

Uwe Schmidt

Schnelle modulare Exponentiation

Bachelorarbeit

Bibliografische Information der Deutschen Nationalbibliothek:

Bibliografische Information der Deutschen Nationalbibliothek: Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de/> abrufbar.

Dieses Werk sowie alle darin enthaltenen einzelnen Beiträge und Abbildungen sind urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsschutz zugelassen ist, bedarf der vorherigen Zustimmung des Verlanges. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen, Auswertungen durch Datenbanken und für die Einspeicherung und Verarbeitung in elektronische Systeme. Alle Rechte, auch die des auszugsweisen Nachdrucks, der fotomechanischen Wiedergabe (einschließlich Mikrokopie) sowie der Auswertung durch Datenbanken oder ähnliche Einrichtungen, vorbehalten.

Copyright © 2005 Diplomica Verlag GmbH
ISBN: 9783832489250

Uwe Schmidt

Schnelle modulare Exponentiation

Uwe Schmidt

Schnelle modulare Exponentiation

Bachelorarbeit
FernUniversität Hagen
Fachbereich Informatik
Abgabe Mai 2005



Diplom.de

Diplomica GmbH ———
Hermannstal 119k ———
22119 Hamburg ———

Fon: 040 / 655 99 20 ———
Fax: 040 / 655 99 222 ———

agentur@diplom.de ———
www.diplom.de ———

ID 8925
Schmidt, Uwe: Schnelle modulare Exponentiation
Hamburg: Diplomatica GmbH, 2005
Zugl.: FernUniversität Hagen, Bachelorarbeit, 2005

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtes.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Die Informationen in diesem Werk wurden mit Sorgfalt erarbeitet. Dennoch können Fehler nicht vollständig ausgeschlossen werden, und die Diplomarbeiten Agentur, die Autoren oder Übersetzer übernehmen keine juristische Verantwortung oder irgendeine Haftung für evtl. verbliebene fehlerhafte Angaben und deren Folgen.

Diplomatica GmbH
<http://www.diplom.de>, Hamburg 2005
Printed in Germany

Inhaltsverzeichnis

Erklärung	III
Inhaltsverzeichnis	V
Abbildungsverzeichnis	VII
Tabellenverzeichnis	IX
Listings	X
Symbolverzeichnis	XI
Kurzfassung	XII
1 Einleitung	1
2 Grundlagen	3
2.1 Euklidischer Algorithmus	3
2.2 Erweiterter euklidischer Algorithmus	3
2.3 Modulare Arithmetik, Restklassen	4
2.3.1 Rechenregeln der modularen Arithmetik	5
2.4 Primzahlen	6
2.5 Chinesischer Restsatz	7
2.6 O-Notation	7
3 Kryptographische Verfahren	8
3.1 Digital Signature Algorithm	9
3.2 ElGamal	10
3.3 Pohlig Hellman	12
3.4 Rabin	12
3.5 RSA	14
3.6 Zusammenfassung	16
4 Modulare Multiplikation	18
4.1 Modulare Multiplikation mit „Schulmethode“	18
4.2 Modulare Quadrierung	21
4.3 Rekursiver Multiplikationsalgorithmus	23

4.4	Montgomery-Multiplikation	24
5	Modulare Exponentiation ohne Precalculation	27
5.1	Square and Multiply	27
5.2	Left to Right Binary Method	29
5.3	m-ary Method	31
5.4	Fenstertechnik	36
5.5	Zusammenfassung	40
5.6	Modulare Exponentiation mit der Montgomery-Multiplikation	42
6	Modulare Exponentiation mit Precalculation	45
6.1	Additionsketten	45
6.1.1	Parallelisierung	50
6.2	Divisionsketten	52
6.3	BMGW-Algorithmus	58
6.4	Exponentiation mit chinesischem Restsatz	63
7	Ergebnis	65
8	Implementierung eines Verfahrens	68
A	Anhang	73
A.1	Versuchsordnung Rekursiver Multiplikationsalgorithmus	73
A.2	Tabellierung der Aufwandsfunktionen des BMGW-Algorithmus	81
A.3	Quellcode für die Implementierung des Divisionskettenverfahrens	84
	Literaturverzeichnis	104

Abbildungsverzeichnis

4.1	Beispiel Multiplikationen mit der „Schulmethode“	21
4.2	Beispiel Quadrierung mit der „Schulmethode“	22
5.1	Multiplikationen bei m-ary abhängig von der Nibblebreite	37
5.2	Vergleich Binärmethode, m-ary und Fenstertechnik	41
5.3	Vergleich Registerbedarf Binärmethode, m-ary und Fenstertechnik	41
5.4	Vergleich Multiplikation Klassisch, Baretts und Montgomery	44
6.1	Kürzeste Additionskette für 63	51
6.2	Parallelisierter AKG für $n = 63$	51
6.3	Anzahl Multiplikationen bei BMGW in Abhängigkeit der Basis b	61
7.1	Funktionspyramide für die Realisierung kryptographischer Verfahren	65
8.1	Klassendiagramm zur Implementierung des Divisionsketten-Verfahrens	69
8.2	Screenshot Testprogramm Divisionsketten Tabreiter Additionsketten	70
8.3	Screenshot Testprogramm Divisionsketten Tabreiter Divisionskette	71
8.4	Screenshot Testprogramm Divisionsketten Tabreiter Exponentiation	72

Tabellenverzeichnis

2.1	Beispiel: Erweiterter euklidischer Algorithmus $ggt(a, b) = 210 * x + b * y$. . .	4
3.1	Anzahl der modularen Exponentiationen in verschiedenen kryptographischen Verfahren	16
4.1	Beispiel Reduktion 55 mod 7	20
4.2	Zeitkomplexität beim rekursiven Multiplikationsalgorithmus	23
4.3	Vergleich der Laufzeiten für 1000 Multiplikationen	24
5.1	Beispiel Left to Right Binary Method	30
5.2	Berechnungen der a^{nib_i} bei m-ary Method mit d=1	33
5.3	Beispiel m-ary Method mit d=1	33
5.4	Berechnungen der a^{nib_i} bei m-ary Method mit d=2	33
5.5	Beispiel a^{250} m-ary Method mit d=2	33
5.6	Berechnung der a^{nib_i} m-ary Method mit d=3	34
5.7	Beispiel m-ary Method mit d=3	34
5.8	Aufwand bei m-ary Method für $c = a^{250} \bmod m$	34
5.9	Multiplikationen bei m-ary abhängig von der Nibblebreite	36
5.10	Prozentuale Ersparnis m-ary Methode gegenüber Binär Methode	36
5.11	Multiplikationen bei Sliding Window abhängig von der Nibblebreite	38
5.12	Prozentuale Ersparnis Fenstermethoden gegenüber m-ary Methode	39
5.13	Berechnung der a^{nib_i} Sliding Window mit d=4	39
5.14	Vergleich Multiplikation Klassisch, Baretts und Montgomery	43
5.15	Zeitgewinn Montgomery vs. Klassisch in Prozent	43
6.1	Beispiel: Erzeugen einer Additionskette für 250 mit der Binary-Methode . . .	49
6.2	Modulare Exponentiation von $g^{250} \bmod m$ mit Additionskette	50
6.3	Divisionskette für 16806267284414399481 mit Strategie 2	56
6.4	Beispiel für Precomputation bei „einfachem BMGW“	58
6.5	Durchschnittliche Anzahl an Multiplikationen bei Vorausberechnung von Zweierpotenzen	59
6.6	Anzahl Multiplikationen beim BMGW-Algorithmus	62
6.7	Anzahl Multiplikationen und Speicherbedarf mit unterschiedlichen Zerlegungsmethoden	63
A.1	Laufzeit von je 1000 Multiplikationen von 1024-Bitzahlen in Millisekunden .	76
A.2	Laufzeit von je 1000 Multiplikationen von 2048-Bitzahlen in Millisekunden .	77

A.3	Laufzeit von je 1000 Multiplikationen von 4096-Bitzahlen in Millisekunden	79
A.4	Laufzeit von je 1000 Multiplikationen von 8192-Bitzahlen in Millisekunden	81
A.5	Ergebnis des Vergleichs RecursiveMult und BigInteger.multiply	81
A.6	Auswertung der Aufwandsfunktionen des BMGW-Algorithmus	83

Listings

4.1	Klassischer Multiplikationsalgorithmus	18
4.2	Multiplikationsalgorithmus für Dualzahlen	19
4.3	Reduktion $a \bmod m$	20
4.4	Quadrierungsalgorithmus	22
4.5	Montgomeryprodukt MP	25
5.1	Left to Right Binary Method	30
5.2	m-ary-Method	32
5.3	Binary-Method mit klassischer modularer Multiplikation	42
5.4	Binary Method mit Montgomerymultiplikation	42
6.1	Erzeugen einer Additionskette mit der Binär-Methode	48
6.2	Modulare Exponentiation mit Additionskette	49
6.3	„Divisionskette Strategie 1“	54
6.4	„Divisionskette Strategie 2“	54
6.5	„Exponentiation mit Divisionsketten“	57
6.6	„Einfaches BMGW“	58
6.7	„BMGW“	59
A.1	Rekursiver Multiplikationsalgorithmus	73
A.2	AddChainCollection (Implementierung des Divisionskettenverfahrens)	84
A.3	AddChainCollection (Implementierung des Divisionskettenverfahrens)	89
A.4	CalcDivChain (Implementierung des Divisionskettenverfahrens)	94
A.5	DivChainExp (Implementierung des Divisionskettenverfahrens)	98
A.6	SQM (Implementierung des Divisionskettenverfahrens)	101

Symbolverzeichnis

$\lfloor x \rfloor$	Kleinste ganze Zahl größer oder gleich x
$\lceil x \rceil$	Größte ganze Zahl kleiner oder gleich x
$a \mid n$	a teilt n d.h, a ist ein Teiler von n und n ein Vielfaches von a
$a \nmid n$	a ist kein Teiler von n
$a \equiv b \pmod{m}$	a ist kongruent zu b modulo m
$a \not\equiv b \pmod{m}$	a ist nicht kongruent zu b modulo m
$v(n)$	Bezeichnet die Anzahl der Einsen in der Binärdarstellung von n
$\text{ggT}(a, b)$	größter gemeinsamer Teiler von a und b
$m\mathbb{Z}$	Die Menge aller Linearkombinationen von m
$\mathbb{Z}/m\mathbb{Z}$	Die Menge aller Restklassen mod m
\mathbb{Z}_m	Vertretersystem von $\mathbb{Z}/m\mathbb{Z}$ das aus jeder Restklasse das kleinste Element enthält.