

Inhaltsverzeichnis

1	Einleitung	1
1.1	Software-Produktlinien	2
1.2	Statische Analyse	5
1.3	Problembeschreibung und Ziele der Arbeit	7
1.4	Aufbau der Arbeit	14
2	Wissenschaftliche Grundlagen	17
2.1	Compiler und statische Analyse	18
2.2	Präprozessor	21
2.3	KConfig	27
2.4	Variabilitätsmanagement	31
2.5	Logik	34
2.5.1	Aussagenlogik	34
2.5.2	Verschachtelung	35
2.5.3	Nicht-boolsche Variablen	36
2.5.4	DIMACS Format und SAT-Solver	37
2.6	Verwandte Arbeiten	40
2.6.1	Arbeiten zur statischen Analyse	40
2.6.2	Arbeiten zur formalen Begriffsanalyse	42
2.7	SuperC	45
2.8	Zusammenfassung des wissenschaftlichen Hintergrunds	53

3	Erweiterung des Syntaxgraphen	55
3.1	Formale Grundlagen	57
3.1.1	Logik	57
3.1.2	Präprozessorblöcke	60
3.1.3	Graphen	62
3.2	Syntaxgraph	69
3.3	Bedingungsknoten	77
3.4	Berechnung der Bedingungen	82
3.5	Optimierung des Syntaxgraphen	86
3.6	Textuelle Reihenfolge der Knoten	89
3.6.1	Die initiale Reihenfolge	90
3.6.2	Berechnung der textuellen Reihenfolge	92
3.7	Bedingte Reihenfolge der Knoten	94
3.8	Algorithmus zur Berechnung der Reihenfolge	101
4	Kontrollfluss	107
4.1	Unbedingter Kontrollfluss	109
4.2	Aufbau des präprozessorbedingten Kontrollflussgraphen	113
4.3	Bedingter Kontrollfluss auf Syntaxgraphen	119
4.4	Bedingte Kontrollflusskanten	123
4.4.1	true-Kante	124
4.4.2	false-Kante	126
4.4.3	Neuberechnung der Bedingungen	129
4.4.4	Beispiel	129
4.5	Das Ende des then-Blocks	136
5	Software-Produktlinien als Verbände	145
5.1	Formale Begriffsanalyse	148
5.2	Formale Begriffsanalyse für SPL	151

5.3	Komplexe Presence Conditions	154
5.4	Erweiterte formale Begriffsanalyse	159
6	Evaluation	165
6.1	Diskussion der asymptotischen Laufzeit	168
6.2	Die Verwendung von Präprozessorvariablen	172
6.3	Laufzeitverhalten der Kontrollkantenbestimmung	176
6.3.1	Versuchsaufbau	176
6.3.2	Einfluss der Kontrollanweisungen auf die Laufzeit	179
6.3.3	Einfluss der Komplexität der Präprozessorausdrücke auf die Laufzeit	187
6.3.4	Einflüsse auf die Validität und auf die Aussagekraft der Eva- luation	190
6.3.5	Fazit der Untersuchung des Laufzeitverhaltens	192
6.4	Linux-Fallstudie	194
6.4.1	Variabilitätsuntersuchung	194
6.4.2	Formale Begriffsanalyse mit einem Variabilitätsmodell aller Linux-Architekturen	196
6.4.3	Architekturspezifische Analyse	208
6.4.4	Einschränkungen der Fallstudie	212
7	Ausblick und Fazit	215
7.1	Alternative Formen bedingter statischer Analyse	217
7.2	Laufzeitverhalten komplizierter Syntaxgraphen	219
7.3	Ausblick bedingte Kontrollabhängigkeit	220
7.3.1	Kontrollabhängigkeit ohne Präprozessor	220
7.3.2	Kontrollabhängigkeit mit Präprozessor	225
7.4	Ausblick bedingter System Dependency Graph	229
7.5	Ausblick zur Analyse von beliebigen C-Dateien	234

7.6	Ergebnis der erweiterten formalen Begriffsanalyse	236
7.7	Schlussworte	239
	Literaturverzeichnis	240
	Anhang	251
	Anhang A Tools	253
A.1	Tool zur Erzeugung von bedingten Kontrollkanten.	253
A.2	Tool für die formale Begriffsanalyse	254
	Anhang B Beispiele	259
B.1	Variablenidentität	259
B.2	Die ISDN Konfiguration: Ein Beispiel einer C-Datei in Linux	264