

Inhalt

1	Einleitung	1
2	Anforderungsanalyse	7
2.1	Ineffizienzen objektorientierter Programme	7
2.1.1	Ineffizienzen durch Abstraktion	8
2.1.2	Ineffizienzen durch anwendungsunabhängigen Entwurf . . .	12
2.1.3	Ineffizienzen durch hohe Kontextabhängigkeit	13
2.2	Benötigte Optimierungen	14
2.2.1	Traditionelle Optimierungen	14
2.2.2	Elimination unnötiger Abstraktionen	15
2.2.3	Spezialisierung des anwendungsunabhängigen Entwurfs . .	17
2.2.4	Reduktion der Kontextabhängigkeit	17
2.3	Benötigte Programminformationen	18
2.3.1	Definitionen und Benutzungen	18
2.3.2	Partielle Ordnung und Abhängigkeiten	20
2.4	Approximation von Programminformationen	22
2.4.1	Numerische Werte	22
2.4.2	Variablen	23
2.4.3	Namensschemata	23
2.4.4	Aliasproblematik	24
2.4.5	Schwache Aktualisierung	25
2.4.6	Essentielle Abhängigkeiten	26
2.5	Beziehungen zwischen Programminformationen	26
2.6	Zusammenfassung	28
3	Stand von Forschung und Technik	31
3.1	Optimierungen	31
3.1.1	Optimierungsstrategien	31
3.1.2	Traditionelle Optimierungen	32
3.1.3	Elimination unnötiger Abstraktionen	34
3.1.4	Spezialisierung des anwendungsunabhängigen Entwurfs . .	36
3.1.5	Reduktion der Kontextabhängigkeit	37
3.2	Programmanalyse	38
3.2.1	Datenflußanalyse	39
3.2.2	Steuerflußanalyse, Typanalyse	41
3.2.3	Zeigeranalyse, Haldenanalyse	43
3.2.4	Genauigkeit und Aufwand	45
3.3	Programmrepräsentation	47
3.3.1	Grundblockgraph	47
3.3.2	Static Single Assignment Darstellungen	49
3.3.3	Ausgedünnte Repräsentationen	55
3.3.4	Ausnahmebehandlung	56
3.3.5	Abstraktionsniveau	58
3.4	Offene Probleme	59

4	Explizite Abhängigkeitsgraphen	61
4.1	Darzustellende Informationen	62
4.1.1	Operationen, Steuerfluß und Datenfluß	62
4.1.2	Speicherabbildung	63
4.1.3	Ausnahmen	63
4.1.4	Typinformationen und statisch initialisierte Daten	64
4.2	Syntax und Semantik expliziter Abhängigkeitsgraphen	65
4.2.1	Operationen und Abhängigkeiten	67
4.2.2	Speicherabhängigkeiten	74
4.2.3	Zugriffsfunktionen	81
4.2.4	Ausnahmen	83
4.2.5	Typinformationen	86
4.3	Aufbau aus der semantischen Analyse	87
4.3.1	Hilfsstrukturen	88
4.3.2	Bestimmung von Abhängigkeiten	89
4.3.3	Vermeidung unnötiger Phi-Operationen	90
4.3.4	Minimalität	92
4.3.5	Aufwand	93
4.4	Abbau der Darstellung	94
4.4.1	Transformation komplexer Operationen	94
4.4.2	Abbildung von Datenflußkanten	94
4.4.3	Auflösung von Phi-Operationen	95
4.5	Zusammenfassung	95
5	Programmanalyse über expliziten Abhängigkeitsgraphen	97
5.1	Analyse von Werten und Abhängigkeiten	97
5.1.1	Abstrakte Bereiche	97
5.1.2	Analyse von Variableninhalten und Ausdrücken	99
5.1.3	Bestimmung von Definitionen und Benutzungen	102
5.2	Beschleunigung der Konvergenz	102
5.2.1	Besuchsreihenfolge	103
5.2.2	Analyse über dem Schleifenbaum	106
5.3	Reduktion des Speicherbedarfs	108
5.3.1	Einsparungspotentiale	108
5.3.2	Bedarfsgesteuerte Analyse	110
5.3.3	Zusammenfassen von Operationen zu Gebieten	112
5.4	Modellierung kontextsensitiver Werte mit χ -Termen	114
5.4.1	Symbolische Darstellung kontextsensitiver Werte	116
5.4.2	χ -Terme als Entscheidungsdiagramme	120
5.4.3	Kontextsensitive Transferfunktionen über χ -Termen	123
5.4.4	Erzwingen einer geeigneten Variablenordnung	124
5.5	Zusammenfassung	125

6	Optimierung expliziter Abhängigkeitsgraphen	127
6.1	Transformationen als Graphersetzung	128
6.1.1	Ersetzungsregeln	129
6.1.2	Gesteuerte Ersetzung	130
6.1.3	Effiziente Erhaltung von Identitäten	130
6.1.4	Normalisierung	132
6.2	Traditionelle Optimierungen	132
6.2.1	Konstantenfaltung und partielle Auswertung	132
6.2.2	Beseitigung von Kopien und totem Code	134
6.2.3	Globale Vermeidung redundanter Berechnungen	135
6.3	Beseitigung objektorientierter Ineffizienzen	139
6.3.1	Reduktion von Polymorphie und Prozeduraufrufen	139
6.3.2	Reduktion von Speicherzugriffen	140
6.3.3	Elimination dynamischer Objekterzeugungen	143
6.4	Reduktion nicht essentieller Abhängigkeiten	144
6.4.1	Verfeinerung des Namensschemas	145
6.4.2	Pfadverkürzung durch lokale Transformation	146
6.4.3	Globale Bestimmung essentieller Abhängigkeiten	147
6.5	Iterative optimierende Übersetzung	150
6.6	Zusammenfassung	154
7	Praktische Ergebnisse	155
7.1	Der Sather-K Übersetzer	155
7.2	Testprogramme	157
7.3	Messungen	159
7.4	Speichereffizienz von χ -Termen	163
7.5	Vergleich zu anderen Übersetzern	166
7.6	Diskussion	166
8	Zusammenfassung	169
	Literatur	171
A	Entscheidungsdiagramme	177
A.1	Geordnete, binäre Entscheidungsdiagramme	177
A.2	Größe von OBDDs	178
A.3	Symbolisches Rechnen auf OBDDs	178
A.4	Effiziente Implementierung	179
A.5	Andere Typen von Entscheidungsdiagrammen	180
B	Beispielprogramme	181
	Sachverzeichnis	185

Optimierung objektorientierter Programme
Übersetzungstechniken, Analysen und
Transformationen

Trapp, M.

2001, IX, 189 S., Hardcover

ISBN: 978-3-540-42352-2