

Inhaltsverzeichnis

| | |
|---|----|
| 1 Einführung: Verteilte, nebenläufige und interaktive Systeme .. | 1 |
| 1.1 Vorbemerkungen | 2 |
| 1.2 Grundbegriffe und Grundkonzepte | 4 |
| 1.3 Wesentliche Eigenschaften vernetzter Systeme | 6 |
| 1.4 Fundamentale Fragen bei der Modellierung vernetzter Systeme | 8 |
| 1.5 Ein einführendes Beispiel: Das Alternating-Bit-Protokoll ... | 9 |
| 1.6 Konventionen, Notationen und grundlegende Strukturen | 20 |
| 1.6.1 Konvention: Sorten und Trägermengen | 20 |
| 1.6.2 Endliche und unendliche Sequenzen | 20 |
| 1.6.3 Prädikate und Zusicherungen | 22 |
| 1.6.4 Linear-temporale Logik (LTL) | 27 |
| 1.7 Historische Bemerkungen | 32 |
| 1.8 Übungsaufgaben | 34 |

Teil I Zustandsbasierte Systemmodelle

| | |
|--|----|
| 2 Zustandssicht: Systeme als Zustandsmaschinen | 43 |
| 2.1 Zustandsmaschinen mit unmarkierten Übergängen | 44 |
| 2.1.1 Zustandsmaschinen und ihre Darstellungsformen.... | 44 |
| 2.1.1.1 Mathematische Definition | 44 |
| 2.1.1.2 Grafische Darstellung | 47 |
| 2.1.1.3 Annotierte Zustandsübergangsdiagramme.. | 49 |
| 2.1.1.4 Matrixdarstellung von Zustandsmaschinen . | 51 |
| 2.1.1.5 Attributierte Zustände | 53 |
| 2.1.2 Semantik: Abläufe und Erreichbarkeit | 57 |
| 2.1.2.1 Abläufe | 57 |
| 2.1.2.2 Erreichbarkeit | 61 |
| 2.1.3 Logische Spezifikation und Invarianten | 67 |
| 2.1.4 Parallele Komposition | 70 |
| 2.1.4.1 Parallele Komposition ohne gemeinsamen Speicher | 71 |

| | | |
|-----------|--|------------|
| 2.1.4.2 | Zustandsmaschinen mit gemeinsamem Speicher | 74 |
| 2.1.4.3 | Stabile Prädikate und Invarianten | 84 |
| 2.1.5 | Verifikation | 87 |
| 2.1.5.1 | Invarianten | 87 |
| 2.1.5.2 | Einschub: Komplexität maschinengestützter Verifikation | 88 |
| 2.1.5.3 | Linearzeitaussagen | 91 |
| 2.2 | Zustandsmaschinen mit actionsmarkierten Übergängen | 92 |
| 2.2.1 | Darstellungsformen | 92 |
| 2.2.1.1 | Analytische Darstellung | 92 |
| 2.2.1.2 | Grafische Darstellung | 97 |
| 2.2.1.3 | Tabellarische Darstellung | 100 |
| 2.2.1.4 | Matrixdarstellung | 102 |
| 2.2.2 | Semantik | 104 |
| 2.2.2.1 | Sequenzbasierte Semantiken | 104 |
| 2.2.2.1.1 | Berechnungen | 104 |
| 2.2.2.1.2 | Abläufe | 104 |
| 2.2.2.1.3 | Spuren | 105 |
| 2.2.2.2 | Erreichbarkeitssemantiken | 106 |
| 2.2.3 | Logische Spezifikation | 107 |
| 2.2.3.1 | Spurlogik | 108 |
| 2.2.3.1.1 | Reguläre und ω -reguläre Ausdrücke | 108 |
| 2.2.3.1.2 | LTL | 108 |
| 2.2.3.1.3 | Prädikatenlogik erster Stufe | 109 |
| 2.2.3.2 | Erreichbarkeitslogik | 111 |
| 2.2.4 | Parallele Zusammensetzung | 112 |
| 2.3 | Zustandsmaschinen mit Eingabe und Ausgabe | 112 |
| 2.3.1 | Analytische Darstellung | 112 |
| 2.3.2 | Zustandsübergänge beschrieben durch Zusicherungen | 116 |
| 2.3.3 | Zustandsübergangsdiagramme | 116 |
| 2.3.4 | Tabellarische Darstellung | 119 |
| 2.3.5 | Matrixdarstellung | 122 |
| 2.3.6 | Pragmatische Spezifikation von Zustandsmaschinen . | 124 |
| 2.4 | Abschließende Bemerkungen zu Zustandsmaschinen | 133 |
| 2.5 | Historische Bemerkungen | 134 |
| 2.6 | Übungsaufgaben | 135 |
| 3 | Attributierte Zustandsräume und parallele Programme | 139 |
| 3.1 | Programmnahe Zustandsübergangssysteme | 139 |
| 3.1.1 | Logische Spezifikation | 140 |
| 3.1.2 | Von Zustandsübergangsdiagrammen zu Zustandsmaschinen | 144 |

| | | |
|-------|--|-----|
| 3.1.3 | Zustandsübergangssysteme als anweisungsorientierte Programme | 147 |
| 3.1.4 | Anweisungsorientierte Programme als Zustandsübergangssysteme | 148 |
| 3.2 | Imperative Programme mit gemeinsamen Variablen | 151 |
| 3.2.1 | Parallele Komposition | 152 |
| 3.2.2 | Syntax | 153 |
| 3.2.3 | Semantik | 154 |
| 3.2.4 | Spezifikation | 155 |
| 3.2.5 | Wechselseitiger Ausschluss | 155 |
| 3.2.6 | Verifikation anweisungsorientierter paralleler Programme | 161 |
| 3.3 | Historische Bemerkungen | 168 |
| 3.4 | Übungsaufgaben | 169 |

Teil II Asynchrone Interaktion, Verteilung und Zeit

| | | |
|-------|---|-----|
| 4 | Schnittstellen asynchroner interaktiver Systeme | 179 |
| 4.1 | Interaktive Schnittstellen | 179 |
| 4.1.1 | Schnittstellensicht für interaktive Systeme | 180 |
| 4.1.2 | Syntaktische Schnittstellen: Kanäle und gemeinsame Variablen | 181 |
| 4.2 | Ströme | 184 |
| 4.3 | Stromverarbeitende Funktionen | 188 |
| 4.4 | Spezifikation von Strömen | 198 |
| 4.5 | Spezifikation stromverarbeitender Funktionen | 200 |
| 4.6 | Kommunikation zwischen imperativen Programmen | 205 |
| 4.6.1 | Imperative Programme mit Kommunikationsanweisungen | 205 |
| 4.6.2 | Prädiktative Spezifikation | 207 |
| 4.7 | Sicherheits- und Lebendigkeitsbedingungen auf Strömen | 218 |
| 4.8 | Spezifikation durch Annahmen und Verpflichtungen | 220 |
| 4.9 | Von Zustandsmodellen zu stromverarbeitenden Funktionen und zurück | 226 |
| 4.9.1 | Zustandsmaschinen zur Beschreibung stromverarbeitender Funktionen | 227 |
| 4.9.2 | Stromverarbeitende Funktionen als Zustandsmaschinen | 228 |
| 4.9.3 | Schnittstellenabstraktion | 229 |
| 4.10 | Historische Bemerkungen | 231 |
| 4.11 | Übungsaufgaben | 232 |
| 5 | Asynchrone Systemarchitekturen | 235 |
| 5.1 | Verteilte Systeme | 236 |
| 5.1.1 | Architekturen verteilter Systeme | 236 |
| 5.1.2 | Parallele Komposition | 238 |

| | | |
|----------|--|------------|
| 5.1.3 | Modularität | 238 |
| 5.2 | Komposition von Systemen zu Datenflussnetzen | 239 |
| 5.2.1 | Systeme als Datenflussnetze | 239 |
| 5.2.2 | Komponenten von Datenflussnetzen | 243 |
| 5.2.3 | Parallele Komposition mit Rückkopplung | 244 |
| 5.2.4 | Komposition von Systemspezifikationen | 246 |
| 5.2.5 | Nichtdeterministische Komponenten | 252 |
| 5.3 | Formen von Datenflussnetzen | 253 |
| 5.3.1 | Algorithmen als Datenflussnetze | 253 |
| 5.3.2 | Kantenführung in Datenflussnetzen | 262 |
| 5.3.3 | Berechnung von Funktionen durch Datenflussnetze . . | 263 |
| 5.3.4 | Annahmen und Verpflichtungen in der Komposition . | 266 |
| 5.4 | Kompositionsformen | 270 |
| 5.4.1 | Parallele Komposition ohne Rückkopplung | 270 |
| 5.4.2 | Sequenzielle Komposition: Pipeline | 271 |
| 5.4.3 | Rückkopplung | 272 |
| 5.4.4 | Das Beispiel Fensterheber | 273 |
| 5.5 | Schichtenarchitekturen | 275 |
| 5.6 | Parallele, anweisungsorientierte, interaktive Programme . . | 280 |
| 5.6.1 | Parallele Komposition interaktiver, anweisungsorientierter Programme | 280 |
| 5.6.2 | Interaktive Systeme anweisungsorientierter Programme | 285 |
| 5.7 | Parallele Programme als stromverarbeitende Funktionen . . | 286 |
| 5.8 | Historische Bemerkungen | 289 |
| 5.9 | Übungsaufgaben | 291 |
| 6 | Zeit- und Datenfluss | 293 |
| 6.1 | Ein einfaches diskretes Zeitmodell | 294 |
| 6.2 | Spezifikation und Verifikation von zeitbehafteten Systemen . | 299 |
| 6.3 | Zeitschnitte | 302 |
| 6.4 | E/A/K-Zustandsmaschinen | 303 |
| 6.5 | Parallele Komposition von E/A/K-Zustandsmaschinen . . | 305 |
| 6.6 | Globale Zeit | 306 |
| 6.6.1 | Fixpunkteigenschaft stark kausaler Systeme | 307 |
| 6.6.2 | Gezeitete Systeme: Ein einführendes Beispiel | 308 |
| 6.6.3 | Zeit als Mittel zum Nachweis von Systemeigenschaften | 309 |
| 6.6.4 | Zeit und Kausalität | 312 |
| 6.6.5 | Zeitsynchrone, nachrichtenynchrone Systeme | 314 |
| 6.7 | Spezifikation stark kausaler Systeme | 316 |
| 6.8 | Komposition | 318 |
| 6.9 | Starke Kausalität für Annahme-Verpflichtungs-Spezifikationen | 321 |
| 6.10 | Sender und Empfangen: „Push“ und „Pull“ | 322 |
| 6.11 | Zeitsynchrone, nachrichtensynchrone Systeme und perfekte Synchronität | 328 |
| 6.12 | Historische Bemerkungen | 328 |

| | |
|--|-----|
| 6.13 Übungsaufgaben | 330 |
| Teil III Prozesse, synchroner Nachrichtenaustausch und Verfeinerung | |
| 7 Prozesse als Abläufe verteilter Systeme | 339 |
| 7.1 Nebenläufige Prozesse | 340 |
| 7.1.1 Aktionsstrukturen | 340 |
| 7.1.2 Eigenschaften von Prozessen | 344 |
| 7.1.3 Sequenzielle Prozesse als Ströme von Aktionen | 349 |
| 7.2 Prozesse als Aktionsfolgen und temporale Logik | 349 |
| 7.3 Koordination | 354 |
| 7.4 Petri-Netze als Prädikate über Prozessen | 355 |
| 7.5 Abläufe und Schnittstellenverhalten | 357 |
| 7.6 Diagramme zur Darstellung von Prozessen | 359 |
| 7.6.1 Kontrollflussdiagramme | 359 |
| 7.6.2 Graphische Darstellung von Prozessen | 362 |
| 7.7 TLA – Temporal Logic of Actions | 366 |
| 7.8 Historische Bemerkungen | 375 |
| 7.9 Übungsaufgaben | 376 |
| 8 Nachrichtensynchrone Systeme | 379 |
| 8.1 Nachrichtensynchrone Komposition von Zustandsmaschinen | 380 |
| 8.2 Syntax von CSP | 382 |
| 8.3 Operationelle Semantik für CSP | 385 |
| 8.4 Bereitschaft und Verweigerung | 389 |
| 8.5 Historische Bemerkungen | 393 |
| 8.6 Übungsaufgaben | 395 |
| 9 Zustands- und Ablaufsicht | 397 |
| 9.1 Komposition von Systemen und Kontexte | 397 |
| 9.2 Aktionsstrukturen und Zustandsbegriff | 398 |
| 9.3 Zustandsübergangsbäume | 399 |
| 9.4 Fairness und temporale Logik zustandsorientierter Systeme .. | 402 |
| 9.5 Äquivalenzbegriffe für Systeme mit aktionsmarkierten Übergängen | 403 |
| 9.6 Bisimulation und Bereitschaftsäquivalenz | 405 |
| 9.7 Historische Bemerkungen | 410 |
| 9.8 Übungsaufgaben | 410 |
| 10 Verfeinerung von Systemen | 413 |
| 10.1 Eigenschaftsverfeinerung | 414 |
| 10.2 Teifunktionalität | 415 |
| 10.3 Repräsentationsverfeinerung | 416 |
| 10.4 Modularität und Kompatibilität | 423 |
| 10.5 Verfeinerung von Zustandsübergangssystemen | 424 |
| 10.5.1 Nichtdeterministische Auswahl | 424 |

| | |
|--|------------|
| 10.5.2 Verfeinerung der Übergangsfunktion | 425 |
| 10.5.3 Repräsentationsverfeinerung für Zustandssysteme | 426 |
| 10.6 Abstraktion von Zustandsmaschinen | 431 |
| 10.7 Historische Bemerkungen | 432 |
| 10.8 Übungsaufgaben | 432 |
| Literaturverzeichnis | 435 |
| Symbolverzeichnis | 451 |
| Sachverzeichnis | 457 |