes aufwies. Die Einzelbilder wurden ebenfalls für eine weitere Auswertung zu Bildsequenzen zusammengefasst.

Die Berechnung der gesamten Feldverteilung auf dem Gelände und in den Gebäuden, in denen sowohl die Wellenausbreitung komplexer als auch die Empfängerdichte höher war, veranschlagte etwa 1500 CPU-Tage auf dem SunFire System. Dabei wurde eine Kombination aus Parallelverarbeitung auf Auftragsebene und SMP Parallelisierung eingesetzt. Die Bearbeitung der parallelen Aufträge wurde mit der SUN GridEngine und selbstge-

<sup>1</sup>Die gesamte Hemisphäre wurde in 1 Grad Schritten in Elevation und Azimut abgetastet.



**Bild 8.17:** Verlauf der elektrischen Feldstärke über dem Gelände und in einzelnen Gebäuden.

schriebenen Skripten realisiert. Jeder Auftrag stellte hierbei eine Position des Radars dar. Seine Abarbeitung bestand aus dem Einlesen der Geometrie, der Konstruktion und der Optimierung des BSP Baumes sowie der Lösung des Ausbreitungsproblems mit der kombinierten Cone Tracing/Spiegelungsmethode. Die Optimierung des BSP Baumes musste für jeden Auftrag neu erfolgen, da sich die Geometrie des Szenario durch die Drehung des Cassegrain-Systems geändert hat. Sie wurde durch eine schnelle zufällige Abtastung des Zustandsraumes (Kapitel 5 Seite 72) realisiert. Die SMP Parallelisierung umfasste die parallele Simulation der Wellenausbreitung innerhalb eines Auftrags. Die Bearbeitung eines Auftrags lag bei etwa 10–15 Minuten, wobei maximal 4 Prozessoren pro Auftrag eingesetzt wurden.

Die Simulationsergebnisse wurden mit Messungen der Feldstärke an einigen Punkten innerhalb und außerhalb von Gebäuden verglichen. Der Vergleich der simulierten und gemessenen Werte ist in der folgenden Tabelle 8.1 aufgeführt.

Messpunkt	Messung [V/m]	Simulation [V/m]	Differenz [V/m]	Differenz [dB]
A (outdoor)	375.6	287.6	88	2.3
B (outdoor)	90.4	92.6	-2.2	-0.2
C (indoor)	120	98	22	1.8
D (indoor)	87	100	-13	-1.2

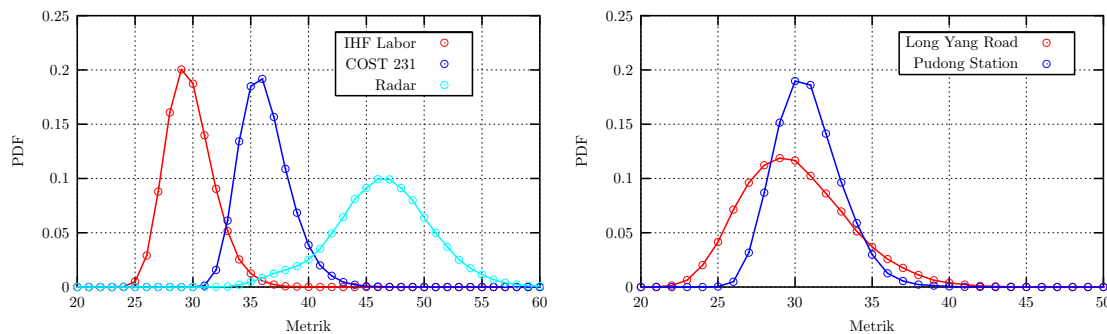
**Tabelle 8.1:** Vergleich der Messung mit der Simulation der Feldstärkeverteilung.

Bei der Bewertung der Messwerte ist zu beachten, dass die Messwerte mit der Richtcharakteristik der Empfangantenne gewichtet sind, während für die Simulation eine isotrope Richtcharakteristik angenommen wurde. Die Messantenne wurde stets auf den Sender ausgerichtet, so dass die dominanten Ausbreitungspfade durch die Messung erfasst wurden. Teilweise ergeben sich jedoch größere Unterschiede in der Nähe von Gebäuden und vor allem in der Nähe von Fahrzeugen und anderen beweglichen Gegenständen auf dem

Gelände, die in dem der Simulation zugrundeliegenden Modell nicht berücksichtigt wurden. Insgesamt ergibt sich eine gute Übereinstimmung der Messung mit der Simulation für die gewählten Messpunkte.

## 8.6 Statistische Eigenschaften des BSP Baumes

Zuletzt sollen noch die statistischen Eigenschaften der BSP Bäume der behandelten Modelle betrachtet werden. Diese Eigenschaften haben, wie bereits im Kapitel 5 auf Seite 72 erwähnt, eine direkte Auswirkung auf die Effizienz des eingesetzten Optimierungsverfahrens. Es wurden die Verteilungsdichten der maximalen Baumhöhen für jedes Modell ermittelt, die gleichzeitig die Metrik der Optimierungsverfahren darstellt. Die Ergebnisse sind im Bild 8.18a und b zu sehen. Für die Bestimmung der Verteilungsdichte eines Modells wurden jeweils 100000 zufällige BSP Bäume der Modellgeometrie konstruiert, ihre Metrik gemessen und in das Histogramm eingetragen. Das Ergebnis aus Bild 8.18 zeigt



**Bild 8.18:** Statistische Verteilung der Häufigkeit der maximalen Baumhöhe für jeweils 100000 verschiedene, zufällig gewählte Bäume. Die lokal eng begrenzte Verteilungsdichtefunktion der Metrik ermöglicht eine schnelle Optimierung mit der *Random Sampling* Methode aus Kapitel 5.

eine deutliche Ähnlichkeit der Verteilungsdichten. Der Grund für genau diesen Verlauf der Verteilungsdichtefunktion ist noch ungeklärt. Die Form wird scheinbar nicht entscheidend durch die Art des Szenario beeinflusst, da der Verlauf für alle Szenarien, unabhängig ob es sich um *Indoor* oder *Outdoor* Szenarien handelt, eine sehr große Ähnlichkeit aufweist. Die Form der Verteilungsdichte deutet darauf hin, dass die gewählte Metrik sehr viele lokale Minima aufweist, jedoch die Wahrscheinlichkeit, das globale Minimum durch Zufall zu finden, verschwindend klein ist. Die untere Grenze der Metrik kann leicht aus der Anzahl der Polygone abgeschätzt werden. Sie ist durch den Logarithmus dualis dieser Anzahl gegeben. Diese Grenze ist jedoch aufgrund der Überschneidung der Polygone nur in Ausnahmefällen erreichbar. Nur ein Algorithmus zur Konstruktion global-optimaler BSP Bäume würde eine Aussage darüber ermöglichen, wie weit das globale Minimum der Metrik von dem Maximum der Verteilungsdichte liegt.

Die Kenntniss des Verlaufs der Verteilungsdichtefunktion kann jedoch einfach parametrisiert und zur Bewertung der Optimierungsverfahren eingesetzt werden. So ist es denkbar, dass während einer Optimierung die Verteilungsdichte der Metrik aufgestellt und parametrisiert wird. Anschließend kann, nachdem eine ausreichende statistische Sicherheit erreicht wurde, anhand der parametrischen Form die Güte des gefundenen lokalen

Minimums bewertet und die Optimierung ggf. abgebrochen werden. Die Form der Verteilungsdichte der Metrik deutet darauf hin, dass bereits die einfache Optimierung durch eine zufällige Abtastung des Zustandsraumes (*Random Sampling* Methode) nach einer kurzen Zeit zu sehr guten Ergebnissen führen kann.

## 8.7 Zusammenfassung

In diesem Kapitel wurde das im Rahmen dieser Arbeit entwickelte Ray Tracing Verfahren anhand von exakten Lösungen und mit Messungen validiert und einige Einsatzbereiche des Verfahrens vorgestellt. Zu diesen Einsatzbereichen gehörte die Berechnung der Wellenausbreitung innerhalb eines Labors, die Planung der Funkversorgung in der Bahnhöfen der Transrapid Strecke in Shanghai sowie eine Untersuchung der Feldstärkeverteilung in der Umgebung einer Radarstation. Insbesondere die letzte Untersuchung wurde für eine Vielzahl der möglichen Positionen der Radarantenne durchgeführt und machte den Einsatz der Parallelverarbeitung zur fristgerechten Lösung des Problems unabdingbar. Abschließend wurden die statistischen Eigenschaften der BSP Bäume der Modelle untersucht und ihre Auswirkung auf die Optimierungsalgorithmen abgeschätzt.



---

## Zusammenfassung und Ausblick

In dieser Arbeit wurde ein adaptives hierarchisches Ray Tracing Verfahren zur parallelen Berechnung der Wellenausbreitung in komplexen Szenarien vorgestellt. Durch die Beschreibung der Szenarien mit polygonalen Flächen sowie die Beschränkung auf die wichtigsten Ausbreitungsmechanismen, die Reflexion und Transmission durch mehrschichtige Wände, Beugung nach der *Uniform Theory of Diffraction* (UTD) sowie die eindimensionale Streuung an rauen Oberflächen, liegt der anvisierte Anwendungsbereich des Verfahrens vor allem im Bereich des Mobilfunks oder allgemein im Bereich der Wellenausbreitung in urbanen Szenarien.

Das Wellenausbreitungsmodell geht von der asymptotischen Näherung der Wellengleichung aus und beschreibt das elektromagnetische Feld durch die Überlagerung planarer transversal-elektromagnetischer Wellenfronten, deren Ausbreitungswege das Fermatsche Gesetz erfüllen und mit Hilfe der Gesetze der geometrischen Optik bzw. der vereinheitlichten Theorie der Beugung bestimmt werden. Die strahlenoptische Lösung eines komplexen Wellenausbreitungsproblems erfordert eine schnelle Lokalisierung des Objekts, das als nächstes von dem Strahl getroffen wird. Diese Kernfunktion, die für jeden Teilstrahl aufgerufen wird, ist entscheidend für die Effizienz und somit für die Geschwindigkeit der Berechnung. Durch die Beschränkung auf eine Beschreibung der Geometrie mit Hilfe von planaren Polygonen sowie das eingesetzte Wandmodell, welches Mehrfachreflexionen innerhalb der Wand exakt berücksichtigt, wird eine hierarchische Darstellung der Geometrie in der Form eines modifizierten räumlichen binären Partitionierungsbaumes möglich. Basierend auf der Konstruktion optimierter räumlicher binärer Partitionierungsbäume konnte eine sehr effiziente Suche nach den von einem Strahl getroffenen Polygonen realisiert werden. Ebenso konnte auf diesem Weg eine deutliche Beschleunigung der Lokalisation der getroffenen Empfänger erreicht werden.

Voraussetzung für eine schnelle Suche nach Ausbreitungspfaden in einem komplexen Szenario ist eine Optimierung des räumlichen binären Partitionierungsbaumes. Durch eine Optimierung wird die Höhe des binären Partitionierungsbaumes minimiert, so dass eine Schnittpunktberechnung mit einer minimalen Anzahl Operationen durchgeführt werden kann. Die Optimierung des Baumes ist vergleichbar mit einer Balancierung eines binären Suchbaumes. Da für räumliche binäre Partitionierungsbäume bislang keine Rebalancierungsalgorithmen bekannt sind, wurden im Rahmen dieser Arbeit Verfahren zur Konstruktion näherungsweise optimaler binärer Partitionierungsbäume implementiert und bewertet. Diese Verfahren setzen entweder Heuristiken oder andere Optimierungsverfahren wie die Simulated Annealing Methode, Genetische Algorithmen oder Random Sampling Methoden ein, um die Reihenfolge der Polygone für die Konstruktion des optimalen Partitionierungsbaumes einer bestimmten Geometrie festzulegen. Basierend auf der Annahme, dass ein balancierter binärer Partitionierungsbaum die minimale Baumhöhe besitzt, wurde die Baumhöhe als Metrik für eine Bewertung der binären Partitionierungsbäume eingesetzt. Durch eine statistische Auswertung der Verteilung der Metrik für eine Vielzahl zufällig gewählter Permutationen der Konstruktionsreihenfolge des Baumes wurde die Vermutung geäußert, dass die Verteilungsfunktion der Baumhöhe näherungsweise der Gaussverteilung entspricht. Somit kann bereits mit einigen wenigen Messungen der Metrik der Bereich der Optimierung stark eingeschränkt und ein automatisches Abbruchkriterium

für die Optimierungsschleife festgelegt werden. Die statistische Verteilung der Baumhöhe wurde anhand verschiedener Geometrien mit unterschiedlicher Anzahl der Polygone verifiziert. Das Ergebnis eines Vergleichs der Optimierungsmethoden deutet darauf hin, dass bereits das einfachste *Closed Loop* Optimierungsverfahren – die Random Sampling Methode, die auf einer zufälligen Abtastung des Zustandsraumes basiert – aufgrund der besonderen Verteilungsfunktion der Metrik, die Konstruktion eines näherungsweise optimalen Baumes erlaubt.

Ein weiterer Schritt in Richtung zu einer schnellen Lösung des Wellenausbreitungsproblems liegt in der Reduktion der Strahlenzahl, die für das Auffinden von allen relevanten Ausbreitungspfaden benötigten werden. Diese Reduktion der Strahlendichte darf jedoch nicht auf Kosten der Genauigkeit vorgenommen werden. In dieser Arbeit wurde ein adaptives *Cone Tracing* Verfahren vorgestellt, das die lokale Strahlendichte automatisch in den von dem Benutzer festgelegten Grenzen an die Geometrie anpasst. Es konnte gezeigt werden, dass durch diese Anpassung sowohl die Strahlenzahl als auch die benötigte Rechenzeit deutlich reduziert werden kann. Desweiteren wurde eine neuartige Identifikation der Ausbreitungswege und somit eine schnelle Strahlendichtenormierung entwickelt und implementiert. Die Cone Tracing Methode wurde mit der *Spiegelungsmethode* kombiniert. Auf diese Weise konnte die Geschwindigkeit der Cone Tracing Methode mit der Genauigkeit der Spiegelungsmethode vereint werden. Die Spiegelung der gebeugten Strahlen kann jedoch nur in Ausnahmefällen einfach durchgeführt werden, da durch die beliebige Orientierung der Beugungskanten die genaue Position eines Beugungspunktes von der Position aller anderen Beugungspunkte entlang des Ausbreitungspfades abhängt. Das Fermatsche Integral führt in diesem Fall zu einer nichtlinearen Funktion, deren Minimum entweder mit der klassischen Newton-Raphson Methode oder mit der adaptiven Simulated Annealing Methode für eine beliebige Zahl von Unbekannten bestimmt werden kann. Die Berechnung der eindimensionalen bistatischen Streukoeffizienten wurde nach der Literatur mit Hilfe der Momentenmethode für fraktale raue Oberflächen realisiert.

Der Einsatzbereich des Verfahrens wird durch eine effiziente Ausnutzung der Parallelverarbeitung weiter ausgedehnt. Durch eine Parallelisierung auf Systemen mit einem gemeinsamen Speicher wird sowohl die Simulationszeit verkürzt als auch Grenze der simulierbaren Szenariogröße ausgeweitet. Die Parallelisierung kann besonders einfach nach dem *Master/Worker* Schema durchgeführt werden und weist selbst für eine größere Zahl von Prozessoren einen näherungsweise linearen *Speedup* auf. Bei einer Parallelisierung auf einer NUMA Architektur muss ein besonderer Wert auf die Lage der Daten im Speicher gelegt werden. Untersuchungen im Rahmen dieser Arbeit zeigen, dass die Implementierung des Strahlengenerators einen wichtigen Einfluss auf die Adressierung und somit auf eine effiziente Nutzung der komplexen Hierarchie der Zwischenspeicher hat. Eine parallele Simulation auf einem System mit verteiltem Speicher ist wegen der längeren Aufsetzzeiten der Simulation erst bei Problemen mit einer sehr hohen Komplexität sinnvoll. Die Parallelisierung für diese Rechnerarchitektur erfolgte ebenfalls nach dem *Master/Worker* Prinzip, wobei die Kommunikation mit der *Message Passing Interface* (MPI) Bibliothek realisiert und mit der *Double Buffering* Technik parallel zur Berechnung durchgeführt wurde.

Das entwickelte Verfahren wurde durch einen Vergleich mit exakten Methoden und mit Messungen validiert. Es wurde zu Untersuchungen der Wellenausbreitung in verschiedenen Szenarien innerhalb und außerhalb von Gebäuden eingesetzt, unter anderem im Rahmen von Projekten zur Planung der Funkversorgung der Transrapid-Trasse in Schanghai und zur Berechnung der Streustrahlung in der Umgebung einer Radarstation.

Weitere Arbeiten können in zwei Bereiche unterteilt werden. Der erste Bereich beinhaltet die Verfeinerung des physikalischen Modells durch die Implementierung und die Validierung weiterer physikalischer Effekte, deren Berücksichtigung das Verfahren zu ei-



nem universellen Werkzeug nicht nur für die Funknetzplanung machen würde. Dazu gehört unter Anderem die Streuung der elektromagnetischen Wellen bei der Transmission durch inhomogene Wände, die mit Hilfe eines stochastischen Modells oder durch eine Kombination mit einem exakten Verfahren wie z.B. der *Finite Differences Time Domain* (FDTD) oder der *Finite Elements Methode* (FEM) realisiert werden kann. Ein weiterer wichtiger Effekt ist die Berücksichtigung der Strahlaufweitung bei der Reflexion der Wellen an elektrisch kleinen Gegenständen. In diesem Fall wirkt sich die elektrisch kleine Fläche ab einer bestimmten Entfernung wie eine Antenne mit einer typischen Charakteristik auf das Feld in der Umgebung der Fläche aus. Die Antennencharakteristik wird durch die Belegung der Apertur bestimmt, die ihrerseits durch die auf die Fläche einfallenden Wellen definiert ist. Die Berechnung dieses Effektes kommt einer Kombination des Ray Tracing Verfahrens mit der Methode der physikalischen Optik gleich, die jedoch nur selektiv für die Berücksichtigung elektrisch kleiner Gegenstände eingesetzt wird. Zu den weiteren Aufgaben auf diesem Gebiet gehört ebenfalls die die Hybridisierung des Verfahrens durch eine Kombination mit anderen Lösungsmethoden für die Teilbereiche der Geometrie, die nicht durch einen strahlenoptischen Ansatz alleine behandelt werden können. Dazu gehören resonante Strukturen, die ein besonderes frequenzselektives Verhalten aufweisen. Dadurch könnte der Anwendungsbereich der Methode bis hin zu der Berechnung eines Radar-Rückstreuquerschnitts erweitert werden, wenn die Modellgeometrie bis zu einer ausreichenden Genauigkeit durch Polygone angenähert werden kann, oder das Verfahren um gekrümmte Flächen erweitert wird.

Die zweite Kategorie betrifft die Verbesserung der algorithmischen Implementierung des Verfahrens. Hier wäre vor allem der Übergang von der relativ einfachen adaptiven Cone Tracing Methode zu dem komplizierteren adaptiven *Triangular Beam Tracing* Verfahren zu erwähnen. Durch diesen Übergang wäre eine Strahlendichtenormierung nicht mehr von Nöten, da sich die pyramidenförmigen Strahlenbereiche nicht überlappen und sich somit die Empfänger nur innerhalb eines Strahls befinden können. Zu den weiteren Vorteilen der adaptiven *Triangular Beam Tracing* Methode gehört neben dem Vermeiden von Mehrfachdetektionen bei den Empfängertests auch das Anti-Aliasing in Kantennähe sowie eine adaptive Minimierung der Strahlendichte in Bereichen mit einer homogenen Struktur. Der Übergang zu Strahlen mit einem beliebigen polygonalen Querschnitt würde eine weitere Verringerung der Strahlendichte mit sich führen. Gleichzeitig wäre dieser Übergang auch mit einem erheblichen Aufwand bei der algorithmischen Auswertung komplexer polygonaler Flächen verbunden.



## ABBILDUNGSVERZEICHNIS

---

2.1	Definition der Hauptkrümmungsradien der Wellenfront . . . . .	8
2.2	Definition der Koordinatensysteme . . . . .	9
2.3	Reflexion und Transmission einer wärmedämmendem Fensterglasscheibe . .	11
2.4	Definition des geometrischen Wandmodells . . . . .	13
2.5	Das Koordinatensystem einer Beugungskante . . . . .	15
2.6	Graphische Lösung der Gleichungen für $N^+$ und $N^-$ . . . . .	17
2.7	Zur Definition der Streuung an einer rauen Oberfläche . . . . .	18
2.8	Die Geometrie und das Koordinatensystem der Streuquelle. . . . .	20
2.9	Bistatischen Streukoeffizienten einer fraktalen Fläche . . . . .	21
2.10	Abhängigkeit der Streukoeffizienten von der fraktalen Dimension . . . . .	21
3.1	Die Strahlabschuss- und Cone Tracing Methode . . . . .	27
3.2	Die Kernfunktion des Ray Tracing Verfahrens . . . . .	29
4.1	Der binäre Partitionierungsbaum in der Ebene. . . . .	34
4.2	Die Konstruktion des modifizierten BSP Baumes . . . . .	36
4.3	Die Partitionierung eines Polygons mit einer Ebene . . . . .	37
4.4	Die Schnittpunktsuche mit Hilfe eines modifizierten BSP Baumes . . . . .	38
4.5	Die rekursive Implementierung der schnellen Schnittpunktsuche. . . . .	39
4.6	Zur Strahlaufteilung und Empfängerdetektion im BSP Baum . . . . .	41
4.7	Messung und Interpolation der Empfängerbeschleunigung . . . . .	43
4.8	Die Testgeometrie und die Beschleunigung der Empfängerdetektion . . . . .	44
4.9	Der 'Punkt in Polygon' Test . . . . .	45
4.10	Die Konstruktion eines klassischen Binärbaumes . . . . .	47
4.11	Darstellung der reellen Achse mit einem BSP Baum . . . . .	48
4.12	Durchführung einer Mengenoperation mit BSP Bäumen . . . . .	50
4.13	Partitionierung ( <i>Splitting</i> ) eines BSP Baumes in $\mathbb{R}^0$ . . . . .	51
4.14	Entfernung redundanter Information während der <i>Pruning</i> Phase . . . . .	53
4.15	Ein Beispiel einer multidimensionalen BSP-Baum Hierarchie . . . . .	54
4.16	Klassifizierung eines Punktes mit der BSP-Baum Hierarchie . . . . .	55
4.17	Die Verdeckung von Polygonen mit unterschiedlichen Prioritäten . . . . .	56
4.18	Erweiterung der BSP-Baum Hierarchie um topologische Zeiger . . . . .	57
4.19	Modell eines typischen Straßenszenario . . . . .	58
4.20	Einteilung des Straßenszenario in konvexe Zellen . . . . .	59
5.1	Das Modell des Vergleichsszenario . . . . .	64
5.2	Darstellung der Baumhöhe des nicht optimierten BSP Baumes . . . . .	65
5.3	Die Boltzmannsche Wahrscheinlichkeitsverteilungsdichte . . . . .	67
5.4	Optimierung mit der Simulated Annealing Methode . . . . .	68
5.5	Die Baumhöhe des optimierten BSP Baumes . . . . .	69
5.6	Optimierung mit einem genetischen Algorithmus . . . . .	70
5.7	Konvergenzrate des genetischen Algorithmus . . . . .	71
5.8	Ergebnis der Optimierung mit einem genetischen Algorithmus . . . . .	72
5.9	Die Wahrscheinlichkeitsverteilungsfunktion der Metrik . . . . .	73

6.1	Zur Mehrfachdetektion eines Strahls . . . . .	78
6.2	Kombination der Strahlabschussmethode mit der Spiegelungsmethode . . . . .	79
6.3	Die Verteilungsdichte eines zufallsgesteuerten Strahlengenerators . . . . .	81
6.4	Das Modell des Ikosaeders und seiner Partitionierung . . . . .	82
6.5	Die Verteilungsdichte der Strahlen des Ikosaedergenerators . . . . .	83
6.6	Die Verteilungsdichte der Strahlen eines Oktaedergenerators . . . . .	84
6.7	Die Strahlendichteverteilung des <i>generalized spiral</i> Generators . . . . .	86
6.8	Aufbau und Datendurchsatz einer Speicherhierarchie . . . . .	87
6.9	Auswirkung des Strahlengenerators auf das Speicherzugriffsverhalten . . . . .	88
6.10	Aliasing als Folge der räumlichen Unterabtastung der Geometrie. . . . .	89
6.11	Adaptive Anpassung der Strahlendichte an die Geometrie . . . . .	90
6.12	Die Strahlendichte eines adaptiven Strahlengenerators . . . . .	92
6.13	Position der Sekundärquellen an der Beugungskante . . . . .	93
6.14	Konvergenzverhalten der iterativen Lösung des Spiegelungsproblems . . . . .	98
6.15	Lösung mit der adaptiven Simulated Annealing Methode . . . . .	98
6.16	Modell der Streuung an rauen Oberflächen . . . . .	99
7.1	Vergleich der Rechnerarchitekturen . . . . .	102
7.2	Das Amdahlsche Gesetz . . . . .	105
7.3	Beschleunigung durch die Parallelisierung auf einem SMP System . . . . .	106
7.4	Die <i>Master/Worker</i> Parallelisierung . . . . .	107
7.5	Die Beschleunigung auf einem System mit verteiltem Speicher . . . . .	108
7.6	Vergleich der Setup-Zeiten . . . . .	109
7.7	Auswirkungen der Nachrichtenlänge auf den Durchsatz . . . . .	110
7.8	Beschleunigung der Optimierung auf einem SMP Rechner . . . . .	111
8.1	Verifikation des Modells mit der Momentenmethode . . . . .	113
8.2	Verifikation des Modells mit der Spiegelungsmethode . . . . .	114
8.3	Das Szenario und das in der Simulation eingesetzte 3D Modell . . . . .	115
8.4	Vergleich mit Messdaten des RUSK ATM CS . . . . .	116
8.5	Das 3D Modell eines Labors des Instituts für Hochfrequenztechnik . . . . .	117
8.6	Die Übertragungsdämpfung im Labor . . . . .	117
8.7	Anzahl der detektierten Ausbreitungspfade . . . . .	117
8.8	Die Magnetschwebetechnik Transrapid . . . . .	118
8.9	Das Modell des Bahnhofs Longyang Road Station in Schanghai . . . . .	119
8.10	Verlauf der Empfangsleistung entlang der Fahrbahnen . . . . .	120
8.11	Verlauf der zu erwartenden Bitfehlerrate . . . . .	120
8.12	Das Modell des Bahnhofs am Internationalen Flughafen in Schanghai . . . . .	121
8.13	Die Anzahl der Ausbreitungswege und der Pegelverlauf . . . . .	121
8.14	Verlauf des Multipath Delay Spread und der Bitfehlerrate . . . . .	122
8.15	Die Kanalimpulsantwort in Abhängigkeit der Position des Fahrzeugs . . . . .	123
8.16	Das Modell des Szenarios . . . . .	124
8.17	Verlauf der elektrischen Feldstärke . . . . .	125
8.18	Statistische Verteilung der Baumhöhe . . . . .	126

## TABELLENVERZEICHNIS

---

4.1	Regeln zur Vereinfachung der Mengenoperationen . . . . .	52
5.1	Vergleich verschiedener Optimierungsmethode für BSP Bäume . . . . .	74
7.1	Daten einiger SMP Systeme . . . . .	103
7.2	Daten einiger ccNUMA Systeme . . . . .	103
7.3	Daten einiger Systeme mit verteiltem Speicher . . . . .	104
8.1	Vergleich der Messung mit der Simulation der Radaranlage . . . . .	125
10.1	Verzeichnis der verwendeten Formelzeichen . . . . .	145



## GLOSSAR

---

- Aliasing Effekt** Der Aliasing Effekt entsteht durch eine Unterabtastung des Raumes mit einer nicht ausreichenden Strahlendichte. Er macht sich besonders in der Nähe von Kanten bemerkbar, indem Kanten- und Schattengrenzen eine treppenförmige Struktur aufweisen. Der Effekt kann durch eine adaptive oder stochastische Überabtastung reduziert oder ganz beseitigt werden (*Anti-Aliasing*).
- Beowulf Cluster** Eine Klasse von *Distributed Memory* Parallelrechnern, die vor allem aus gewöhnlichen *off-the-shelf* Bauteilen wie handelsüblichen PCs und Netzwerkkomponenten besonders preiswert aufgebaut werden können. Kennzeichnend für einen Beowulf Cluster ist unter anderem der Einsatz des freien Betriebssystems *Linux*.
- Bounding Object** Ein *Bounding Object* dient der Beschleunigung der Schnittpunkt-berechnung eines Strahls mit einem komplexen Objekt. Zu diesem Zweck wird um das komplexe Objekt ein einfaches Objekt, meistens in der Form einer Kugel, eines Zylinders oder einer Box, mit dem eine Schnittpunkt-berechnung schnell durchgeführt werden kann. Erst dann, wenn das *Bounding Object* von dem Strahl getroffen wurde, muss die Schnittpunkt-berechnung mit dem realen Objekt im Inneren des *Bounding Objects* durchgeführt werden. Die Form des *Bounding Object* muss möglichst eng das reale Objekt umschließen, damit die Anzahl der benötigten aufwendigen Schnittpunkt-berechnungen mit dem komplexen realen Objekt minimal wird.
- BSP Baum** Der Binary Space Partitioning (BSP) Baum entsteht durch eine rekursive Aufteilung des Raumes durch Partitionierungsebenen in Teilräume. Da eine Partitionierungsebene einen Raum in zwei neue Teilräume aufteilt, entsteht so eine binäre Baumstruktur. Die Partitionierungsebenen befinden sich in den Knoten des Baumes, die Teilräume sind in den Baumblättern enthalten.



**Cache**

Mit *Cache* wird ein schneller Zwischenspeicher bezeichnet, der sich zwischen dem schnellen Prozessor und dem langsamen Hauptspeicher befindet. Die Aufgabe des *Cache* ist es, häufig benutzte Daten und Instruktionen im schnellen Speicher in der Nähe des Prozessors zu halten und somit die Anzahl der Zugriffe auf den viel langsameren Hauptspeicher zu minimieren. Ein *Cache* besteht in der Regel aus einer Hierarchie von L1, L2 und L3 *Cache*-Speichern, wobei die Größe des Speichers zum Hauptspeicher hin zunimmt, Zugriffsgeschwindigkeit dagegen abnimmt. Der L1 und der L2 *Cache* sind meistens im Prozessor integriert, der L3 *Cache* wird häufig nur im Serverbereich eingesetzt und ist über eine besondere Schnittstelle an den Prozessor angeschlossen. Der *Cache* hilft, das sog. *von Neumann Bottleneck* zu lösen. Damit ist der Flaschenhals der Speicherschnittstelle gemeint, die charakteristisch für die von Neumann Rechnerarchitektur ist. Die meisten der heute eingesetzten Rechner besitzen eine von Neumann Rechnerarchitektur.

**Cone Tracing**

Die *Cone Tracing* Methode stellt eine Abwandlung der klassischen *Ray Tracing* Methode dar. Während bei der klassischen Methode linienförmige Strahlen und ausgedehnte Empfänger betrachtet werden, setzt die *Cone Tracing* Methode konusförmige Strahlen und punktförmige Empfänger ein. Beide Methoden sind physikalisch äquivalent, die *Cone Tracing* Methode besitzt jedoch Vorteile bei der Implementierung eines schnellen Strahlverfolgungsalgorithmus, da eine eindeutige Zuordnung der punktförmigen Empfänger zu den geometrischen Teilräumen möglich ist.

**Distributed Memory**

Eine Parallelrechner-Architektur, bei der die Parallelrechner aus vielen autonomen Rechenknoten mit lokalem Speicher bestehen und untereinander über ein Kommunikationsnetzwerk Daten austauschen (*Message Passing*), um das gestellte Problem gemeinsam zu lösen.

**Divide And Conquer**

Das *Divide And Conquer* (teile und herrsche) Prinzip stellt ein wichtiges theoretisches Paradigma bei der Entwicklung schneller Algorithmen dar. Das komplexe Problem wird rekursiv in einfachere Teilprobleme aufgeteilt, die wesentlich schneller parallel abgearbeitet werden können als das eigentliche komplexe Problem. Das Prinzip kommt ebenfalls zum Einsatz bei der Organisation großer Datenmengen für schnelle Suchoperationen, wobei die Datenbasis nach einem Ordnungskriterium in eine Hierarchie disjunkter Teilmengen aufgeteilt wird. Typisches Beispiel stellen die binären Suchbäume dar. Algorithmen nach diesem Prinzip können im besten Fall eine logarithmische Komplexität aufweisen.

- Double Buffering** Mit *Double Buffering* wird eine Methode zur parallelen Ausführung der Berechnung und der Kommunikation in einem *Worker Prozess* bezeichnet. In der Regel wird für die Speicherung der Parameter und der Ergebnisse eines Teilproblems ein Datenpuffer verwendet, der von dem *Master* Prozess mit den Parametern der Teilaufgabe gefüllt und anschließend von dem *Worker* Prozess abgearbeitet wird. Durch diese sequenzielle Vorgehensweise muss der *Worker* Prozess warten, bis der *Master* Prozess den Puffer gefüllt hat. Diese Wartezeit kann bei einem langsamen Kommunikationsnetzwerk bestimmend sein für die gesamte Rechenzeit des Problems. Durch die Hinzunahme eines zweiten Datenpuffers kann dieser parallel zu der Berechnung gefüllt werden. Anschließend werden die Puffer einfach umgeschaltet und die Berechnung kann ohne Wartezeiten fortgesetzt werden. Eine effektive Nutzung dieser Technik setzt eine dedizierte Kommunikationshardware voraus, die Daten in den Speicher übertragen kann, ohne den Prozessor zu belasten.
- Fermatsches Prinzip** Das Fermatsche Prinzip besagt, dass sich Wellen von allen möglichen Ausbreitungspfaden stets auf dem Pfad mit der kürzesten optischen Länge bzw. der kleinsten Laufzeit ausbreiten. Die Erweiterung des Fermatschen Prinzips fordert eine stationäre Laufzeit des Ausbreitungspfades gegenüber einer transversalen Variation der Pfades. Somit kann die Laufzeit ebenfalls maximal sein oder einen Sattelpunkt annehmen. Das Fermatsche Prinzip stellt einen Spezialfall des allgemeineren Prinzips der kleinsten Wirkung dar.
- Genetic Algorithm** Genetischen Algorithmen bilden eine Klasse iterativer Optimierungsverfahren, die das Darwinsche Prinzip nachbilden, um ein komplexes Optimierungsproblem näherungsweise zu lösen. Die Anwendung eines genetischen Algorithmus setzt eine geeignete Darstellung des Parameterraumes durch Chromosomen (wobei einzelne Parameter den Genen entsprechen) sowie die Definition einer Fitness-Funktion voraus. Im iterationalen Genetic Algorithm Modell wird die erste Generation durch die Wahl einer bestimmten Anzahl zufälliger Chromosomen erzeugt und mit der Fitness-Funktion bewertet. Die besten Chromosomen dieser Generation werden durch Kreuzung *Crossover* für den Aufbau der nächsten Generation eingesetzt. Durch Mutation werden zufällig zwei beliebige Gene eines Chromosoms vertauscht. So wird eine zu schnelle Konvergenz des genetischen Pools verhindert.
- Hash Funktion** Die *Hash* Funktion projiziert eine Menge mit sehr vielen Elementen auf eine Menge mit einer begrenzten Anzahl der Elemente. Die *Hash* Funktion ist nicht reversibel. Ein typisches Beispiel einer *Hash* Funktion ist die Cyclic Redundancy Checksum (CRC), die häufig zur Prüfung von Datenpaketen in Kommunikationsnetzwerken eingesetzt wird. *Hash* Funktionen werden ebenfalls zur Generierung von Schlüsseln für schnelle Datenbankabfragen eingesetzt (*Hash Tabelle*).

<b>Hash Tabelle</b>	Ein sehr schnelles Suchverfahren mit Hilfe einer Datenbank, die über eine <i>Hash Funktion</i> adressiert wird.
<b>Kaustik</b>	Die Brennpunkte oder Brennlinsen eines Strahls an den Stellen, an denen einer oder beide Hauptkrümmungsradien der Wellenfront zu Null werden.
<b>k-d Tree</b>	Eine Variante des <i>Octree</i> . Jeder Knoten des <i>k-d Trees</i> enthält nur eine Partitionierungsebene, die parallel zu einer der drei Hauptebenen orientiert ist.
<b>Linux</b>	Ein freies, unter der GNU <i>General Public License</i> (GPL) im Quelltext publiziertes, UNIX-ähnliches Betriebssystem, das von Linus Torvalds und einer weltweiten Entwicklergemeinschaft im Internet entwickelt wird.
<b>Load Balancing</b>	Mit <i>Load Balancing</i> ist der Lastausgleich zwischen den Prozessoren des Parallelrechners gemeint. Es ist in der Regel die Aufgabe des <i>Master Prozesses</i> , dafür zu sorgen, dass die Last der Berechnung gleichmäßig auf alle Prozessoren verteilt wird.
<b>LOS</b>	Line-Of-Sight (LOS) Ausbreitung. Bezeichnung der besonderen Eigenschaften von Kommunikationskanälen, bei denen eine direkte Sichtverbindung zwischen dem Sender und dem Empfänger besteht.
<b>Master/Worker</b>	Dieser Begriff bezeichnet einen Ansatz bei der Parallelisierung eines Programms. Dieser Ansatz kann immer dann eingesetzt werden, wenn das Problem in kleinere, voneinander unabhängige Teilaufgaben aufgeteilt werden kann. Der zentrale <i>Master</i> Prozess verteilt die Teilaufgaben auf die parallel laufenden <i>Worker</i> Prozesse. Diese arbeiten die Teilaufgabe ab und fordern eine neue Aufgabe bei dem <i>Master</i> Prozess an. Dadurch wird ein automatisches <i>Load Balancing</i> der Prozessoren erreicht.
<b>Message Passing</b>	Ein theoretisches Paradigma bei der Entwicklung paralleler Programme für <i>Distributed Memory</i> Parallelrechner. Die Prozesse des parallelen Algorithmus werden auf die verschiedenen Rechenknoten des Parallelrechners verteilt. Sie tauschen während der Berechnung Daten in der Form von Datenpaketen ( <i>Messages</i> ) über das Kommunikationsnetzwerk, um gemeinsam das Problem zu lösen. Es stehen verschiedene freie Bibliotheken zur Verfügung, die bekanntesten unter ihnen sind die <i>Parallel Virtual Machine</i> (PVM) sowie der neue internationale Standard <i>Message Passing Library</i> (MPI).

- Meta-Hierarchie** Eine Meta-Hierarchie abstrahiert die Repräsentation einer Geometrie und definiert eine eindeutige Schnittstelle zum *Ray Tracing* Algorithmus. Durch die Abstrahierung des geometrischen Objekts wird das *Ray Tracing* Verfahren unabhängig von der Geometrie, da die Eigenheiten der geometrischen Objekte hinter der funktionalen Schnittstelle des Meta-Objekts im Sinne der objektorientierten Programmierung versteckt werden.
- Monte Carlo** Eine stochastische Methode zur Lösung komplexer Probleme wie zum Beispiel partieller Differentialgleichungen oder Integralgleichungssysteme, die nur schwer oder gar nicht analytisch gelöst werden können. Die Monte Carlo Methode generiert eine geeignete Zufallszahlenfolge und beobachtet, welcher Anteil dieser Zahlen eine bestimmte Eigenschaft besitzt. Als Beispiel kann die Ausbreitung von optischen Photonen im Raum dienen, die den Maxwell'schen Gleichungen gehorcht. Die Lichtquelle generiert Photonen nach einer vorgegebenen Zufallszahlenverteilung. Die Bahnen der Photonen werden durch den Raum verfolgt und ihre Interaktion mit der Materie durch geeignete statistische Funktionen beschrieben. Der Anteil der Photonen, die den/die Detektor(en) erreichen, ist die statistische Lösung der Maxwell'schen Gleichung für die gegebene Geometrie. Der Name der Methode geht auf S. Ulam (1946) zurück in Anlehnung an die zufällige Natur eines Glücksspiels.
- NLOS** No-Line-Of-Sight (NLOS) Ausbreitung. Bezeichnung der besonderen Eigenschaften von Kommunikationskanälen, bei denen keine direkte Sichtverbindung zwischen dem Sender und dem Empfänger besteht. Die Kanaleigenschaften werden vor allem durch Reflexion, Transmission und die Beugung bestimmt.
- Octree** Eine voxelartige Baumstruktur im dreidimensionalen Raum. Jeder Knoten des Baumes teilt den Raum in acht neue Teilräume auf (daher der Name). Für diese Aufteilung muss lediglich der Schnittpunkt der Partitionierungsebenen, die stets parallel zu den Hauptebenen liegen, angegeben werden. Jeder Teilraum kann wiederum rekursiv aufgeteilt werden.
- Platonische Körper** Als Platonische Körper werden alle regulären konvexen Polyeder bezeichnet, deren Oberfläche aus demselben regelmäßigen Vieleck bestehen und in jeder Ecke des Körpers gleich viele dieser Vielecke zusammenstoßen. Ihr Name geht auf Platon zurück, der erstmals zeigen konnte, dass es nur fünf solche reguläre konvexe Polyeder gibt: Tetraeder, Hexaeder, Oktaeder, Dodekaeder und Ikosaeder.
- Ray Launching** Ein *Ray Tracing* Verfahren, bei dem die Strahlen von der Quelle zu den Empfängern verfolgt werden. Auch *Forward Ray Tracing* genannt. Gegensatz zu dem häufig im Bereich der Computergraphik eingesetzten *Backward Ray Tracing*, bei dem Strahlen vom Auge des Beobachters durch die Pixel der Projektionsebene verfolgt werden.

- Ray Tracing** Allgemeine Bezeichnung für eine Klasse von Simulationsmethoden, die auf der asymptotischen Näherung der Wellengleichung für unendlich hohe Frequenzen beruhen und diese durch die geometrische Berechnung und Verfolgung von Partikelbahnen in der gegebenen Geometrie lösen.
- Rope Tree** Unter *Rope Trees* wird eine Erweiterung der Struktur eines *BSP Baumes* um zusätzliche Daten verstanden, die eine Traversierung des Baumes in horizontaler Richtung ermöglichen und somit eine schnellere Schnittpunktsuche erlauben.
- Simulated Annealing** Die *Simulated Annealing* Methode gehört ebenso wie die *genetischen Algorithmen* in die Klasse der iterativen Optimierungsverfahren. Die Methode wurde zuerst von Metropolis zur Simulation von Phasenübergängen in der Thermodynamik eingesetzt. Die Methode bildet den Erstarrungsprozess einer Schmelze nach. Die Gesamtenergie des Systems wird minimiert, indem einzelne Atome umgruppiert werden. Entsprechend der Energiedifferenz werden jedoch mit einer Wahrscheinlichkeit, die durch die Boltzmann-Verteilung gegeben ist, auch energetisch ungünstige Atompositionen akzeptiert. Dadurch wird vermieden, dass Atome in lokalen Minima der Energiefunktion gefangen bleiben. Die Übergangswahrscheinlichkeit sinkt mit der Temperatur der Schmelze, die nach einem vorgegebenen *Annealing Schedule* erniedrigt wird. Berühmtheit erreichte die *Simulated Annealing* Methode durch die Lösung des *Travelling Salesman Problems*, das stellvertretend für eine Kategorie von Pfadminimierungsproblemen steht, die eine große Rolle in vielen Bereichen der Technik spielen (z.B. bei der Platzierung und Kontaktierung von Chips).
- Source Imaging** Die Spiegelungsmethode ersetzt die Geometrie des Raumes durch eine Anzahl geeignet gewählter Spiegelquellen. Jede Spiegelquelle repräsentiert hierbei einen eindeutigen Ausbreitungsweg vom Sender zum Empfänger. Eine typische Eigenschaft dieser Methode ist die genaue Berechnung des Ausbreitungsweges zwischen der Position des Senders und dem punktförmigen Empfänger, so dass alle gefundenen Strahlenwege beide Orte exakt verbinden. Dadurch können die genauen elektrischen Parameter der Wellenfront am Ort des Empfängers berechnet werden. Diese Eigenschaft ist bei der Berechnung des schnellen Schwunds bei der Funknetzplanung oder bei der Auralisierung in der Akustik von entscheidender Bedeutung. Der wichtigste Nachteil der Spiegelungsmethode liegt in dem mit der Anzahl der Spiegelquellen exponentiell ansteigenden Rechen- und Speicheraufwand. Dieser Nachteil kann durch eine Kombination der Spiegelungsmethode mit der Strahlabschussmethode behoben werden.

- Strahlenbaum** Der Strahlenbaum entsteht durch die Verfolgung eines Strahls durch das Szenario. Der einfache Strahl, der von einem Sender ausgesandt wird, spaltet sich bei der ersten Interaktion mit der Geometrie in mehrere neue Strahlen auf, die jedes für sich weiterverfolgt werden. Durch weitere Interaktionen kann jedes einzelne wieder aufgeteilt werden. Diese baumförmige Struktur wird als der Strahlenbaum bezeichnet. Die Knoten des Baumes bezeichnen die Wechselwirkungen des einfallenden Strahls mit der Geometrie, die Kanten des Baumes sind die durch die Wechselwirkung neu erzeugten Teilstrahlen.
- Thread** *Threads* werden auch als *light-weight processes* bezeichnet. Es handelt sich um Funktionen, die innerhalb eines Programmkontext parallel ausgeführt werden können. Durch die Ausführung im selben Kontext bleibt den Funktionen der Zugriff auf den gemeinsamen Datenbereich des Programms erhalten. Der Zugriff auf die gemeinsamen Daten muss jedoch mit Semaphoren oder *Mutual Exclusive Access* (Mutex) Variablen synchronisiert werden. Ein anderer wesentlicher Vorteil von *Threads* gegenüber einem gewöhnlichen Prozess besteht im deutlich geringeren Aufwand beim Umschalten des Kontext im Betriebssystem.
- Topologischer Baum** Unter diesem Begriff wird die Erweiterung des *BSP Baumes* um topologische Information verstanden. Topologische Daten beschreiben den Zusammenhang geometrischer Objekte, so zum Beispiel die Zugehörigkeit von Flächen zu Kanten und von Kanten zu Eckpunkten einer Geometrie. Die topologische Information wird für die Lokalisierung physikalischer Effekte benötigt.
- TREAPS** Ein Treap stellt eine Hybridstruktur aus einem Binärbaum und einem Heap (daher der Name Treap = Tree + Heap) dar. Jeder Knoten des Baumes besteht aus zwei Komponenten: Der Schlüsselkomponente und der Prioritätskomponente. Die Schlüsselkomponente genügt der Suchbaumbedingung und die Prioritätskomponente genügt der Heapbedingung. Da die Prioritätskomponenten zufällig gewählt werden, werden schlechte Konstruktionsfolgen wie sie bei natürlichen Binärbäumen auftreten, vermieden. Treaps gehören in die Klasse der 'randomisierten Suchbäume'.
- UTD** Unified Theory of Diffraction. Die vereinheitlichte Theorie der Beugung ergänzt die Koeffizienten der geometrische Theorie der Beugung um Korrekturterme höherer Ordnung, die die Unstetigkeiten des geometrisch gebeugten Feldes an der Reflexions- und Schattengrenze korrigieren.





LISTE DER FORMELZEICHEN

Tabelle 10.1: Verzeichnis der verwendeten Formelzeichen

Formelzeichen	Bereich	Siehe Seite	Bedeutung
<b>Kapitel 2: Physikalische Grundlagen</b>			
$\nabla$	<b>Operatoren</b>	6	Der Nabla Operator
$\partial_t$		6	Die Partialableitung nach der Zeit
$\Delta$		7	Der Laplace Operator
$\bar{1}$		7	Die Einheitsdyade
$\Re$		12	Der Realteil einer komplexen Größe
$\Im$		12	Der Imaginärteil einer komplexen Größe
$  \vec{A}  $		94	Die Länge des Vektors $\vec{A}$
$[x]$		83	Die obere Gaussklammer
$\mathcal{O}()$		12	Die Komplexität
$L(C)$	<b>Fermatsches Prinzip</b>	5	Die optische Länge des Pfades $C$
$n(\vec{r})$		5	Der ortsabhängige Brechungsindex
$\vec{r}_a, \vec{r}_b$		5	Der Start- und Endpunkt eines Strahls
$A$		5	Die Amplitude der skalaren Welle
$k_0$		5	Die Wellenzahl der Welle im freien Raum
$\vec{D}$	<b>Maxwellsche Gleichungen</b>	6	Dielektrische Verschiebungsdichte
$\vec{E}$		6	Der Vektor der elektrischen Feldstärke
$\vec{H}$		6	Der Vektor der magnetischen Feldstärke
$\vec{B}$		6	Der Vektor der magnetischen Induktion
$\vec{J}$		6	Der Vektor der elektrischen Stromdichte
$\rho$		6	Die Ladungsdichte
$\epsilon$		6	Die tensorielle Permittivität des Materials
$\mu$		6	Die tensorielle Permeabilität des Materials
$\sigma$		6	Die tensorielle Leitfähigkeit des Materials
$t$		6	Die Zeit
$f$		6	Die Frequenz
$\omega$	6	Die Kreisfrequenz	
$\Psi(\vec{r})$	<b>Luneberg-Kline Reihenentwicklung</b>	7	Die ortsabhängige Phasenfunktion
$\vec{E}_m(\vec{r})$		7	Die vektorielle Amplitude des $m$ -ten Terms
$\vec{R}(s)$		7	Der parametrische Strahlenweg im homogenen isotropen Medium
$\underline{E}_v, \underline{E}_h$	<b>Übertragung</b>	8	Komplexe Amplitude der vertikalen und der horizontalen Polarisation der Welle
$\underline{\mathbf{S}} = \begin{bmatrix} \underline{S}_{vv} & \underline{S}_{vh} \\ \underline{S}_{hv} & \underline{S}_{hh} \end{bmatrix}$		8	Komplexe polarimetrische Übertragungsmatrix eines Ausbreitungsweges
$\underline{\mathbf{S}}_i$		10	Die Streumatrix der $i$ -ten Grenzschicht
$\mathbf{T}_i$		10	Die zugehörige Koordinatentransformation

Fortsetzung auf der nächsten Seite

Tabelle 10.1: Verzeichnis der verwendeten Formelzeichen

Fortsetzung von der vorhergehenden Seite

Formelzeichen	Bereich	Siehe Seite	Bedeutung
$(r, \theta, \phi)$	<b>Koordinaten</b>	9	Sphärische Koordinaten
$\vec{e}_r, \vec{e}_\theta, \vec{e}_\phi$		9	Rechtwinkliges Dreibein zu $(\theta, \phi)$
$\vec{e}_d, \vec{e}_s, \vec{e}_p$		9	Lokales Koordinatensystem der Einfallsebene
$\vec{e}_N$	<b>Grenzschicht</b>	9,14	Der Normalenvektor der Grenzfläche
$\vec{E}_s, \vec{E}_p$		9	Komponenten der elektrischen Feldstärke in den lokalen Koordinaten der Einfallsebene
$\underline{E}_p, \underline{E}_s$		10	Zugehörige komplexe Wellenamplituden
$\varphi_i, \varphi_r, \varphi_t$		10	Der Einfallswinkel der einfallenden Welle und die Ausfallwinkel der reflektierten und der transmittierten Welle.
$\underline{R}_p, \underline{R}_s$	<b>Wandparameter</b>	10	Die komplexen Reflexionsfaktoren für beide Feldkomponenten in den lokalen Koordinaten der Einfallsebene
$\underline{T}_p, \underline{T}_s$		11	Die komplexen Transmissionsfaktoren beider Feldkomponenten in den lokalen Koordinaten der Einfallsebene
$d_n$		12	Die Dicke der $n$ -ten dielektrischen Schicht
$\epsilon_{r,n}$		12	Die komplexe relative Permeabilität der $n$ -ten dielektrischen Schicht
$\mu_{r,n}$		12	Die komplexe relative Permittivität der $n$ -ten dielektrischen Schicht
$\gamma_n$		12	Die komplexe Ausbreitungskonstante in der $n$ -ten dielektrischen Schicht
$k_m$		11	Die Wellenzahl in der $m$ -ten Schicht
$\underline{Z}_m$		11	Der Feldwellenwiderstand der $m$ -ten Schicht
$N$		12	Die Anzahl der dielektrischen Schichten der Wand
$\underline{P}_m, \underline{R}_m$		12	Die sich in der $m$ -ten Schicht ausbreitenden Wellenkomponenten, mit $m = 0, 1, \dots, N + 1$
$\vec{e}_E$	<b>Beugung</b>	15	Der Tangentialvektor der Beugungskante
$\vec{e}_N$		15	Der Normalenvektor zur Beugungskante
$\vec{e}_B$		15	Der Bi-Normalenvektor der Beugungskante
$\eta$		16	Der normierte Öffnungswinkel der Kante (Definition siehe Bild 2.5 auf Seite 15)
$\beta$		16	Der Elevationswinkel der einfallenden Welle
$\varphi'$		16	Der Azimutwinkel der einfallenden Welle
$\varphi$		16	Der Azimutwinkel der gebeugten Welle
$L = r \sin^2 \beta$		16	Länge des Ausbreitungspfades einer lokal ebenen Welle
$a^\pm(\varphi \pm \varphi')$		16	Der Winkelabstand des Beobachterpunktes von der Reflexions- und Schattengrenze
$N^\pm$		16	Faktoren der verschiedenen Beugungsbereiche (siehe Bild 2.6 auf Seite 17)
$\underline{D}_s, \underline{D}_h$		16	Die komplexen Beugungskoeffizienten nach der vereinheitlichten Theorie der Beugung
$F(x)$		16	Das Fresnelsche Integral.

Fortsetzung auf der nächsten Seite

**Tabelle 10.1:** Verzeichnis der verwendeten Formelzeichen*Fortsetzung von der vorhergehenden Seite*

Formelzeichen	Bereich	Siehe Seite	Bedeutung
$\Psi$		16	Winkel zwischen dem einfallenden Strahl und dem Normalenvektor der Keilfläche auf der strahlzugewandten Seite
$\underline{R}_p, \underline{R}_s$		16	Die komplexen Reflexionsfaktoren für beide Feldkomponenten in den lokalen Koordinaten der Einfallsebene
$\Delta h$		17	Die Standardabweichung der normalverteilten lokalen Rauigkeit der Keiloberfläche
$\underline{R}_s^0, \underline{R}_h^0$		16	Die Fresnelschen Reflexionskoeffizienten einer nicht ideal leitenden, rauen Beugungskante
$\Delta\phi$	<b>Streuung</b>	18	Phasendifferenz der gestreuten Wellen.
$\delta$		18	Höhenschwankungen der rauen Oberfläche
$\theta_i$		18	Einfallswinkel der einfallenden Welle
$k_0$		18	Die Wellenzahl einer Welle im Vakuum
$\lambda_0$		18	Die Wellenlänge einer Welle im Vakuum
$\sigma$		18	Die Oberflächenrauigkeit
$\underline{S}_{pq}$		19	Komplexe Komponenten der Streumatrix $\underline{\mathbf{S}}$
$\underline{S}_{pq}^{coh}$		19	Kohärente Komponente der Streumatrix $\underline{\mathbf{S}}$
$\underline{S}_{pq}^{inc}$		19	Inkohärente Komponente der Streumatrix $\underline{\mathbf{S}}$
$\underline{R}_{p,s}^{mod}$		19	Modifizierte Fresnelsche Reflexionskoeffizienten einer rauen Oberfläche
$d_s$		20	Die fraktale Dimension
$k_u, k_l$		20	Die höchste und die niedrigste Wellenzahl im Spektrum der fraktalen rauen Oberfläche
$\alpha$		20	Das Verhältnis der beiden Grenzfrequenzen des Spektrums der fraktalen rauen Oberfläche
$N$		20	Die Anzahl der spektralen Komponenten der fraktalen rauen Oberfläche
$\Xi$	20	Die bistatischen Streukoeffizienten der fraktalen rauen Oberfläche	

**Kapitel 3: Algorithmische Grundlagen**

$f$	<b>Freiraumdämpfung</b>	28	Die Frequenz
$d$		28	Die Freiraumlänge des Ausbreitungsweges
$G_t$		28	Der Gewinn der Sendeantenne
$G_r$		28	Der Gewinn der Empfangsantenne
$A$		28	Die Dämpfung des Ausbreitungsweges

**Kapitel 4: Konstruktion binärer Partitionierungsbäume**

$\vec{x}$	<b>Hyperebene</b>	33	Ortsvektor eines Punktes in $\mathbb{R}^N$
$\vec{a}$		33	Der Normalenvektor der Hyperebene
$\vec{p}$		33	Position eines Punktes in der Hyperebene
$d$		33	Projektionsmaß der Ebene entspricht dem Skalarprodukt des Normalenvektors mit einem beliebigen Punkt in der Ebene

*Fortsetzung auf der nächsten Seite*

Tabelle 10.1: Verzeichnis der verwendeten Formelzeichen

Fortsetzung von der vorhergehenden Seite

Formelzeichen	Bereich	Siehe Seite	Bedeutung
$N$		33	Dimension des Raumes
$\vec{e}_N$	<b>Partitionierung</b>	37	Der Normalenvektor der Ebene
$\vec{O}$		37	Ein Bezugspunkt in der Ebene
$\vec{P}_i$		37	Ortsvektoren der Polygoneckpunkte
$d_i$		37	Abstände der Polygoneckpunkte von der Partitionierungsebene
$\vec{P}$	<b>Schnittpunktsuche</b>	41, 41	Der Anfangspunkt des Teilstrahls
$\vec{e}_D$		41, 41	Der Richtungsvektor des Teilstrahls
$\vec{e}_N$		41	Der Normalenvektor der Ebene
$p$		41	Entfernung des Punktes $\vec{P}$ von der Ebene
$\vec{E}$	<b>Empfängerdetektion</b>	41	Die Position des Empfängers
$\delta$		41	Projektion des Empfängers auf den Strahl
$\vec{x}$		41	Der kürzeste Differenzvektor zur Strahlachse
$\alpha$		42	Der Öffnungswinkel des Strahlkonus
$l_P$		42	Die Länge des Strahls vom Sender bis zum Anfangspunkt des Teilstrahls
$r_p$		42	Der Radius des Strahls im Anfangspunkt $\vec{P}$ des Teilstrahls
$A$	<b>Beschleunigung</b>	43	Amplitude des Gauss-Terms
$\delta$		43	Koeffizient der Gaussverteilung
$B$		43	Amplitude des polynomialen Terms
$\alpha$		43	Exponent des polynomialen Terms
$S_i$	<b>Mengentheoretische BSP Bäume</b>	47	Menge aller Punkte im Teilraum $i$
$E_i$		47	Menge des Partitionierers im Teilraum $i$
$S_{i+1}^+, S_{i+1}^-$		47	Menge aller Punkte der Teilräume $i + 1$ auf beiden Seiten des Partitionierers $E_i$
$\forall X \in S_i$		47	Für alle Elemente aus der Menge $S_i$ ...
$x_i$		47	Diskrete Partitionierer der Menge
$\emptyset$		47	Die leere Menge
$\cap$		47	Die Schnittmenge
$\cup$		47	Die Vereinigungsmenge
<b>Kapitel 5: Optimierung binärer Partitionierungsbäume</b>			
$T$	<b>Metrik</b>	62	Ein Teilbaum des BSP Baumes
$T^\pm$		62	Der positive/negative Teilbaum
$E_{cost}[T]$		62	Die Metrik des Teilbaumes $T$
$p^\pm$		62	Wahrscheinlichkeitsmodell der Operation
$r$		63	Der zum Knoten $T$ gehörende Raum
$r^\pm$		63	Der positive/negative Teilraum von $r$
$V(r)$		63	Das Volumen des Teilraumes $r$
$p^0$		63	Die Wahrscheinlichkeit eines Punktes in der Partitionierungsebene
$M$		63	Das Meta-Objekt.

Fortsetzung auf der nächsten Seite

Tabelle 10.1: Verzeichnis der verwendeten Formelzeichen

Fortsetzung von der vorhergehenden Seite

Formelzeichen	Bereich	Siehe Seite	Bedeutung	
$k$	<b>Simulated Annealing</b>	66	Die Boltzmann-Konstante	
$T$		66	Die Temperatur	
$E$		66	Die Energie	
$p(E)$		66	Die Boltzmannsche Verteilungsfunktion	
$\Delta E$		67	Energiedifferenz der Zustände $n + 1$ und $n$	
$T_0$		67	Die Anfangstemperatur des Systems	
$T_n$		67	Die Systemtemperatur nach $n$ Schritten	
$D$		68	Der Zustandsraum der $N!$ Permutationen	
$X_0 \in D$		68	Der Anfangszustand des Systems	
$X_n \in D$		68	Der Zustand des Systems nach $n$ Schritten	
$P_0$	<b>Genetische Algorithmen</b>	70	Die Anfangspopulation	
$P_n$		70	Die $n$ -ten Population	
$X_k \in D$		70	Ein Zustand aus dem Zustandsraum $D$	
$M(X_k)$		70	Die Metrik des $k$ -ten Zustands $X_k$	
$F_k = M(X_k)$		70	Die Fitness Funktion des $k$ -ten Zustands	
<b>Kapitel 6: Adaptives Ray Tracing</b>				
$(\phi, \theta)$	<b>Stochastische Strahlengenerierung</b>	80	Die Abschusswinkel des Strahls	
$\mathcal{U}(-1, +1)$		80	Zufallszahlengenerator mit einer konstanten Verteilung über dem Intervall $(-1, +1)$	
$\alpha$		80	Der Öffnungswinkel eines konischen Strahls	
$p(\alpha)$		80	Die Verteilungsdichte der Öffnungswinkel	
$\gamma$		80	Anteil der Strahlen mit $\alpha < \alpha_0(N)$	
$N$		80	Die Anzahl der Strahlen	
$(x_i, y_i)$		<b>Platonische Körper</b>	82	Die Eckpunkte einer Fläche des Ikosaeders
$(x_0, y_0)$			83	Mittelpunkt des umschriebenen Kreises
$r$	83		Radius des umschriebenen Kreises	
$D$	83		Die Rekursionstiefe des Generators	
$F$	83		Der Formfaktor des Platonischen Körpers	
$\kappa$	83		Die resultierende Strahlendichte	
$\lceil x \rceil$	83		Die obere Gaussklammer	
$V(r)$	<b>Generalized Spiral Points</b>	84	Das Potential	
$r^\beta$		82	Die Entfernungsabhängigkeit des Potentials	
$-2 < \beta < 2$		82	Exponent des entfernungsabhängigen Terms	
$h_k$		85	Der Generatorterm	
$\Delta\phi_k$		85	Der Winkelinkrement	
$(\theta_k, \phi_k)$		85	Die Winkelkoordinaten des $k$ -ten Strahls	
$d_N$		85	Der maximale Durchmesser des Kreises	
$\vec{P}, \vec{D}$		<b>Adaptive Strahlengenerierung</b>	90	Die Position und Richtung einer Kante
$t$	90		Der Kantenparameter	
$\vec{X}(t)$	90		Position des Schnittpunktes mit der Kante	
$\vec{A}$	90		Der Axialvektor des Strahls	

Fortsetzung auf der nächsten Seite

Tabelle 10.1: Verzeichnis der verwendeten Formelzeichen

Fortsetzung von der vorhergehenden Seite

Formelzeichen	Bereich	Siehe Seite	Bedeutung
$\theta$		90	Der Öffnungswinkel des Strahlkonus
$\vec{V}$		90	Ein Punkt auf der Kegeloberfläche
$d_i, d$	<b>Spiegelung der Beugungsquellen</b>	94	Länge eines Teilstrahls und die Gesamtlänge
$\vec{D}_i$		94	Der Vektor des Teilstrahls
$d_i = \ \vec{D}_i\ $		94	Die Länge des Teilstrahls
$(\vec{P}_i, \vec{E}_i)$		94	Lokale Koordinaten der $i$ -ten Kante
$\vec{T}, \vec{R}$		94	Positionen des Senders und des Empfängers
$N$		94	Die Anzahl der Beugungskanten
$l_i$		94	Unbekannte Position des Beugungspunktes im Koordinatensystem der $i$ -ten Kante
$\vec{l}$		94	Die $N$ -dimensionale Unbekannte
$f_i(l_{i-1}, l_i, l_{i+1})$		94	Erste partielle Ableitung der Funktion $d(\vec{l})$
$\beta_i$		94	Winkel des einfallenden Strahls zur Kante
$\beta_o$	94	Winkel des gebeugten Strahls zur Kante	
$\vec{l}_0$		95	Der Startvektor der Newton Iteration
$\vec{l}_n$		95	Der Lösungsvektor nach $n$ Iterationen
<b>J</b>		95	Die Jacobi-Matrix
<b>A</b>		95	Die Prädiktionier-Matrix
<b>Kapitel 7: Paralleles Ray Tracing</b>			
$N$	<b>Amdahlsches Gesetz</b>	105	Die Anzahl der Prozessoren
$T$		105	Die gesamte Laufzeit des Programms
$T_s$		105	Der sequentielle Anteil der Laufzeit $T$
$T_p$		105	Der parallele Anteil der Laufzeit $T$
$T_N$		105	Die Laufzeit auf $N$ Prozessoren
$T_0 = T$		105	Die Laufzeit auf einem Prozessor
$\beta(N)$		105	Die Beschleunigung auf $N$ Prozessoren
$\gamma = T_s/T$		105	Anteil des sequentiellen Programmteils an der gesamten Laufzeit
$n$	<b>Pipeline Theorie</b>	109	Die Länge der Nachricht in Bytes
$\tau$		109	Zykluszeit des Kommunikationsnetzwerks
$T_{setup} = s\tau$		109	Die Aufsetz-Zeit der Kommunikation
$T$		109	Die gesamte Zeit der Kommunikation
$T_{unit}$		109	Die Übertragungszeit für ein Byte
$r$		109	Der Durchsatz des Kommunikationskanals
$n_{1/2}$		109	Die Länge einer Nachricht, bei der die Pipeline den halben maximalen Durchsatz erreicht
$r_\infty$		109	Der maximale Durchsatz der Pipeline bei einer unendlich langen Nachricht

## ABKÜRZUNGSVERZEICHNIS

---

BER	<u>B</u> it <u>E</u> rror <u>R</u> atio
BS	<u>B</u> asisstation
BREP	<u>B</u> oundary <u>R</u> epresentation
BSP	<u>B</u> inary <u>S</u> pace <u>P</u> artitioning
ccNUMA	<u>C</u> ache- <u>C</u> oherent <u>N</u> on- <u>U</u> niform <u>M</u> emory <u>A</u> ccess
CS	<u>C</u> hannel <u>S</u> ounder
COST	European <u>C</u> ooperation in the Field of <u>S</u> cientific and <u>T</u> echnical Research
CPU	<u>C</u> entral <u>P</u> rocessing <u>U</u> nit
CPFSK	<u>C</u> ontinuous <u>P</u> hase <u>F</u> requency <u>S</u> hift <u>K</u> eing
CSG	<u>C</u> ombinatorial <u>S</u> olid <u>G</u> eometry
CRC	<u>C</u> yclic <u>R</u> edundancy <u>C</u> hecksum
CW	<u>C</u> ontinuous <u>W</u> ave
DECT	<u>D</u> igital <u>E</u> nanced <u>C</u> ordless <u>T</u> elecommunication
DL	<u>D</u> ownlink
EMC	<u>E</u> lectromagnetic <u>C</u> ompatibility
EMVU	<u>E</u> lektromagnetische <u>V</u> erträglichkeit zur <u>U</u> mwelt
FDM	<u>F</u> requency <u>D</u> ivision <u>M</u> ultiplex
FDTD	<u>F</u> inite <u>D</u> ifferences <u>T</u> ime <u>D</u> omain
FEM	<u>F</u> inite <u>E</u> lements <u>M</u> ethode
FIFO	<u>F</u> irst <u>I</u> n <u>F</u> irst <u>O</u> ut
GO	<u>G</u> eometrische <u>O</u> ptik
GPRS	<u>G</u> eneral <u>P</u> acket <u>R</u> adio <u>S</u> ervice
GSM	<u>G</u> lobal <u>S</u> ystem for <u>M</u> obile Communication
GTD	<u>G</u> eometrical <u>T</u> heory of <u>D</u> iffraction
IEEE	<u>I</u> nstitute of <u>E</u> lectrical and <u>E</u> lectronics <u>E</u> ngineers
LAN	<u>L</u> ocal <u>A</u> rea <u>N</u> etwork
LOS	<u>L</u> ine- <u>O</u> f- <u>S</u> ight
LIFO	<u>L</u> ast- <u>I</u> n- <u>F</u> irst- <u>O</u> ut
MA	<u>M</u> aximum <u>A</u> rea
MAGLEV	<u>M</u> agnetically <u>L</u> eviated
MI	<u>M</u> inimum <u>I</u> ntersections
MIMO	<u>M</u> ultiple- <u>I</u> nput- <u>M</u> ultiple- <u>O</u> utput
MMS	<u>M</u> ultimedia <u>M</u> essage <u>S</u> ystem
MPDS	<u>M</u> ultipath <u>D</u> elay <u>S</u> pread
MPI	<u>M</u> essage <u>P</u> assing <u>I</u> nterface
MS	<u>M</u> obilstation
Mutex	<u>M</u> utual <u>E</u> xclusive Access
NLOS	<u>N</u> o- <u>L</u> ine- <u>O</u> f- <u>S</u> ight
NURBS	<u>N</u> on- <u>U</u> niform <u>R</u> ational <u>B</u> - <u>S</u> plines
NUMA	<u>N</u> on- <u>U</u> niform <u>M</u> emory <u>A</u> ccess
PDA	<u>P</u> ersonal <u>D</u> igital <u>A</u> ssistents
PO	<u>P</u> hysikalische <u>O</u> ptik
PVM	<u>P</u> arallel <u>V</u> irtual <u>M</u> achine
SIMD	<u>S</u> ingle- <u>I</u> nstruction <u>M</u> ultiple- <u>D</u> ata
SPM	<u>S</u> mall <u>P</u> erturbation <u>M</u> ethod



SMP	<u>S</u> ymmetric <u>M</u> ulti- <u>P</u> rocessor
STA	<u>S</u> pace- <u>T</u> ime <u>A</u> lgebra
TBSP	<u>T</u> opological <u>B</u> inary <u>S</u> pace <u>P</u> artitioning Trees
TEM	<u>T</u> ransversale <u>E</u> lektromagnetische Welle
TREAP	<u>T</u> ree+ <u>H</u> ea <u>P</u>
UMA	<u>U</u> niform <u>M</u> emory <u>A</u> ccess
UMTS	<u>U</u> niversal <u>M</u> obile <u>T</u> elecommunication <u>S</u> ystem
UTD	<u>U</u> niform <u>T</u> heory of <u>D</u> iffraction
WLAN	<u>W</u> ireless <u>L</u> ocal <u>A</u> rea <u>N</u> etwork
WU	<u>W</u> ork <u>U</u> nit

## LITERATURVERZEICHNIS

---

- J. B. ALLEN und D. A. BERKLEY. *Image Method for Efficiently Simulating Small-Room Acoustics*. *J. Acoust. Soc. Am.*, Bd. 65, Nr. 4, April 1979. 27
- J. AMANATIDES. *Ray Tracing with Cones*. *Computer Graphics*, Bd. 18, Nr. 3, S. 129–135, Juli 1984. 27
- G. M. AMDAHL. *Validity of the single-processor approach to achieving large scale computing capabilities*. In *AFIPS Conference Proceedings*, Band 30, S. 483–485. AFIPS Press, Reston, Va., Atlantic City, N.J., 18. – 20. April 1967. 104
- J. ARVO. *Ray tracing with meta-hierarchies*. In *SIGGRAPH '90 Advanced Topics in Ray Tracing*. ACM Press, August 1990. Course notes. 29, 31
- J. ARVO und D. KIRK. *A Survey of Ray Tracing Acceleration Techniques*. In A. GLASNER, Hrsg., *An Introduction to Ray Tracing*. Academic Press, New York, 1989. 29
- M. J. ATALLAH. *Algorithms and Theory of Computation Handbook*. CRC Press LLC, 1998. 61
- C. A. BALANIS. *Antenna Theory: Analysis and Design*. John Wiley & Sons, 1997, ISBN 0-471-59268-4. 113
- D.J. BECKER, STERLING, T.D. SAVARESE, J.E. DORBAND, U.A. RANAWAK und C.V. PACKER. *BEOWULF: A parallel workstation for scientific computation*. In *Proceedings, International Conference on Parallel Processing*, 1995. 104
- G. E. BLELLOCH und M. REID-MILLER. *Fast Set Operations Using Treaps*. In *10th Annual AMC Symposium on Parallel Algorithms and Architectures (SPAA '98)*, Puerto Vallarta, Juni 1998. 46
- BSP-FAQ. *Binary Space Partitioning: A List Of Frequently Asked Questions*, 2002. Available via anonymous FTP from rtfm.mit.edu. 36
- G. J. BURKE und A. J. POGGIO. *Numerical Electromagnetics Code (NEC) - Method of Moments*. Lawrence Livermore Laboratory, 1981. 113, 114
- T. CASSEN, K. R. SUBRAMANIAN und Z. MICHALEWICZ. *Near-Optimal Construction of Partitioning Trees Using Evolutionary Techniques*. In *Proceedings of Graphics Interface '95*, Quebec City, Mai 1995. <http://www.cs.uncc.edu/krs/publ.html> 62, 63, 71
- D. J. CICHON. *Propagation models for small and micro-cells*. In E. DAMOSSO und L. M. CORREIA, Hrsg., *COST Action 231, Final Report*. Office for official publications of the european communities, Luxembourg, 1999. 63
- J. COMBA und B. NAYLOR. *Conversion of Binary Space Partitioning Trees to Boundary Representation*. 1996. 31, 57
- L. DAVIS. *Genetic Algorithms and Simulated Annealing*. Morgan Kaufmann Publishers, 1987. 70
- G. A. DESCHAMPS. *Ray Techniques in Electromagnetics*. *Proceedings of the IEEE*, Bd. 60, Nr. 9, S. 1022–1035, September 1972. 5
- D. DIDASCALOU, F. KÜCHEN und W. WIESBECK. *Ein neuartiges Normierungsverfahren für die strahlenoptische Wellenausbreitungsmodellierung in beliebig geformten Tunneln*. *Frequenz*, Bd. 53, Nr. 9 – 10, S. 182–188, 1999. 78
- J. E. DIETERT. *Channel Model for Mobile Communications Systems with Adaptive Antennas*. Dissertation, Institut für Hochfrequenztechnik, RWTH Aachen, Juni 2001. 115
- K. DOWD und C.R. SEVERANCE. *High Performance Computing*. O'Reilly & Associates, 2. Auflage, 1998. 86, 103

- I. A. DRUMM und Y. W. LAM. *The Adaptive Beam Tracing Algorithm*. In *IOA Proceedings, Auditoria*, 1999. 78
- R. P. FEYNMAN. *Feynman Lectures On Physics*. Addison-Wesley, 1970. 5, 94
- W. FISCHER. *Radio network planning for the Maglev-Train Transrapid*. In *URSI F Conference*, Garmisch-Patenkirchen, Februar 2002a. 118
- W. FISCHER. *Radio network planning for track-bound vehicles*. Dissertation, Institut für Hochfrequenztechnik, RWTH Aachen, Februar 2002b. 118, 119
- T. FRACH und W. FISCHER. *Construction of near-optimal meta-hierarchies for fast parallel ray-tracing propagation simulations with evolutionary techniques*. In *Proceedings of the Poster 2002 Conference, Prague*, 2002. 30, 63, 119
- T. FRACH, W. FISCHER und B. REMBOLD. *Fast parallel ray-tracing simulator*. In *Proceedings of the Millenium Conference on Antennas & Propagation, Davos*, 2000. 31, 57
- F. G. FRIEDLANDER. *Geometrical optics and Maxwell's equations*. *Proc. Cambridge Phil. Soc.*, Bd. 43, S. 284, 1947. 6
- H. FUCHS, Z. M. KEDEM und B. F. NAYLOR. *On Visible Surface Generation by a Priori Tree Structures*. *Computer Graphics*, Bd. 14, Nr. 3, S. 124–133, 1980. 2, 30
- T. FUNKHOUSER, I. CARLBOM, G. ELKO, G. PINGALI, M. SONDHI und J. WEST. *A Beam Tracing Approach to Acoustic Modeling for Interactive Virtual Environments*. In *ACM Computer Graphics, Proceedings, SIGGRAPH'98*, 1998. <http://www.cs.princeton.edu/gfx/papers/funkhouser98bta.pdf> 31, 59, 78
- S. GEMAN und D. GEMAN. *Stochastic Relaxation, Gibbs distribution and the Bayesian restoration in images*. *IEEE Trans. Patt. Anal. Mac. Int.*, Bd. 6, Nr. 6, S. 721–741, 1984. 67
- N. GENG und W. WIESBECK. *Planungsmethoden für die Mobilkommunikation*. Springer, 1998. 10, 18, 22, 122
- D. GOLDBERG. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989. 2, 70
- J. GOLDSMITH und J. SALMON. *Automatic creation of object hierarchies for ray tracing*. *IEEE Computer Graphics and Applications*, S. 14–20, Mai 1987. 62
- JÜRGEN GROSSKOPF. *Wellenausbreitung I/II*. Bibliographisches Institut, 1970. 28
- S. GULL, A. LASENBY und C. DORAN. *Imaginary Numbers are not Real. – the Geometric Algebra of Spacetime*. *Found. Phys.*, Bd. 23, Nr. 9, S. 1175, 1993. 6
- J. GUSTAFSON. *The Scaled-Sized Model: A Revision of Amdahl's Law*. In *Proceedings of the Third International Conference on Supercomputing*, Mai 1988. 105
- R. F. HARRINGTON. *Field Computation by Moment Methods*. Robert E. Krieger Publishing Company, Malabar, Florida, 2. Auflage, 1982, ISBN 0-89874-465-2. 19, 113
- V. HAVRAN, T. KOPAL, J. BITTNER und J. ŽÁRA. *Ray Tracing with Rope Trees*. *Proceedings of 13th Spring Conference On Computer Graphics, Budmerice, Slovakia*, 1998. 74
- D. HESTENES. *Space-Time Algebra*. Gordon and Breach Science Publishing, 1. Auflage, 1966. 6
- E. W. HOCKNEY und C. R. JESSHOPE. *Parallel Computers*. Hilger, Bristol, 1981. 109
- B. C. IGNATOWSKY. *Trans. State Opt. Inst. (Petrograd)*, Bd. 1, Nr. 3, 1919. 5
- A. L. INGBER. *Very Fast Simulated Re-Annealing*. *Mathematical and Computer Modelling*, Bd. 12, S. 967–973, 1989. <http://www.ingber.com> 66, 67, 68, 97, 98
- A. L. INGBER. *Adaptive Simulated Annealing (ASA): Lessons learned*. *Control and Cybernetics*, 1995. <http://www.ingber.com> 97
- J. B. KELLER. *Geometrical Theory of Diffraction*. *J. Opt. Soc. Am.*, Bd. 52, S. 116 – 130, 1962. 14

- D. KIRK und J. ARVO. *The Ray Tracing Kernel*. In *Proceedings of Ausgraph '88, Melbourne, Australia*, S. 75–88, 1988. 29, 31
- S. KIRKPATRICK, C. D. GELATT und M. P. VECCHI. *Optimization by Simulated Annealing*. *Science*, Bd. 220, S. 671–680, Mai 1983. 66
- S. KIRKPATRICK und M. P. VECCHI. *Global wiring by simulated annealing*. *IEEE Transactions on Computer Aided Design*, Bd. CAD-2, S. 215–222, 1983. 66
- M. KLINE und I. W. KAY. *Electromagnetic Theory and Geometrical Optics*. Interscience Publishers, New York, London, Sydney, 1965. 7
- R. G. KOUYOUMJIAN. *Asymptotic High-Frequency Methods*. *Proceedings of the IEEE*, Bd. 53, Nr. 8, S. 864 – 876, 1965. 5
- R. G. KOUYOUMJIAN und P. H. PATHAK. *A Uniform Geometrical Theory of Diffraction for an Edge in a Perfectly Conducting Surface*. *Proceedings of the IEEE*, Bd. 62, Nr. 11, S. 1448 – 1461, 1974. 14
- J. KŘIVÁNEK und V. BUBNIK. *Ray tracing with BSP and Rope trees*. 2000. <http://www.cg.tuwien.ac.at/studentwork/CESCG-2000/JKrivanek/> 74
- R. LÜBBERS. *Finite Conductivity Uniform GTD Versus Knife Edge Diffraction in Prediction of Propagation Path Loss*. *IEEE Transactions on Antennas and Propagation*, Bd. 32, Nr. 1, S. 70 – 76, 1984. 16
- R. LÜBBERS. *A Heuristic UTD Slope Diffraction Coefficient for Rough Lossy Wedges*. *IEEE Transactions on Antennas and Propagation*, Bd. 37, Nr. 2, S. 206 – 211, 1989. 17
- M. LOTT. *Entwurf eines drahtlosen multihop Ad-hoc-Funknetzes mit Dienstgüteunterstützung*. Dissertation, RWTH Aachen, Germany, 2002. 78
- J. M. MARTINEZ. *Continuous Optimization: state of the art*. In E. SPEDICATO, Hrsg., *Algorithms for Solving Nonlinear Systems of Equations*, S. 81–108. Kluwer, 1994. 95
- J. C. MAXWELL. *A Treatise On Electricity & Magnetism*. Dover Publications, New York, 1954. 6
- M. METROPOLIS, A. ROSENBLUTH, M. ROSENBLUTH, A. TELLER und E. TELLER. *Equation of state calculations by fast computing machines*. *Journal of Chemical Physics*, Bd. 21, S. 1087–1092, 1953. 2, 66, 68
- H. MEUER, E. STROHMAIER, J. DONGARRA und H. D. SIMON. *Weltrangliste der 500 schnellsten Rechnerinstallationen*, 2003. <http://www.top500.org> 104
- B. NAYLOR, J. AMANATIDES und W. C. THIBAUT. *Merging BSP Trees Yields Polyhedral Set Operations*. *Computer Graphics*, Bd. 24, Nr. 4, S. 115–124, August 1990. 33, 53, 54
- B. F. NAYLOR. *Constructing good partitioning trees*. In *Proceedings of Graphics Interface '93*, Toronto, CA, Mai 1993. 62
- J. O'ROURKE. *Computational Geometry in C*. Cambridge University Press, 2. Auflage, 1998. 45
- N. PREDIGER, A. PLATTNER und B. REMBOLD. *60 GHz Tunnel Propagation Measurement and Simulation*. In *Proceedings of the Millenium Conference on Antennas & Propagation, Davos*, 2000. 114
- W. H. PRESS, S. A. TEUKOLSKY, W. T. VETTERLING und B. P. FLANNERY. *Numerical Recipes in C*. Cambridge University Press, 1988. 66, 79, 95
- E. A. RAKHMANOV, E. B. SAFF und Y. M. ZHOU. *Minimal Discrete Energy on the Sphere*. *Mathematical Research Letters*, Bd. 1, 1994. 84, 85
- B. REMBOLD. *Simulation der Funkübertragung in einem Tunnel*. *Frequenz*, Bd. 47, Nr. 11 – 12, S. 270–275, 1993. 114
- B. REMBOLD. *MathCAD Skripte zur Berechnung der Reflexions- und Transmissionsfaktoren in geschichteten Dielektrika*. Unveröffentlichte private Kommunikation, 2000. 12

- W. RÜTTEN. *Untersuchungen zu Laser-Doppler-Verfahren in stark streuenden Medien mit Hilfe der Monte-Carlo-Methode Wellenausbreitung in pikozellularen Funknetzen*. Dissertation, RWTH Aachen, 1994, ISBN 3-8265-0392-9. 30
- P. J. SCHNEIDER und D. H. EBERLY. *Geometric Tools for Computer Graphics*. Morgan Kaufmann Publishers, 2002, ISBN 1-558-60594-0. 90, 91
- T. SCHÖBERL. *Polarimetrische Modellierung der elektromagnetischen Wellenausbreitung in pikozellularen Funknetzen*. Dissertation, RWTH Aachen, 1997, ISBN 3-8265-3098-5. 2, 16, 26, 31, 33, 58, 59, 78, 79, 80, 115, 116
- A. SCHROTH. *Berechnung der Strahlungscharakteristik von Antennen auf komplizierten Körpern im Hochfrequenzfall - Prinzip der geometrischen Beugungstheorie*. Deutsches Zentrum für Luft- und Raumfahrt (DLR), 1978. DFVLR-FB 78-O2. 7
- A. SCHROTH und V. STEIN. *Moderne numerische Verfahren zur Lösung von Antennen- und Streuproblemen*. R. Oldenbourg Verlag GmbH, München, 1985, ISBN 3-486-27751-0. 6, 7, 8, 16, 17
- P. SEBAH und X. GOURDON. *Newton's Method and High Order Iterations*, 2001. <http://numbers.computation.free.fr/Constants/constants.html> 95
- R. SEIDEL und C. R. ARAGON. *Randomized Search Trees*. *Algorithmica*, Bd. 16, Nr. 4/5, 1996. 46
- A. SOMMERFELD und J. RUNGE. *Anwendung der Vektorrechnung auf die Grundlagen der Geometrischen Optik*. *Ann. Phys.*, Bd. 35, S. 277 – 298, 1911. 5
- K. R. SUBRAMANIAN. *Adapting Search Structures to Scene Characteristics for Ray Tracing*. Dissertation, Department of Computer Sciences, The University of Texas at Austin, Dezember 1990. 30, 62
- K. R. SUBRAMANIAN und D. S. FUSSELL. *Factors affecting performance of ray tracing hierarchies*. Technischer Bericht TR-90-12, Department of Computer Sciences, The University of Texas at Austin, Juli 1990. <http://www.cs.uncc.edu/krs/publ.html> 29, 75
- K. R. SUBRAMANIAN und B. F. NAYLOR. *Converting Discrete Images to Partitioning Trees*. *IEEE Transactions on Visualization and Computer Graphics*, Bd. 3, Nr. 3, Juli 1997. 31
- K.R. SUBRAMANIAN und D.S. FUSSELL. *Automatic Termination Criteria for Ray Tracing Hierarchies*. In *Proceedings of Graphics Interface '91*, Calgary, Alberta, Oktober 1991. <http://www.cs.uncc.edu/krs/publ.html> 62
- W THIBAUT und B. NAYLOR. *Set Operations On Polyhedra Using Binary Space Partitioning Partitioning Trees*. *Computer Graphics*, Bd. 21, Nr. 4, S. 153–162, Juli 1987. 31, 49, 50, 52
- J.J. THOMSON. *On the Structure of the Atom: an Investigation of the Stability and Periods of Oscillation of a number of Corpuscles arranged at equal intervals around the Circumference of a Circle; with Application of the Results to the Theory of Atomic Structure*. *Philosophical Magazine*, Bd. 7, Nr. 39, S. 237 – 265, 1904. 84
- TRANSRAPID. <http://www.transrapid.de>, 2002. 118
- L TSANG, J. A. KONG, K. DING und C. O. AO. *Scattering of Electromagnetic Waves, Numerical Simulations*. John Wiley & Sons, 2001, ISBN 0-471-38800-9. 19, 20, 21, 98
- B.L. VAN DER WAERDEN. *Punkte auf der Kugel. Drei Zusätze*. *Math. Ann.*, Bd. 125, S. 213 – 222, 1952. 85
- M. VORLÄNDER. *Untersuchungen zur Leistungsfähigkeit des raumakustischen Schallteilchenmodells*. Dissertation, Rheinisch- Westfälische Technische Hochschule Aachen, 1989. 28, 78, 79, 80
- E. W. WEISSTEIN. *CRC Concise Encyclopedia of Mathematics*. Chapman & Hall/CRC, 2. Auflage, 2002. 20, 81, 82

- N. WIRTH. *Algorithmen und Datenstrukturen, Pascal-Version*. Teubner Verlag, 5. Auflage, 2000, ISBN 3-519-22250-7. [47](#)
- Z ZHANG, Z. YUN und M. F. ISKANDER. *3D tetrahedron ray tracing algorithm*. *Electronics Letters*, Bd. 37, Nr. 6, S. 334–335, März 2001. [31](#), [59](#)
- Y. M. ZHOU. *Arrangements of points on the sphere*. Dissertation, University of South Florida, Tampa, FL, 1995. [84](#), [85](#)





## LEBENS LAUF

---



8. Oktober 1968 Geboren in Hranice na Moravě (Tschechische Republik)
- 1975–1983 Grundschule, Hranice na Moravě
- 1983–1984 Gymnasium, Hranice na Moravě
- 1985–1989 Gymnasium, Antoniuskolleg in Neunkirchen-Seelscheid
- 10/1989–11/1996 Studium der Elektrotechnik an der RWTH Aachen
- 03/1997–12/2002 Wissenschaftlicher Mitarbeiter am Institut für Hochfrequenztechnik der RWTH Aachen
- ab 02/2003 Wissenschaftlicher Mitarbeiter, Philips Forschungslaboratorien in Aachen