

Kooperationsprodukte formaler Sprachen und schlichte Homomorphismen

Peter Ochsenschläger
Institut für Telekooperationstechnik der GMD

Zusammenfassung

Mit dem Begriffs- und Methodenapparat formaler Sprachen wird ein kompositionelles Verifikationsinstrumentarium für kooperierende Systeme entwickelt. Kooperierende Systeme sind solche verteilten Systeme, welche durch eine "lose Kopplung" und "Entscheidungsfreiheiten" ihrer Komponenten charakterisiert sind. Typische Beispiele hierfür sind Telefonsysteme, Chipkartensysteme, Kommunikationsprotokolle etc.

Das dynamische Verhalten eines solchen Systems wird durch die Menge aller möglichen "Aktionsfolgen", also durch eine formale Sprache dargestellt. Die Verifikationsmethode basiert auf einem Abstraktionsbegriff, welcher durch eine spezielle Klasse von Sprachhomomorphismen (schlichte Homomorphismen) formalisiert ist. Im Kontext kooperierender Systeme berücksichtigen diese Homomorphismen sowohl Sicherheits- als auch Lebendigkeitseigenschaften in adäquater Weise. Das Zusammenspiel der Komponenten eines kooperierenden Systems wird durch Kooperationsprodukte formaler Sprachen beschrieben. Diesem Konzept entsprechen auf automatentheoretischer Seite sogenannte asynchrone Produktautomaten. Zum Nachweis der Schlichtheit von Homomorphismen auf Kooperationsprodukten wird der Begriff der Kooperativität der Komponenten von Kooperationsprodukten eingeführt.

Mit der entwickelten Methode können in kompositioneller Weise homomorphe Bilder von Systemverhalten berechnet und die Schlichtheit der Homomorphismen überprüft werden. Dabei werden unnötige Nebenläufigkeiten vermieden, was bei wohlstrukturierten Systemspezifikationen zu erheblichen Komplexitätsreduktionen führt. Das Verfahren kann auch iterativ benutzt werden und ermöglicht bei parametrisierten Systemspezifikationen Induktionsbeweise.

Einleitung

Gegenstand dieser Arbeit ist die Untersuchung des dynamischen Verhaltens verteilter Systeme - genauer gesagt kooperierender Systeme - mit dem Begriffs- und Methodenapparat formaler Sprachen. Von besonderem Interesse ist dabei der Vergleich von Systemspezifikationen unterschiedlichen Abstraktionsniveaus, welcher üblicherweise Verifikation genannt wird. Mit dem Begriff kooperierende Systeme sind solche verteilten Systeme gemeint, welche durch eine "lose Kopplung" und "Entscheidungsfreiheiten" ihrer Komponenten charakterisiert sind. Typische Beispiele hierfür sind Telefonsysteme, Chipkartensysteme, Kommunikationsprotokolle etc.

Von einem naiven Standpunkt aus erscheint es nahezu natürlich, das dynamische Verhalten eines solchen Systems durch die Menge aller möglichen "Aktionsfolgen", also durch eine formale Sprache darzustellen. Die zum Vergleich von Systemspezifikationen benötigten "Abstraktionen" lassen sich dann mit (alphabetischen) Sprachhomomorphismen beschreiben. Bei genauerer Betrachtungsweise zeigt es sich aber, daß dabei die Lebendigkeitseigenschaften, die ja auf das "zukünftige" Systemverhalten abzielen /AS85/, nicht adäquat berücksichtigt werden, d. h. Lebendigkeitseigenschaften eines homomorphen Bildes gelten nicht unbedingt auch für die Menge der Aktionsfolgen eines Systems.

Bekanntlich werden diese Schwierigkeiten durch verschiedene Ansätze umgangen, welche von einer Darstellung von Systemverhalten mittels beschrifteter Transitionssysteme (labelled transition systems, LTS) /Arn94/ ausgehen. Verifikation beruht dann entweder auf Äquivalenzbegriffen für diese LTS, welche feiner als die Sprachäquivalenz sind /BW90,Mil90/, oder es werden Systemeigenschaften mittels temporaler Logik oder ω -Sprachen formuliert und direkt an den LTS nachgewiesen (model checking) /CES86,Kur93/. Gegenüber diesen Ansätzen wurde mit der Definition der Schlichtheit von Sprachhomomorphismen in /Och92,94a/ ein anderer Weg eingeschlagen, welcher durch Einschränkung der betrachteten Homomorphismenklasse den natürlichen Ansatz der formalen Sprachen für die Systemverifikation nutzbar macht. Es sei bemerkt, daß auch jedes LTS durch seine lokale Sprache /Eil74/ in eindeutiger Weise mittels einer formalen Sprache dargestellt ist.

Wegen des nichtdeterministischen Verhaltens kooperierender Systeme hat es sich gezeigt /NO95/, daß in diesem Zusammenhang Lebendigkeitseigenschaften nur in einer "approximativen Weise" sinnvoll sind, und daß dies adäquat von schlichten Homomorphismen berücksichtigt wird. Bezüglich temporaler Logik wurden in /Nit96a,b/ den schlichten Homomorphismen entsprechende syntaktische Transformationen der Logikformeln angegeben. Die schlicht-homomorphen Bilder der Menge der Aktionsfolgen eines Systems liefern eine kompakte und bezüglich der gewählten abstrahierenden Sichtweise auch vollständige Darstellung des Systemverhaltens, die schon in mehreren komplexen Anwendungen /C&96a,b,Neb94,Och94c/ erfolgreich für Verifikationszwecke eingesetzt wurde.

Vor diesem Hintergrund stellt sich natürlich die Frage nach einem Kompositionsbegriff für formale Sprachen, welcher dem Zusammenspiel der einzelnen Komponenten eines kooperierenden Systems entspricht. Im ersten Kapitel der vorliegenden Arbeit wird mit dem Kooperationsprodukt formaler Sprachen ein solches Konzept entwickelt, welches hinreichend flexibel angelegt ist, das breite Spektrum der verschiedensten Kopplungsmechanismen von Systemkomponenten abzudecken. Die im zweiten Kapitel eingeführten asynchronen Produktautomaten (APA) sind eine Verallgemeinerung asynchroner Zellularautomaten /Zie89/ und bilden gewissermaßen das automatentheoretische Gegenstück zu den Kooperationsprodukten. Sie sind ein sehr

allgemeines und flexibles operationales Beschreibungskonzept für kooperierende Systeme, das unterschiedliche syntaktische Ausprägungen erlaubt. Darunter fallen die verschiedenen Arten "einfacher" und "höherer" Petrinetze /OP95/, aber auch alle Arten kommunizierender Automaten wie z. B. SDL /SSR89/ oder Estelle /BD87/.

Ebenso wie andere Verifikationsansätze für verteilte Systeme ist auch der hier vorgestellte mit dem Problem der Zustandsexplosion konfrontiert. Daher wird mit den Sätzen des ersten und dritten Kapitels eine Methode entwickelt, die es unter bestimmten Voraussetzungen erlaubt, homomorphe Bilder von Kooperationsprodukten auf eine kompositionelle Weise zu bestimmen. Bei diesem Vorgehen können die einzelnen Systemkomponenten vollständig separat oder in "abstrakte Umgebungen eingebettet" betrachtet werden, was Nebenläufigkeiten vermeidet, die irrelevant für die homomorphen Bilder sind, aber erheblich zur Zustandsexplosion beitragen. Auch die Schlichtheit von Homomorphismen auf Kooperationsprodukten kann auf diese Weise überprüft werden. Dazu wird im vierten Kapitel der Begriff der Kooperativität der Komponenten von Kooperationsprodukten bezüglich eines Homomorphismus eingeführt.

Zusammen mit den Sätzen der letzten beiden Kapitel liegt dann eine kompositionelle Methode vor, die iterativ benutzt werden kann und, wie an einem Beispiel demonstriert wird, bei parametrisierten Systemspezifikationen auch Induktionsbeweise ermöglicht. Gerade dieses iterative Vorgehen kann zu erheblichen Komplexitätsreduktionen führen und zeigt, wie globale Systemeigenschaften aus lokalen Eigenschaften der Komponenten abgeleitet werden können.

Der in dieser Arbeit vorgestellte Verifikationsansatz ist sowohl mit Ansätzen aus dem Gebiet der Prozessalgebren /BW90,Mil90/ als auch mit solchen, die in direkter Weise ω -Sprachen benutzen /AH95,Kur93/, vergleichbar. Im Kontext kooperierender Systeme liegt sein Vorteil in dem äußerst flexiblen Abstraktionsbegriff, der besonders bei einem iterativen Vorgehen zum Tragen kommt. Dagegen sind in diesem Zusammenhang die Äquivalenzbegriffe der Prozessalgebren oft zu fein /Och94a/, und Ansätze wie in /AH95,Kur93/ verfügen nicht in einem ausreichendem Maße über Abstraktionskonzepte. Insgesamt zeigt sich, daß die klassische Grundidee der formalen Sprachen und Homomorphismen zu einem Verifikationsinstrumentarium für kooperierende Systeme führt, welches bei wohlstrukturierten Systemspezifikationen die Komplexität realer Anwendungen durchaus beherrschbar macht.

1 Kooperationsprodukte

Die in diesem Papier betrachteten formalen Sprachen beschreiben “Verhaltensweisen von Systemen” (Mengen möglicher “Aktionsfolgen”), und die Homomorphismen beschreiben “Abstraktionen” von solchen Verhaltensweisen. Deshalb wird generell vorausgesetzt, daß die Sprachen *präfixstabil* und die Homomorphismen *alphabetisch* sind. D.h. jeder Präfix eines Wortes aus einer Sprache liegt ebenfalls in der Sprache, und für jeden Homomorphismus $f : \Phi^* \rightarrow \Phi'^*$ gilt $f(\Phi) \subset \Phi' \cup \{\varepsilon\}$. Außerdem sei bemerkt, daß die Alphabete nicht notwendigerweise als endlich vorausgesetzt werden.

Zur Motivation betrachten wir folgendes Beispiel eines kooperierenden Systems:

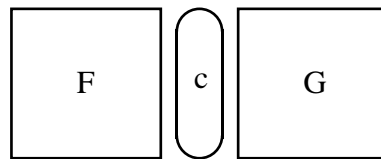


Abb. 1

Ein Server F bearbeitet bestimmte Aufträge und benötigt dazu jeweils bestimmte Ressourcen, welche von einem Resource-Manager G verwaltet werden. F muß also bei G Ressourcen anfordern und diese nach Bearbeitung eines Auftrags auch wieder freigeben. Dazu bedienen sich F und G eines Kommunikationsmediums c. Das Verhalten von F bzw. von G ist durch den Automaten in Abb. 2 bzw. Abb. 3 beschrieben. Dabei ist 1 der jeweilige Anfangszustand; das sei generell auch für alle anderen Automaten vorausgesetzt, falls nichts anderes vereinbart wird.

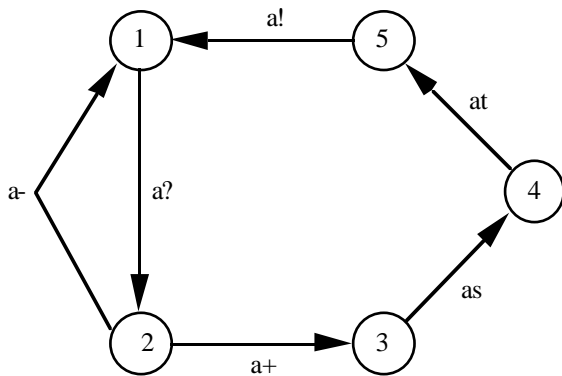


Abb. 2

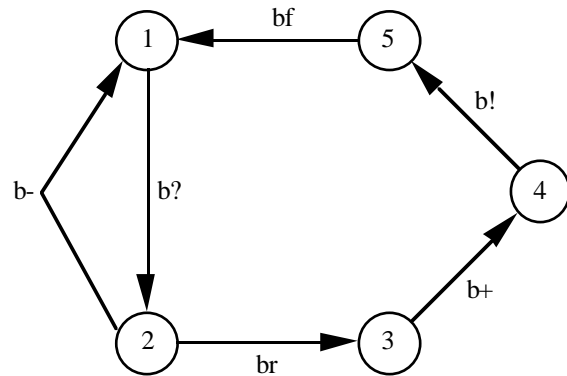


Abb. 3

Die mit ?, +, - und ! gekennzeichneten Aktionen beziehen sich auf die Kommunikation; ? bedeutet Beantragung einer Resource, + bzw. - ist die positive bzw. negative Antwort auf den Antrag, und mit ! wird mitgeteilt, daß die Resource nicht mehr benötigt wird. Aus dem Kontext wird jeweils klar, ob es sich um eine “Sende-” oder “Empfangsaktion” handelt. Die restlichen Aktionen sind “interne” Aktionen. Mit as bzw. at beginnt bzw. beendet F die Bearbeitung eines Auftrags. Mit br bzw. bf wird eine Resource reserviert bzw. freigegeben.

Das Kommunikationsmedium c sei ein gemeinsamer Puffer, der maximal ein Element aus der “Nachrichtenmenge” {?, +, -, !} aufnehmen kann (Kapazität 1). Kennzeichnet man das Hineinlegen einer entsprechenden Nachricht in den leeren Puffer mit dem Präfix s und das Herausnehmen einer Nachricht aus dem vollen Puffer mit r, dann beschreibt der Automat in Abb. 4 die möglichen “Aktionsfolgen am

Kommunikationspuffer”.

Man kann sich nun leicht überlegen, daß durch diese Festlegung das globale Systemverhalten, welches sowohl die Aktionen von F als auch die von G in ihrem Zusammenspiel berücksichtigt, durch den Automaten in Abb. 5 beschrieben wird.

An Stelle dieser Automaten kann man auch die formalen Sprachen betrachten, welche von den Automaten erkannt werden. Dabei sind alle Zustände Endzustände; das sei generell auch für alle anderen Automaten vorausgesetzt, falls nichts anderes vereinbart wird. Mit $\Phi = \{a?, a+, a-, as, at, a!\}$, $\Gamma = \{b?, b+, b-, br, bf, b!\}$ und $\Sigma = \{s?, s+, s-, s!, r?, r+, r-, r!\}$ seien $F \subset \Phi^*$, $G \subset \Gamma^*$ und $C \subset \Sigma^*$ die Sprachen, welche von den Automaten in Abb. 2, 3 und 4 erkannt werden. Unter Benutzung spezieller Homomorphismen kann bei dieser Sichtweise die intuitive Konstruktion des globalen Systemverhaltens formal beschrieben werden. Dies verlangt noch einige Definitionen:

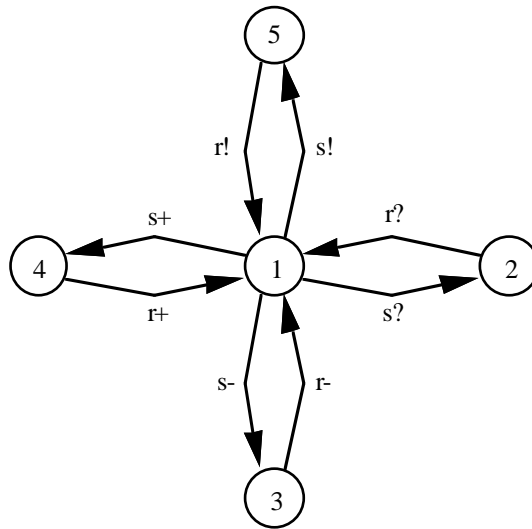


Abb. 4

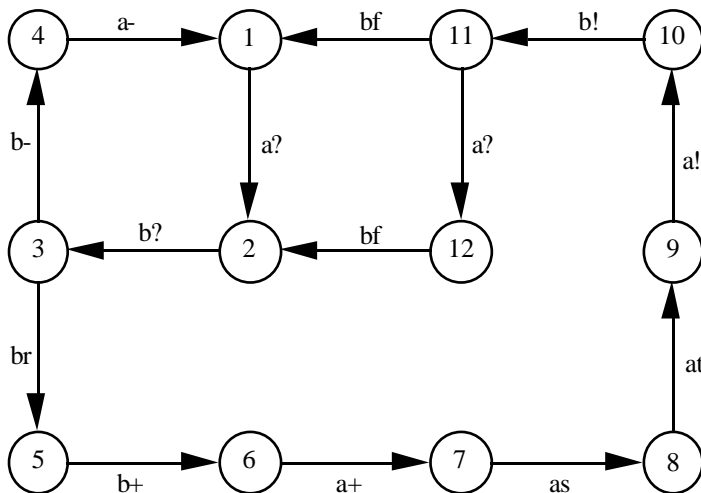


Abb. 5

Für ein Alphabet Φ sei $I_\Phi : \Phi^* \rightarrow \Phi^*$ der *Identitätshomomorphismus*, der jedes Wort auf sich selbst abbildet und $E_\Phi : \Phi^* \rightarrow \emptyset^*$ der *Epsilonhomomorphismus*, der jedes Wort auf das leere Wort ε abbildet ($\emptyset^* = \{\varepsilon\}$). Für zwei Homomorphismen $f : \Phi^* \rightarrow \Phi'^*$ und $g : \Gamma^* \rightarrow \Gamma'^*$ mit $\Phi \cap \Gamma = \emptyset$ sei die *direkte Summe* $[f,g] : (\Phi \cup \Gamma)^* \rightarrow (\Phi' \cup \Gamma')^*$ definiert durch $[f,g](x) = f(x)$ falls $x \in \Phi$ und $[f,g](x) = g(x)$ falls $x \in \Gamma$. Aus der Definition folgt unmittelbar die Kommutativität und Assoziativität der direkten Summe von Homomorphismen. Es gilt $[I_\Phi, I_\Gamma] = I_{\Phi \cup \Gamma}$ sowie $[E_\Phi, E_\Gamma] = E_{\Phi \cup \Gamma}$, und falls $\Phi' \cap \Gamma' = \emptyset$, dann ist $f \circ [I_\Phi, E_\Gamma] = [I_{\Phi'}, E_{\Gamma'}] \circ [f,g]$ und $g \circ [E_\Phi, I_\Gamma] = [E_{\Phi'}, I_{\Gamma'}] \circ [f,g]$.

Als formale Sprache betrachtet ist das globale Systemverhalten eine Teilmenge von $(\Phi \cup \Gamma)^*$; es ist sogar eine Teilmenge von $[I_\Phi, E_\Gamma]^{-1}(F) \cap [E_\Phi, I_\Gamma]^{-1}(G)$, da es sich ja aus den lokalen Verhaltensweise F und G "zusammensetzt". Diese Teilmengenbeziehung ist natürlich i.allg. echt, da $[I_\Phi, E_\Gamma]^{-1}(F) \cap [E_\Phi, I_\Gamma]^{-1}(G)$ noch nicht die Kommunikation zwischen F und G berücksichtigt. Zur Beschreibung der jeweiligen Aktionen von F und G "am Kommunikationsmedium" betrachten wir die Homomorphismen $\phi : \Phi^* \rightarrow \Sigma^*$ und $\gamma : \Gamma^* \rightarrow \Sigma^*$ mit $\phi(a?) = s?$, $\phi(a+) = r+$, $\phi(a-) = r-$, $\phi(a!) = s!$ und $\phi(as) = \phi(at) = \varepsilon$ sowie $\gamma(b?) = r?$, $\gamma(b+) = s+$, $\gamma(b-) = s-$, $\gamma(b!) = r!$ und $\gamma(br) = \gamma(bf) = \varepsilon$. Der Einfluß der Kommunikation zwischen F und G auf das globale Systemverhalten wird jetzt durch $[\phi, \gamma]^{-1}(C)$ erfaßt. Insgesamt wird also das globale Systemverhalten durch die Sprache $[I_\Phi, E_\Gamma]^{-1}(F) \cap [E_\Phi, I_\Gamma]^{-1}(G) \cap [\phi, \gamma]^{-1}(C)$ dargestellt. Dies motiviert folgende Definitionen:

Seien $\phi : \Phi^* \rightarrow \Sigma^*$ und $\gamma : \Gamma^* \rightarrow \Sigma^*$ zwei Homomorphismen mit $\Phi \cap \Gamma = \emptyset$, und sei $C \subset \Sigma^*$, dann definiert $c = (\Phi, \Gamma, \Sigma, \phi, \gamma, C)$ eine *Kooperationsform*. Den Spezialfall $\Sigma = \emptyset$ und $C = \emptyset^*$ nennen wir die *leere Kooperationsform* und bezeichnen sie für alle Alphabete Φ und Γ der Kürze wegen mit \diamond , also $\diamond = (\Phi, \Gamma, \emptyset, E_\Phi, E_\Gamma, \emptyset^*)$.

Das *Kooperationsprodukt* $[F,G]_c$ zweier formaler Sprachen $F \subset \Phi^*$ und $G \subset \Gamma^*$ bezüglich der Kooperationsform c ist durch $[F,G]_c = [I_\Phi, E_\Gamma]^{-1}(F) \cap [E_\Phi, I_\Gamma]^{-1}(G) \cap [\phi, \gamma]^{-1}(C)$ definiert. Aus dieser Definition folgt unmittelbar $[F,G]_c \subset (\Phi \cup \Gamma)^*$ sowie die Kommutativität des Kooperationsproduktes. Speziell für \diamond gilt also $[F,G]_\diamond = [I_\Phi, E_\Gamma]^{-1}(F) \cap [E_\Phi, I_\Gamma]^{-1}(G) \cap [E_\Phi, E_\Gamma]^{-1}(\emptyset^*) = [I_\Phi, E_\Gamma]^{-1}(F) \cap [E_\Phi, I_\Gamma]^{-1}(G) \cap (\Phi \cup \Gamma)^* = [I_\Phi, E_\Gamma]^{-1}(F) \cap [E_\Phi, I_\Gamma]^{-1}(G)$. Das ist das übliche *Shuffleprodukt* von F und G /Eil74/. Es liefert die Darstellung $[F,G]_c = [F,G]_\diamond \cap [\phi, \gamma]^{-1}(C)$. Diese Darstellung macht deutlich, daß es sich bei dem Kooperationsprodukt um eine Art Shuffleprodukt handelt, welches zusätzlich noch durch die Homomorphismen ϕ und γ sowie die Sprache C "kontrolliert" wird.

Interessiert man sich dafür, ob ein aus Teilsystemen zusammengebautes komplexes System wirklich das "erwartete" globale Systemverhalten zeigt (*Verifikation* genannt), dann müssen dafür in aller Regel nur bestimmte Aktionen betrachtet werden. Oft genügt es, ein homomorphes Bild des Kooperationsproduktes zu inspizieren. So kann man in unserem Beispiel dann von einem korrekten globalen Systemverhalten sprechen, wenn vor jeder Bearbeitung eines Auftrags eine entsprechende Resource reserviert und nach Bearbeitung des Auftrags wieder freigegeben wird. Es muß also $v([F,G]_c)$ betrachtet werden, wobei $v : (\Phi \cup \Gamma)^* \rightarrow \Xi^*$ mit $\Xi = \{as, at, br, bf\}$ und $v = [I_\Xi, E_{(\Phi \cup \Gamma) \setminus \Xi}]$. Aus Abb. 5 läßt sich leicht der in Abb. 6 dargestellte Minimalautomat für $v([F,G]_c)$ bestimmen. Dieser zeigt in der Tat das gewünschte Verhalten.

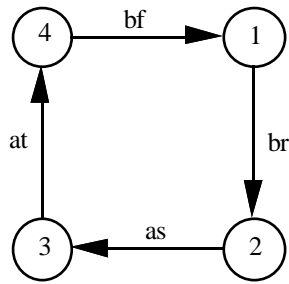


Abb. 6

Betrachtet man die Zustandszahl der Automaten (im Fall von regulären Sprachen), dann können Kooperationsprodukte schon recht einfacher Sprachen durch einen multiplikativen Effekt schon sehr komplex werden. Dies ist unter dem Schlagwort *Explosion des Zustandsraums* wohlbekannt. Zur Verdeutlichung dieses Phänomens betrachten wir eine leichte Modifikation unseres Beispiels: Der Server möge, wie in Abb. 7 dargestellt, nach Anforderung einer Resource noch eine interne Aktion a_i durchführen, bevor er die Antwort vom Resource-Manager entgegennimmt. Der Resource-Manager kann, wie Abb. 8 zeigt, mit einer internen Aktion b_n von seinem Grundzustand aus in einen Exception-Zustand übergehen, in welchem er jede Resource-Anfrage negativ beantwortet. Aus diesem Exception-Zustand kann er mit einer anderen internen Aktion b_i wieder in den Grundzustand gelangen.

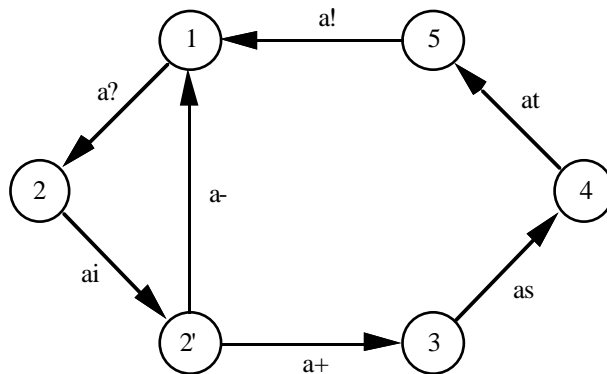


Abb. 7

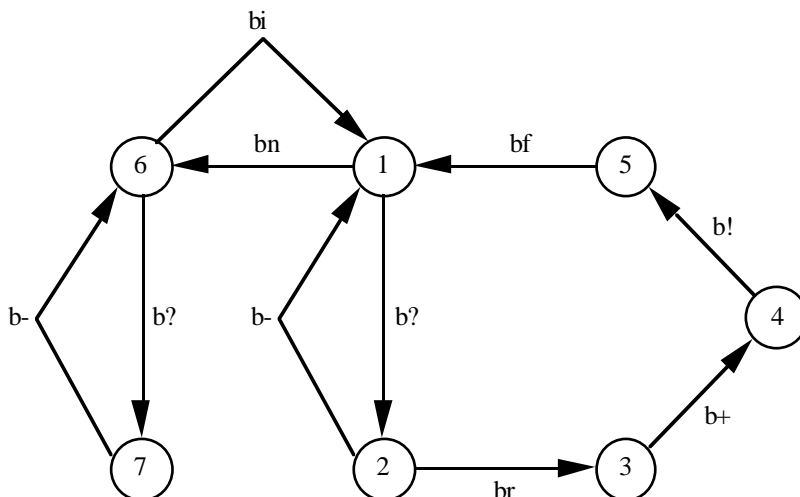


Abb. 8

Die Automaten in Abb. 7 und 8 definieren Sprachen $F1 \subset \Phi1^*$ und $G1 \subset \Gamma1^*$ mit $\Phi1 = \Phi \cup \{a_i\}$ und $\Gamma1 = \Gamma \cup \{b_n, b_i\}$. Für die Kooperationsform $c1 = (\Phi1, \Gamma1, \Sigma, \phi1, \gamma1, C)$ seien die Homomorphismen ϕ und γ zu Homomorphismen $\phi1 : \Phi1^* \rightarrow \Sigma^*$ und $\gamma1 : \Gamma1^* \rightarrow \Sigma^*$ fortgesetzt mit $\phi1(a_i) = \gamma1(b_n) =$

$\gamma_1(\text{bi}) = \varepsilon$. Der in Abb. 9 dargestellte Minimalautomat von $[F_1, G_1]_{C_1}$ ist also durch kleine Modifikationen von F und G wesentlich komplexer als der von $[F, G]_C$. (Zur Vermeidung von Kantengewirr ist dieser Automat durch "Aufspalten seiner Zustände" dargestellt; gleich benannte Zustände sind dabei zu identifizieren.) Vor diesem Hintergrund ist es also für Verifikationszwecke von großem Interesse, daß zur Bestimmung homomorpher Bilder von Kooperationsprodukten unter bestimmten Voraussetzungen diese komplexen Kooperationsprodukte nicht bekannt sein müssen.

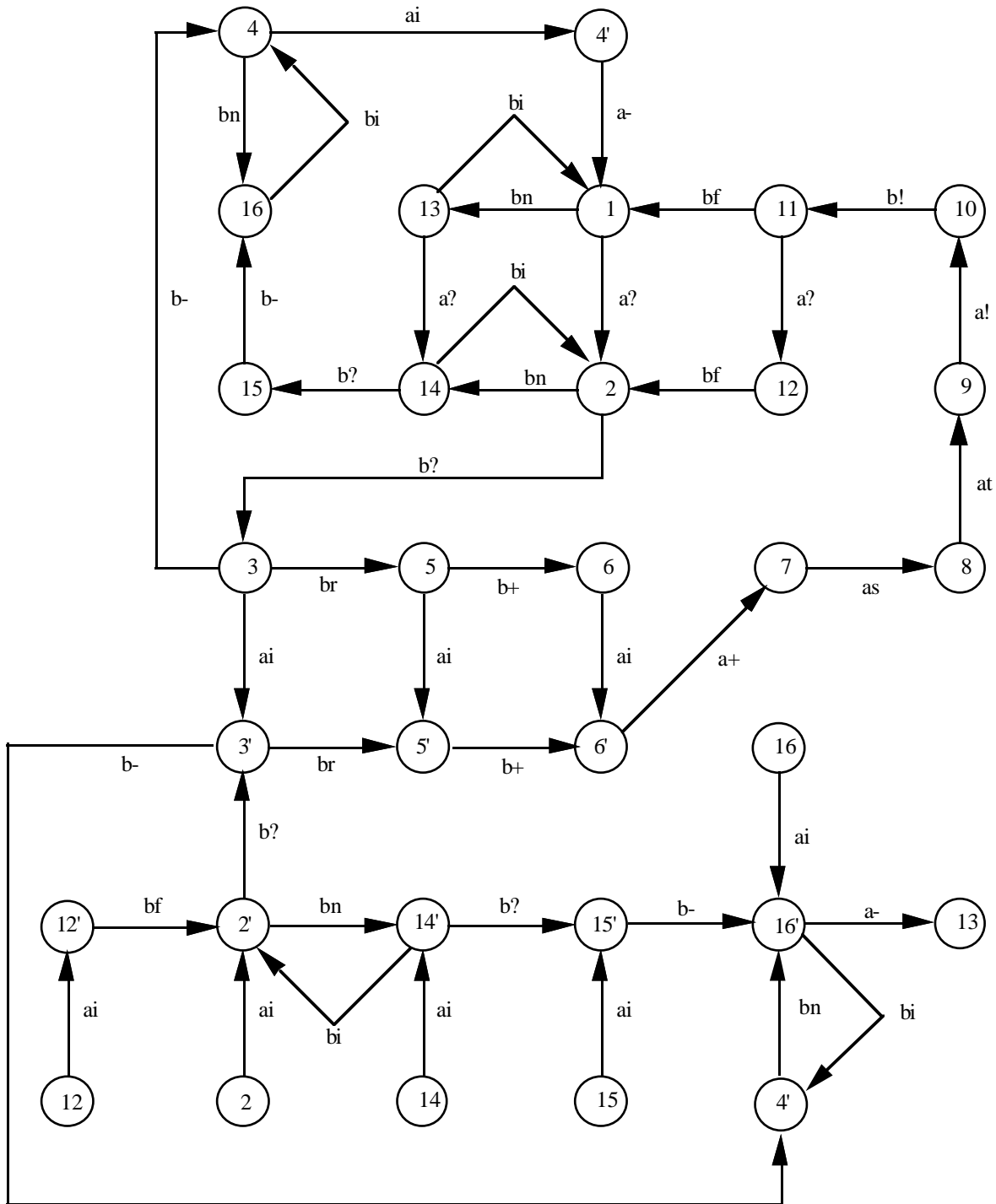


Abb. 9

Zur Formulierung eines entsprechenden Satzes sei $c = (\Phi, \Gamma, \Sigma, \phi, \gamma, C)$ eine Kooperationsform und $f : \Phi^* \rightarrow \Phi'^*$ sowie $g : \Gamma^* \rightarrow \Gamma'^*$ zwei Homomorphismen mit $f \prec \phi$ (f ist *feiner* als ϕ), $g \prec \gamma$ und $\Phi' \cap \Gamma' = \emptyset$. Unter dieser Voraussetzung gibt es Homomorphismen $\phi' : \Phi'^* \rightarrow \Sigma^*$ und $\gamma' : \Gamma'^* \rightarrow \Sigma^*$ mit $\phi = \phi' \circ f$ und $\gamma = \gamma' \circ g$, also $[\phi, \gamma] = [\phi', \gamma'] \circ [f, g]$. Sei c' die Kooperationsform, welche definiert ist durch $c' = (\Phi', \Gamma', \Sigma, \phi', \gamma', C)$. Mit diesen Voraussetzungen gilt dann

Satz 1: $[f, g]([F, G]_c) = [f(F), g(G)]_{c'}$

Beweis:

Sei $x \in [F, G]_c$. Aus $f \circ [I_\Phi, E_\Gamma] = [I_{\Phi'}, E_{\Gamma'}] \circ [f, g]$ und $x \in [I_\Phi, E_\Gamma]^{-1}(F)$ folgt $[f, g](x) \in [I_{\Phi'}, E_{\Gamma'}]^{-1}(f(F))$, aus $g \circ [E_\Phi, I_\Gamma] = [E_{\Phi'}, I_{\Gamma'}] \circ [f, g]$ und $x \in [E_\Phi, I_\Gamma]^{-1}(G)$ folgt $[f, g](x) \in [E_{\Phi'}, I_{\Gamma'}]^{-1}(g(G))$, und aus $[\phi, \gamma] = [\phi', \gamma'] \circ [f, g]$ und $x \in [\phi, \gamma]^{-1}(C)$ folgt $[f, g](x) \in [\phi', \gamma']^{-1}(C)$. Damit gilt $[f, g]([F, G]_c) \subset [f(F), g(G)]_{c'}$. Zu jedem $y \in [f(F), g(G)]_{c'}$ existiert ein $x \in [f, g]^{-1}(y)$ mit $[I_\Phi, E_\Gamma](x) \in F$ und $[E_\Phi, I_\Gamma](x) \in G$. Wegen $[\phi, \gamma](x) = [\phi', \gamma']([f, g](x)) = [\phi', \gamma'](y) \in C$ folgt daraus $[f(F), g(G)]_{c'} \subset [f, g]([F, G]_c)$, was den Beweis vervollständigt.

Soll mit diesem Satz ein beliebiges homomorphes Bild eines Kooperationsproduktes bestimmt werden, dann muß ggf. der betrachtete Homomorphismus noch dahingehend verfeinert werden, daß er eine Darstellung der Form $[f, g]$ besitzt, welche die Voraussetzungen des Satzes erfüllt. Im betrachteten Beispiel führt dies für $v([F, G]_c)$ zu keiner Komplexitätsreduktion, denn es müßte $f = I_\Phi$ und $g = I_\Gamma$ gewählt werden.

Für eine entsprechende Verifikation von $[F1, G1]_{c1}$ muß der Homomorphismus v durch $v1(ai) = v1(bn) = v1(bi) = \varepsilon$ zu einem Homomorphismus $v1 : (\Phi1 \cup \Gamma1)^* \rightarrow \Xi^*$ fortgesetzt werden. Mit Abb. 9 läßt sich nachprüfen, daß $v1([F1, G1]_{c1}) = v([F, G]_c)$, was die zu zeigende Eigenschaft von $v1([F1, G1]_{c1})$ nachweist. Mit Satz1 kann $v1([F1, G1]_{c1})$ bestimmt werden, ohne vorher das komplexe Kooperationsprodukt $[F1, G1]_{c1}$ zu berechnen. Dazu seien die Homomorphismen $f1 : \Phi1^* \rightarrow \Phi^*$ und $g1 : \Gamma1^* \rightarrow \Gamma^*$ mit $f1 = [I_\Phi, E_{\{ai\}}]$ und $g1 = [I_\Gamma, E_{\{bn, bi\}}]$ definiert. Diese erfüllen die Voraussetzungen von Satz 1, und es gilt dann $[f1, g1]([F1, G1]_{c1}) = [f1(F1), g1(G1)]_c = [F, G]_c$. Beachtet man $v1 = v \circ [f1, g1]$, dann folgt jetzt $v1([F1, G1]_{c1}) = v([f1, g1]([F1, G1]_{c1})) = v([f1(F1), g1(G1)]_c) = v([F, G]_c)$.

Zur Anwendung dieses Verfahrens auf komplexe Systemspezifikationen muß in aller Regel das System in mehr als zwei Teilsysteme aufgespalten werden. Damit dabei nicht spezielle "Klammerungen" berücksichtigt werden müssen, ist die im nachfolgenden Satz gezeigte Assoziativität des Kooperationsproduktes hilfreich.

Satz 2:

Sei $\Phi \cap \Gamma = \Gamma \cap \Pi = \Pi \cap \Phi = \emptyset$, $F \subset \Phi^*$, $G \subset \Gamma^*$, $P \subset \Pi^*$, $C \subset \Sigma^*$, $D \subset \Delta^*$ und $\phi : \Phi^* \rightarrow \Sigma^*$, $\gamma : \Gamma^* \rightarrow \Sigma^*$, $\delta : \Gamma^* \rightarrow \Delta^*$ sowie $\pi : \Pi^* \rightarrow \Delta^*$. Mit den Kooperationsformen $c = (\Phi, \Gamma, \Sigma, \phi, \gamma, C)$, $c \sim = (\Phi, \Gamma \cup \Pi, \Sigma, \phi, [\gamma, E_\Pi], C)$, $d = (\Gamma, \Pi, \Delta, \delta, \pi, D)$ und $d \sim = (\Phi \cup \Gamma, \Pi, \Delta, [E_\Phi, \delta], \pi, D)$ gilt $[[F, G]_c, P]_{d \sim} = [F, [G, P]_d]_{c \sim}$.

Beweis:

Wegen der vorausgesetzten Disjunktheit der Alphabete läßt sich leicht nachrechnen, daß $[[F, G]_c, P]_{d \sim} = [I_\Phi, E_{\Gamma \cup \Pi}]^{-1}(F) \cap [I_\Gamma, E_{\Phi \cup \Pi}]^{-1}(G) \cap [I_\Pi, E_{\Phi \cup \Gamma}]^{-1}(P) \cap [[\phi, \gamma], E_\Pi]^{-1}(C) \cap [[\delta, \pi], E_\Phi]^{-1}(D) =$

$[F, [G, P]_d]_{c^-}$.

Zur Demonstration der iterativen Anwendung von Satz 1 erweitern wir, wie in Abb. 10 dargestellt, unser Beispiel um ein drittes Teilsystem P. In diesem komplexeren System ist P zusammen mit G1 ein verteilter Resource- Manager. Bezüglich der Anforderung einer Resource besitzt P einerseits eine "Weiterleitungsfunktion", kann aber andererseits Anforderungen auch direkt ablehnen. Das konkrete Verhalten von P ist in Abb. 11 dargestellt.

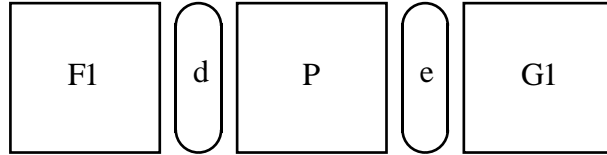


Abb. 10

Die Aktionen cn und ci sind interne Aktionen; sie entsprechen bn und bi bei G1. Alle anderen Aktionen beziehen sich auf die Kommunikation, wobei $c?$, $c+$, $c-$ sowie $c!$ die Schnittstelle d und $c??$, $c?+$, $c?-$ sowie $c?!$ die Schnittstelle e betreffen.

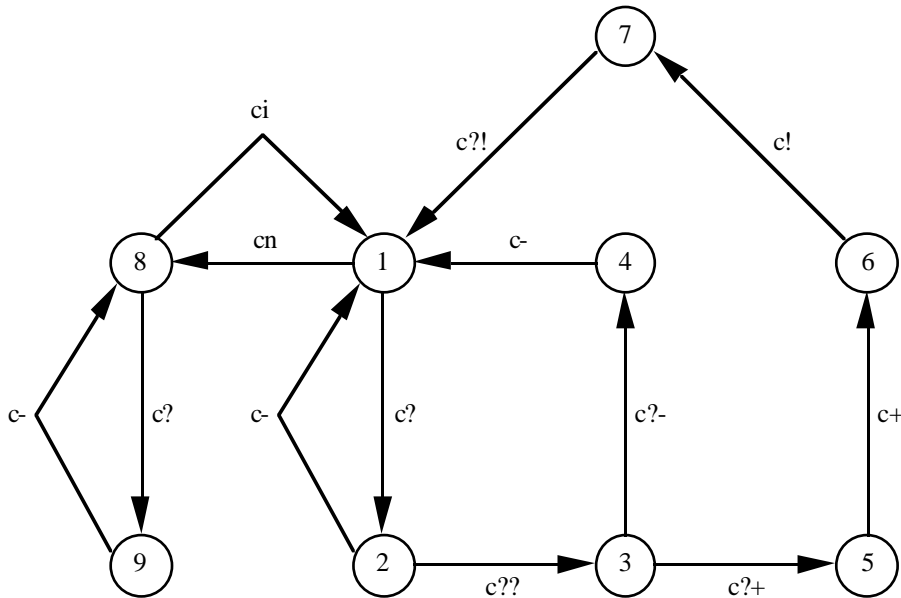


Abb. 11

Zur Darstellung als Kooperationsprodukt sei also $\Pi = \{cn, ci, c?, c+, c-, c!, c??, c?+, c?-, c?!\}$ und $P \subset \Pi^*$ die durch Abb.11 definierte Sprache. Für die Kooperationsformen $d = (\Phi_1, \Pi, \Sigma, \phi_1, \alpha, C)$ und $e = (\Phi_1 \cup \Pi, \Gamma_1, \Sigma, \beta, \gamma_1, C)$ seien die Homomorphismen $\alpha : \Pi^* \rightarrow \Sigma^*$ und $\beta : (\Phi_1 \cup \Pi)^* \rightarrow \Sigma^*$ durch $\alpha(c?) = r?$, $\alpha(c+) = s+$, $\alpha(c-) = s-$, $\alpha(c!) = r!$, $\alpha(\Pi \setminus \{c?, c+, c-, c!\}) = \{\epsilon\}$, $\beta(c??) = s?$, $\beta(c?+) = r+$, $\beta(c?-) = r-$, $\beta(c?!) = s!$ und $\beta((\Phi_1 \cup \Pi) \setminus \{c??, c?+, c?-, c?!\}) = \{\epsilon\}$ definiert. Das globale Systemverhalten von Abb. 10 wird jetzt durch das Kooperationsprodukt $[[F_1, P]_d G_1]_e$ beschrieben. Zum Nachweis der korrekten Zuteilung und Freigabe der Resource muß $v_1([[\Phi_1 \cup \Gamma_1, E_\Pi]]([F_1, P]_d G_1]_e)) = v([F, G]_c)$ gezeigt werden.

Eine passende Darstellung von $(v_1 \circ [[\Phi_1 \cup \Gamma_1, E_\Pi]]([F_1, P]_d G_1]_e))$ ermöglicht die Anwendung des Satzes 1. Sei dazu der Homomorphismus $h : (\Phi_1 \cup \Pi)^* \rightarrow \Phi^*$ durch $h(as) = as$, $h(at) = at$, $h(c??) = a?$, $h(c?+) = a+$, $h(c?-) = a-$, $h(c?!) = a!$ und $h((\Phi_1 \cup \Pi) \setminus \{as, at, c??, c?+, c?-, c?!\}) = \{\epsilon\}$ definiert. Dann gilt $v_1 \circ [[\Phi_1 \cup \Gamma_1, E_\Pi]] = v \circ [h, g_1]$. Da $h \langle \beta$ und $g_1 \langle \gamma_1$ folgt jetzt mit Satz 1

$v1(\Pi_{\Phi_1 \cup \Gamma_1}, E_{\Pi}][([F1, P]_d, G1]_e)) = v([h([F1, P]_d), g1(G1)]_c) = v([h([F1, P]_d), G1]_c)$. Es muß also nur noch $h([F1, P]_d) = F$ gezeigt werden.

Mit dem durch $k(as) = as$, $k(at) = at$, $k(c??) = a?$, $k(c?+) = a+$, $k(c?-) = a-$, $k(c?!) = a!$ und $k((\Phi \cup \Pi) \setminus \{as, at, ci, cn, c??, c?+, c?-, c?! \}) = \{\varepsilon\}$ definierten Homomorphismus $k : (\Phi \cup (\Pi \setminus \{ci, cn\}))^* \rightarrow \Phi^*$ gilt $h = k \circ [f1, [\Pi \setminus \{ci, cn\}, E_{\{ci, cn\}}]]$. Aus Satz 1 folgt jetzt $h([F1, P]_d) = k([f1, [\Pi \setminus \{ci, cn\}, E_{\{ci, cn\}}]]([F1, P]_d)) = k([f1(F1), [\Pi \setminus \{ci, cn\}, E_{\{ci, cn\}}](P)]_d) = k([F, [\Pi \setminus \{ci, cn\}, E_{\{ci, cn\}}](P)]_d)$. Abb. 12 zeigt die Automattendarstellung von $[\Pi \setminus \{ci, cn\}, E_{\{ci, cn\}}](P)$. Daraus kann die Automattendarstellung von $[F, [\Pi \setminus \{ci, cn\}, E_{\{ci, cn\}}](P)]_d$ in Abb. 13 berechnet werden. An diesem Automaten läßt sich dann leicht $h([F1, P]_d) = k([F, [\Pi \setminus \{ci, cn\}, E_{\{ci, cn\}}](P)]_d) = F$ verifizieren.

Unter Benutzung von Satz 1 kann also eine Automattendarstellung von $v1(\Pi_{\Phi_1 \cup \Gamma_1}, E_{\Pi}][([F1, P]_d, G1]_e))$ berechnet werden, wobei Automaten mit maximal 14 Zuständen benötigt werden. Dem steht gegenüber, daß der Minimalautomat von $[[F1, P]_d, G1]_e$ 89 Zustände besitzt. Ersetzt man, wie Abb. 14 zeigt, in Abb. 10 das Teilsystem P durch eine "Kette von Kopien" von P, dann kann der obige Beweis ohne zusätzliche Automatenberechnung zu einem Induktionsbeweis für die korrekte Zuteilung und Freigabe der Resource in diesem parametrisierten System umformuliert werden. Der Beweis von $h([F1, P]_d) = F$ muß dabei lediglich als Induktionsschritt interpretiert werden. In /Och94b/ wird ein ähnlicher Induktionsbeweis für ein komplexes Verbindungsauf- und -abbauprotokoll durchgeführt.

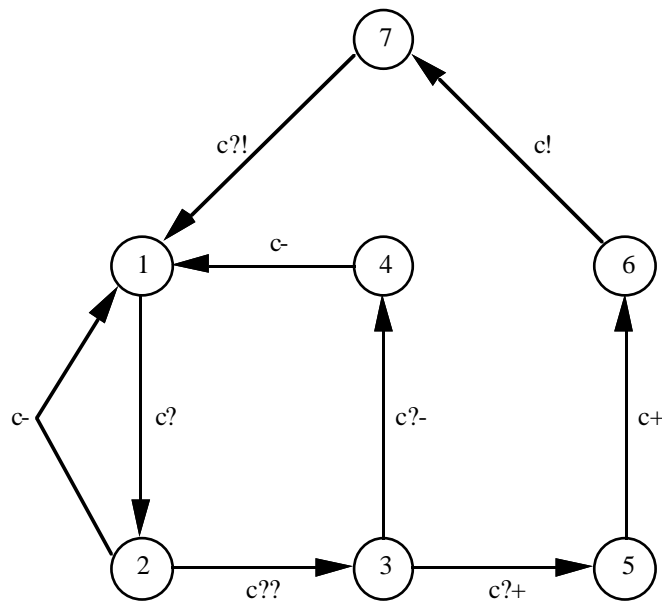


Abb. 12

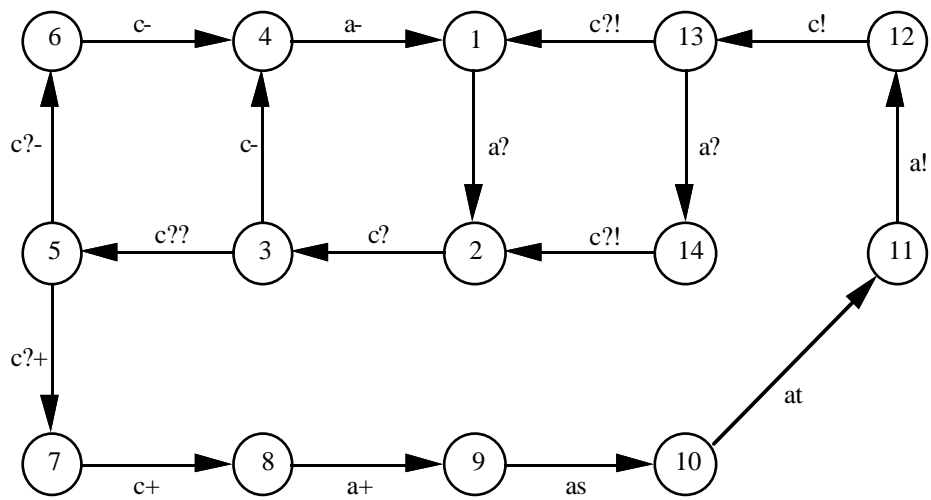


Abb. 13

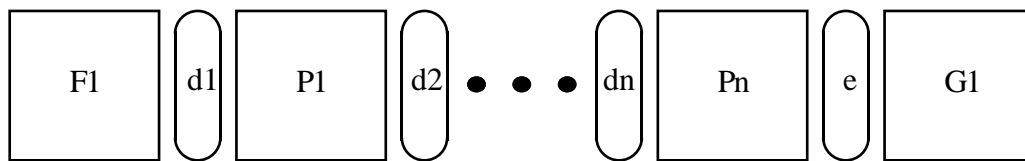


Abb. 14

2 Asynchrone Produktautomaten

Aus der Definition von $[F,G]_C$ folgt unmittelbar, daß $[I_\Phi, E_\Gamma]([F,G]_C) \subset F$ und $[E_\Phi, I_\Gamma]([F,G]_C) \subset G$. Wie man sich leicht überlegen kann, ist diese Inklusion i. allg. echt. Es kann also sein, daß schon echte Teilmengen von F und G das Kooperationsprodukt $[F,G]_C$ "vollständig aufspannen". $[I_\Phi, E_\Gamma]([F,G]_C)$ und $[E_\Phi, I_\Gamma]([F,G]_C)$ sind darunter die kleinsten, denn es läßt sich leicht verifizieren, daß $[F,G]_C = [[I_\Phi, E_\Gamma]([F,G]_C), [E_\Phi, I_\Gamma]([F,G]_C)]_C$. Entsprechendes gilt auch für die Sprache C der Kooperationsform c : $[\phi, \gamma]([F,G]_C) \subset C$, und $[F,G]_C = [F,G]_{C \sim}$, wobei $c \sim = (\Phi, \Gamma, \Sigma, \phi, \gamma, [\phi, \gamma]([F,G]_C))$. $[\phi, \gamma]([F,G]_C)$ wird *Schnittstellensprache* von $[F,G]_C$ genannt.

In dem bisher betrachteten Beispiel waren F , G und C in einem gewissen Sinne minimal, denn zur "Erzeugung" von $[F,G]_C$ wurden alle ihre Elemente "benötigt". Beim Entwurf komplexer kooperierender Systeme kennt man in aller Regel nicht a priori diese minimalen Sprachen, da sie vom globalen Systemverhalten abhängen, was ja gerade Gegenstand der Untersuchung ist. In vielen Fällen ist es einfacher, für das Verhalten von Systemkomponenten echte Obermengen dieser minimalen Sprachen anzugeben. Das nachfolgende Beispiel wird diesen Sachverhalt verdeutlichen.

Dazu betrachten wir eine einfache Version eines Verbindungsauf- und -abbauprotokolls. Das Protokoll beschreibt das Zusammenspiel eines Senders F und eines Empfängers G , welche zum Nachrichtentransport in beiden Richtungen jeweils durch eine FIFO-Queue verbunden sind. Die Grobstruktur des Systems ist in Abb. 15 dargestellt.

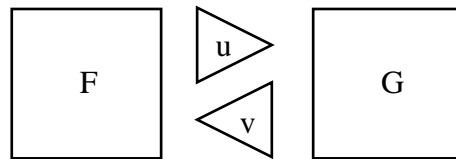


Abb. 15

Wenn der Sender F zum Empfänger G eine Verbindung aufbauen möchte, dann teilt er dies G mittels der Nachricht C (connect) mit. G kann diesen Wunsch mit einem D (disconnect) ablehnen, falls er z.B. momentan über keine freien Betriebsmittel für diese Verbindung verfügt, oder er bestätigt mit der Nachricht C an F diesen Verbindungsaufbau. Im positiven Fall befinden sich dann F und G in der sogenannten Datenphase, d.h. G ist nach Absenden eines C bereit, Daten zu empfangen, und F kann nach Empfang eines C Daten senden. Beide Kooperationspartner F und G besitzen unabhängig voneinander jederzeit, d.h. sowohl in der Aufbau- als auch in der Datenphase die Entscheidungsfreiheit zu einem Abbau der Verbindung. Die Verständigung darüber (ebenso wie die Ablehnung eines Verbindungsaufbauwunsches) geschieht durch einen sogenannten symmetrischen Handshake mit der Nachricht D . Das genaue Verhalten von Sender und Empfänger ist durch die Automaten in Abb. 16 und 17 beschrieben. Der Zustand 3 beschreibt in beiden Automaten die Datenphase.

Für eine Darstellung des Systemverhaltens als Kooperationsprodukt werden also die durch Abb. 16 und 17 definierten Sprachen $F \subset \Phi^*$ und $G \subset \Gamma^*$ mit $\Phi = \{FCS, FCR, FDS, FDR\}$ und $\Gamma = \{GCS, GCR, GDS, GDR\}$ betrachtet. Dabei bedeutet FCS , daß F die Nachricht C verschickt, und FCR bedeutet, daß F die Nachricht C empfängt. Die anderen Elemente von Φ und Γ sind entsprechend zu interpretieren.

Zur Definition einer geeigneten Kooperationsform $c = (\Phi, \Gamma, \Sigma, \phi, \gamma, K)$ beachte man, daß alle Aktionen von F und G Aktionen an den beiden FIFO-Queues sind, und daß all diese Aktionen bezüglich der beiden FIFO-Queues voneinander unterschieden werden müssen. Es bietet sich daher $\Sigma = \Phi \cup \Gamma$, $\phi = I_\Phi$ und $\gamma = I_\Gamma$ an. Da ohne Vorabanalyse nicht bekannt ist, wieviele Nachrichten sich in den beiden Queues befinden werden, muß für die Bestimmung von $K \subset \Sigma^*$ zuerst einmal von unbeschränkten Queues ausgegangen werden. Die Sprache K beschreibt also die möglichen Sende- und Empfangsaktionen an zwei entgegengesetzt gerichteten unbeschränkten FIFO-Queues bezüglich der Nachrichten C und D. Man kann sich nun leicht überlegen, daß unter dieser Annahme K sicher keine reguläre Sprache ist.

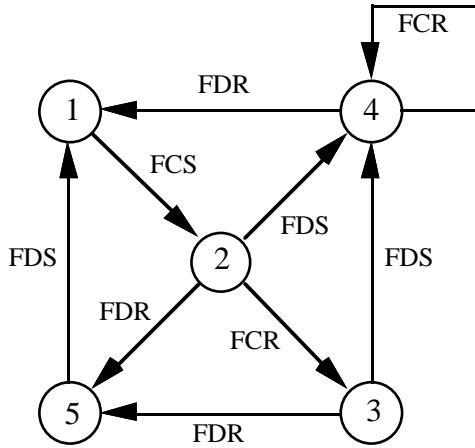


Abb. 16

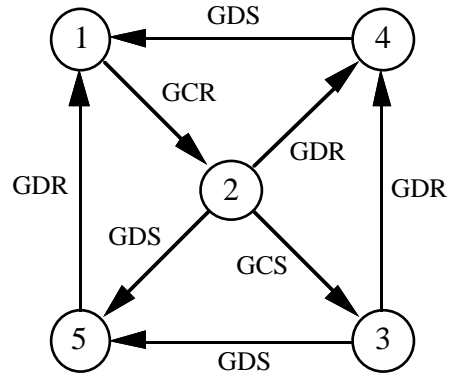


Abb. 17

Eine relativ einfache Automatendarstellung $\mathbb{K} = (\Sigma, S, \Delta, s_0)$ der Sprache $K \subset \Sigma^*$ erhält man dadurch, daß man die Inhalte der beiden FIFO-Queues als Zustandsmenge wählt. Diese ist natürlich bei der getroffenen Annahme unendlich. Sei also die Zustandsmenge S des entsprechenden Automaten gegeben durch $S = \{C,D\}^* \times \{C,D\}^*$ mit dem Anfangszustand $s_0 = (\varepsilon, \varepsilon)$. Spezielle Endzustände werden nicht benötigt, da K präfixstabil ist; alle Zustände können als Endzustände betrachtet werden. Für die Zustandsübergangsrelation $\Delta \subset S \times \Sigma \times S$ gilt dann $\Delta = \bigcup_{u,v \in \{C,D\}^*} \{ ((u,v), FCS, (Cu,v)), ((u,v), FDS, (Du,v)), ((u,Cv), FCR, (u,v)), ((u,Dv), FDR, (u,v)), ((u,v), GCS, (u,vC)), ((u,v), GDS, (u,vD)), ((uC,v), GCR, (u,v)), ((uD,v), GDR, (u,v)) \}$.

Bei endlichen Automaten ist die Zustandsübergangsrelation Δ eine endliche Menge. Sie kann deshalb bildlich als endlicher gerichteter Graph dargestellt werden, dessen Knoten Elemente von S sind und dessen Kanten, welche die Zustandsübergangsrelation Δ repräsentieren, mit Elementen von Σ beschriftet sind. Generell sprechen wir von einer *Automatendarstellung* $\mathbb{L} = (\Sigma, S, \Delta, s_0)$ einer präfixstabilen Sprache $L \subset \Sigma^*$, falls Σ und S beliebige Mengen sind, $\Delta \subset S \times \Sigma \times S$, $s_0 \in S$ und $L = \{ x \in \Sigma^* \mid \text{für } 1 \leq i \leq |x| \text{ existieren } a_i \in \Sigma \text{ und } s_i \in S \text{ mit } (s_{i-1}, a_i, s_i) \in \Delta \text{ und } x = a_1 \dots a_{|x|} \}$. Dabei bezeichnet $|x|$ die Länge eines Wortes $x \in \Sigma^*$.

Es sei bemerkt, daß es zu jeder präfixstabilen Sprache $L \subset \Sigma^*$ eine Automatendarstellung $\mathbb{L} = (\Sigma, S, \Delta, s_0)$ gibt, falls auch unendliche Zustandsmengen S erlaubt sind. Eine geeignete Zustandsmenge liefern die *Linksquotienten* $x^{-1}(L)$ /Eil74/. Für $L \subset \Sigma^*$ und $x \in \Sigma^*$ ist $x^{-1}(L)$ durch $x^{-1}(L) = \{ y \in \Sigma^* \mid xy \in L \}$ definiert. Sei also $S = \{ x^{-1}(L) \mid x \in L \}$ und $s_0 = \varepsilon^{-1}(L) = L$. Mit $\Delta = \{ (x^{-1}(L), a, (xa)^{-1}(L)) \mid x \in L, a \in \Sigma \text{ und } a \in x^{-1}(L) \}$ ist dann eine Automatendarstellung für L gegeben.

Dieser Automat ist auch deterministisch, denn aus $(p,a,q) \in \Delta$ und $(p,a,q') \in \Delta$ folgt $q = q'$. Man kann sich leicht überlegen, daß diese allgemeine Konstruktion in unserem Beispiel für die Sprache K zu dem angegebenen Automaten \mathbb{K} führt. Dazu ist lediglich zu beachten, daß $x^{-1}(K)$ vollständig durch das Paar $(u,v) \in \{C,D\}^* \times \{C,D\}^*$ charakterisiert ist, welches die beiden Queue-Inhalte nach der Aktionsfolge x beschreibt.

Bekanntlich ist eine Sprache $L \subset \Sigma^*$ genau dann regulär, wenn $\{x^{-1}(L) \mid x \in L\}$ eine endliche Menge ist; die obige Definition von Δ liefert dann gerade den Minimalautomaten von L /Eil74/.

Aus der Definition des Kooperationsproduktes formaler Sprachen läßt sich unmittelbar eine Automatenarstellung für $[F,G]_C$ ableiten, wenn man entsprechende Automatenarstellungen $(\Phi, S_F, \Delta_F, s_{0F})$, $(\Gamma, S_G, \Delta_G, s_{0G})$ und $(\Sigma, S_C, \Delta_C, s_{0C})$ für die Sprachen $F \subset \Phi^*$, $G \subset \Gamma^*$ und $C \subset \Sigma^*$ kennt. Beachtet man, daß das Kooperationsprodukt $[F,G]_C$ ein Shuffleprodukt aus F und G ist, das zusätzlich noch durch die Sprache C und die Homomorphismen ϕ und γ "kontrolliert" wird, dann kann man leicht verifizieren, daß die nachfolgenden Definitionen eine Automatenarstellung $(\Phi \cup \Gamma, S, \Delta, s_0)$ für $[F,G]_C$ liefern. Sei dazu $S = S_F \times S_C \times S_G$, $s_0 = (s_{0F}, s_{0C}, s_{0G})$ und $\Delta \subset S \times (\Phi \cup \Gamma) \times S$ mit $\Delta = \{((r,s,t), a, (r',s',t)) \in S \times (\Phi \setminus \phi^{-1}(\epsilon)) \times S \mid (r,a,r') \in \Delta_F \text{ und } (s,\phi(a),s') \in \Delta_C\} \cup \{((r,s,t), a, (r',s',t)) \in S \times (\Phi \cap \phi^{-1}(\epsilon)) \times S \mid (r,a,r') \in \Delta_F\} \cup \{((r,s,t), a, (r,s',t')) \in S \times (\Gamma \setminus \gamma^{-1}(\epsilon)) \times S \mid (t,a,t') \in \Delta_G \text{ und } (s,\gamma(a),s') \in \Delta_C\} \cup \{((r,s,t), a, (r,s,t')) \in S \times (\Gamma \cap \gamma^{-1}(\epsilon)) \times S \mid (t,a,t') \in \Delta_G\}$. Dieser Automat entspricht intuitiv, wie Abb. 18 zeigt, einem "Geflecht" von vier "Elementarautomaten", welche durch gemeinsame Komponenten ihrer Zustandsmengen miteinander "verklebt" sind. Dabei bewirkt beispielsweise eine "Aktion" von FR sowohl einen Zustandsübergang in der Komponente S_C als auch in S_F , wohingegen eine "Aktion" von FI nur einen Zustandsübergang in S_F hervorruft. Eine genaue Definition der Arbeitsweise eines solchen "Geflechts von Elementarautomaten" folgt weiter unten.

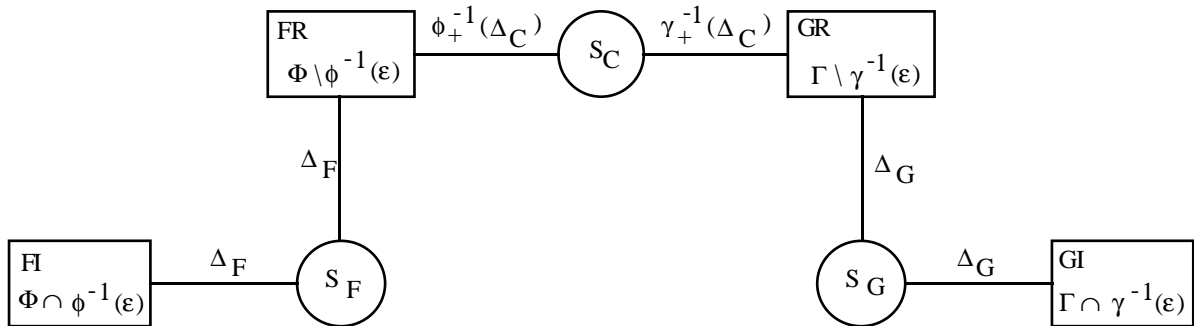


Abb. 18

Dabei seien die Abbildungen $\phi_+ : (S_C \times (\Phi \setminus \phi^{-1}(\epsilon)) \times S_C) \rightarrow (S_C \times \Sigma \times S_C)$ und $\gamma_+ : (S_C \times (\Gamma \setminus \gamma^{-1}(\epsilon)) \times S_C) \rightarrow (S_C \times \Sigma \times S_C)$ durch $\phi_+(u,v,w) = (u,\phi(v),w)$ und $\gamma_+(x,y,z) = (x,\gamma(y),z)$ definiert.

Bei dieser Konstruktion einer Automatenarstellung für $[F,G]_C$ wird natürlich nur der "Teil" von S und Δ benötigt, der vom Anfangszustand s_0 aus "erreichbar" ist. Eine schrittweise Konstruktion dieses "Teils" nennt

man *Erreichbarkeitsanalyse*. Die nachfolgende Tabelle zeigt für unser Beispiel diesen “relevanten Teil” von Δ .

$(1, (\varepsilon, \varepsilon), 1)$	- FCS	-	$(2, (C, \varepsilon), 1)$
$(2, (C, \varepsilon), 1)$	- FDS	-	$(4, (DC, \varepsilon), 1)$
$(2, (C, \varepsilon), 1)$	- GCR	-	$(2, (\varepsilon, \varepsilon), 2)$
$(4, (DC, \varepsilon), 1)$	- GCR	-	$(4, (D, \varepsilon), 2)$
$(2, (\varepsilon, \varepsilon), 2)$	- FDS	-	$(4, (D, \varepsilon), 2)$
$(2, (\varepsilon, \varepsilon), 2)$	- GCS	-	$(2, (\varepsilon, C), 3)$
$(2, (\varepsilon, \varepsilon), 2)$	- GDS	-	$(2, (\varepsilon, D), 5)$
$(4, (D, \varepsilon), 2)$	- GCS	-	$(4, (D, C), 3)$
$(4, (D, \varepsilon), 2)$	- GDR	-	$(4, (\varepsilon, \varepsilon), 4)$
$(4, (D, \varepsilon), 2)$	- GDS	-	$(4, (D, D), 5)$
$(2, (\varepsilon, C), 3)$	- FCR	-	$(3, (\varepsilon, \varepsilon), 3)$
$(2, (\varepsilon, C), 3)$	- FDS	-	$(4, (D, C), 3)$
$(2, (\varepsilon, C), 3)$	- GDS	-	$(2, (\varepsilon, CD), 5)$
$(2, (\varepsilon, D), 5)$	- FDS	-	$(4, (D, D), 5)$
$(2, (\varepsilon, D), 5)$	- FDR	-	$(5, (\varepsilon, \varepsilon), 5)$
$(4, (D, C), 3)$	- FCR	-	$(4, (D, \varepsilon), 3)$
$(4, (D, C), 3)$	- GDR	-	$(4, (\varepsilon, C), 4)$
$(4, (D, C), 3)$	- GDS	-	$(4, (D, CD), 5)$
$(4, (\varepsilon, \varepsilon), 4)$	- GDS	-	$(4, (\varepsilon, D), 1)$
$(4, (D, D), 5)$	- FDR	-	$(1, (D, \varepsilon), 5)$
$(4, (D, D), 5)$	- GDR	-	$(4, (\varepsilon, D), 1)$
$(3, (\varepsilon, \varepsilon), 3)$	- FDS	-	$(4, (D, \varepsilon), 3)$
$(3, (\varepsilon, \varepsilon), 3)$	- GDS	-	$(3, (\varepsilon, D), 5)$
$(2, (\varepsilon, CD), 5)$	- FCR	-	$(3, (\varepsilon, D), 5)$
$(2, (\varepsilon, CD), 5)$	- FDS	-	$(4, (D, CD), 5)$
$(5, (\varepsilon, \varepsilon), 5)$	- FDS	-	$(1, (D, \varepsilon), 5)$
$(4, (D, \varepsilon), 3)$	- GDR	-	$(4, (\varepsilon, \varepsilon), 4)$
$(4, (D, \varepsilon), 3)$	- GDS	-	$(4, (D, D), 5)$
$(4, (\varepsilon, C), 4)$	- FCR	-	$(4, (\varepsilon, \varepsilon), 4)$
$(4, (\varepsilon, C), 4)$	- GDS	-	$(4, (\varepsilon, CD), 1)$
$(4, (D, CD), 5)$	- FCR	-	$(4, (D, D), 5)$
$(4, (D, CD), 5)$	- GDR	-	$(4, (\varepsilon, CD), 1)$
$(4, (\varepsilon, D), 1)$	- FDR	-	$(1, (\varepsilon, \varepsilon), 1)$
$(1, (D, \varepsilon), 5)$	- FCS	-	$(2, (CD, \varepsilon), 5)$
$(1, (D, \varepsilon), 5)$	- GDR	-	$(1, (\varepsilon, \varepsilon), 1)$
$(3, (\varepsilon, D), 5)$	- FDS	-	$(4, (D, D), 5)$
$(3, (\varepsilon, D), 5)$	- FDR	-	$(5, (\varepsilon, \varepsilon), 5)$
$(4, (\varepsilon, CD), 1)$	- FCR	-	$(4, (\varepsilon, D), 1)$
$(2, (CD, \varepsilon), 5)$	- FDS	-	$(4, (DCD, \varepsilon), 5)$
$(2, (CD, \varepsilon), 5)$	- GDR	-	$(2, (C, \varepsilon), 1)$

(4 , (DCD , ε) , 5) - GDR - (4 , (DC , ε) , 1)

Die Tatsache, daß hier die Erreichbarkeitsanalyse nach endlich vielen Schritten beendet ist, zeigt, daß ein Kooperationsprodukt $[F,G]_C$ regulär sein kann, ohne daß alle “beteiligten” Sprachen regulär sind. In dem betrachteten Beispiel liegt das daran, daß sich, wie die Erreichbarkeitsanalyse zeigt, in den FIFO-Queues maximal 3 Nachrichten befinden. Für die Definition von K wurde hingegen keine obere Schranke für die Anzahl der Nachrichten in einer der beiden FIFO-Queues angenommen.

Sieht man in der obigen Automatendarstellung für ein Kooperationsprodukt formaler Sprachen von der speziellen Struktur dieses Geflechts von vier Elementarautomaten ab, dann ist folgende Definition motiviert:

Ein *asynchroner Produktautomat (APA)* besteht aus einer Familie von *Zustandsmengen* $(Z_s)_{s \in \mathbb{S}}$, einer Familie von *Elementarautomaten* $(\Phi_t, \Delta_t)_{t \in \mathbb{T}}$ und einer *Nachbarschaftsbeziehung* $N : \mathbb{T} \rightarrow \wp(\mathbb{S})$; $\wp(X)$ bezeichnet dabei die Potenzmenge einer Menge X . Für jeden Elementarautomaten (Φ_t, Δ_t) ist

- Φ_t sein *Alphabet* und
- $\Delta_t \subset \times_{s \in N(t)} (Z_s) \times \Phi_t \times \times_{s \in N(t)} (Z_s)$ seine *Zustandsübergangsrelation*.

Die *Zustände* eines APA sind Elemente von $\times_{s \in \mathbb{S}} (Z_s)$. Zur Vermeidung pathologischer Fälle wird generell $\mathbb{S} = \bigcup_{t \in \mathbb{T}} N(t)$ und $N(t) \neq \emptyset$ für alle $t \in \mathbb{T}$ vorausgesetzt. Jeder APA besitzt einen *Anfangszustand* $q_0 = (q_{0s})_{s \in \mathbb{S}} \in \times_{s \in \mathbb{S}} (Z_s)$. Ein APA \mathbb{A} ist also gegeben durch $\mathbb{A} = ((Z_s)_{s \in \mathbb{S}}, (\Phi_t, \Delta_t)_{t \in \mathbb{T}}, N, q_0)$.

Ein APA kann als eine Familie von Elementarautomaten betrachtet werden, deren jeweilige Zustandsmenge als Produktmenge strukturiert ist, wobei die einzelnen Elementarautomaten durch gemeinsame Komponenten ihrer Zustandsmengen “verklebt” sind.

Ein Elementarautomat (Φ_t, Δ_t) heißt *aktiviert in einem Zustand* $p = (p_s)_{s \in \mathbb{S}} \in \times_{s \in \mathbb{S}} (Z_s)$ *bezüglich einer Interpretation* $i \in \Phi_t$, falls es $(q_s)_{s \in N(t)} \in \times_{s \in N(t)} (Z_s)$ gibt mit $((p_s)_{s \in N(t)}, i, (q_s)_{s \in N(t)}) \in \Delta_t$. Ein aktivierter Elementarautomat (Φ_t, Δ_t) kann *schalten* und erzeugt den *Nachfolgezustand* $q = (q_r)_{r \in \mathbb{S}} \in \times_{s \in \mathbb{S}} (Z_s)$, falls $q_r = p_r$ für $r \in \mathbb{S} \setminus N(t)$ und $((p_s)_{s \in N(t)}, i, (q_s)_{s \in N(t)}) \in \Delta_t$. In diesem Fall bezeichnet $(p, (t, i), q)$ den entsprechenden *Schaltschritt*.

Zur formalen Interpretation der in Abb. 18 angegebenen allgemeinen Automatendarstellung für ein Kooperationsprodukt $[F,G]_C$ als APA seien die Indexmengen \mathbb{T} und \mathbb{S} durch $\mathbb{T} = \{FR, GR, FI, GI\}$ und $\mathbb{S} = \{F, C, G\}$ definiert. Die durch Verbindungslinien dargestellte Nachbarschaftsbeziehung wird dann mit der Abbildung $N : \{FR, GR, FI, GI\} \rightarrow \wp(\{F, C, G\})$ beschrieben, wobei $N(FR) = \{F, C\}$, $N(GR) = \{C, G\}$, $N(FI) = \{F\}$ und $N(GI) = \{G\}$. Die Familie $(Z_s)_{s \in \mathbb{S}}$ von Zustandsmengen kann direkt aus der Automatendarstellung für $[F,G]_C$ übernommen werden, es gilt also $Z_F = S_F$, $Z_C = S_C$ und $Z_G = S_G$. Der Anfangszustand $q_0 = (q_{0s})_{s \in \mathbb{S}}$ ist durch $q_{0F} = s_{0F}$, $q_{0C} = s_{0C}$ und $q_{0G} = s_{0G}$ gegeben. Für die Alphabete Φ_t gilt $\Phi_{FR} = \Phi \setminus \phi^{-1}(\epsilon)$, $\Phi_{GR} = \Gamma \setminus \gamma^{-1}(\epsilon)$, $\Phi_{FI} = \Phi \cap \phi^{-1}(\epsilon)$ und $\Phi_{GI} = \Gamma \cap \gamma^{-1}(\epsilon)$. Die

Zustandsübergangsrelationen Δ_t entsprechen den vier Teilmengen, aus denen die Zustandsübergangsrelation Δ der Automatendarstellung für $[F,G]_c$ zusammengesetzt sind. Im einzelnen gilt:

$$\Delta_{FR} = \{ ((r,s),a,(r',s')) \in (S_F \times S_C) \times (\Phi \setminus \Phi^{-1}(\epsilon)) \times (S_F \times S_C) \mid (r,a,r') \in \Delta_F \text{ und } (s,\phi(a),s') \in \Delta_C \}$$

$$\Delta_{FI} = \{ (r,a,r') \in S_F \times (\Phi \cap \Phi^{-1}(\epsilon)) \times S_F \mid (r,a,r') \in \Delta_F \}$$

$$\Delta_{GR} = \{ ((s,t),a,(s',t')) \in (S_C \times S_G) \times (\Gamma \setminus \Gamma^{-1}(\epsilon)) \times (S_C \times S_G) \mid (t,a,t') \in \Delta_G \text{ und } (s,\gamma(a),s') \in \Delta_C \}$$

$$\Delta_{GI} = \{ (t,a,t') \in S_G \times (\Gamma \cap \Gamma^{-1}(\epsilon)) \times S_G \mid (t,a,t') \in \Delta_G \}$$

Die APA sind ein sehr allgemeines und flexibles operationales Beschreibungskonzept für kooperierende Systeme, welches unterschiedliche syntaktische Ausprägungen erlaubt. Darunter fallen die verschiedenen Arten "einfacher" und "höherer" Petrinetze /OP95/, aber auch alle Arten kommunizierender Automaten wie z. B. SDL /SSR89/ oder Estelle /BD87/.

Abgesehen von den Interpretationen sind die APA ähnlich den asynchronen Zellularautomaten aus /Zie89/. Allerdings kann dort ein Schaltschritt nicht alle Zustandskomponenten aus der Nachbarschaft des entsprechenden Elementarautomaten "verändern", sondern jeweils nur eine, nämlich die, welche dem schaltenden Elementarautomaten fest zugeordnet ist. Eine deterministische Version der APA-Definition wurde in /Och94b,95/ gegeben.

Eine Folge $w = (q_1, (t_1, i_1), q_2) (q_2, (t_2, i_2), q_3) \dots (q_k, (t_k, i_k), q_{k+1}) (q_{k+1}, (t_{k+1}, i_{k+1}), q_{k+2}) \dots (q_n, (t_n, i_n), q_{n+1})$ von Schaltschritten eines APA mit $n \geq 1$, bei der also im Fall $n > 1$ die dritte Komponente des vorhergehenden Schaltschrittes gleich der ersten Komponente des nachfolgenden Schaltschrittes ist heißt *Schaltfolge*. Dabei ist q_1 der *Startzustand* von w und q_{n+1} der *Zielzustand* von w . Falls eine solche Schaltfolge w existiert, dann heißt q_{n+1} *erreichbar* von q_1 . Jeder Zustand sei per Definition aus sich selbst erreichbar.

Mit \mathbb{Q} bezeichnen wir die Menge aller Zustände $q \in \times_{s \in \mathbb{S}} (Z_s)$, die vom Anfangszustand q_0 aus erreichbar sind und mit Ψ die Menge aller Schaltschritte, deren erste Komponente in \mathbb{Q} liegt. Die Menge $L \subset \Psi^*$ aller Schaltfolgen mit dem Startzustand q_0 einschließlich der leeren Folge ϵ heißt *Schaltfolgensprache* des entsprechenden APA. Sie ist eine präfixstabile Sprache.

Die Menge Ψ kann als Kantenmenge eines gerichteten Graphen interpretiert werden, dessen Knotenmenge \mathbb{Q} ist und dessen Kanten mit Paaren (t,i) beschriftet sind, wobei $t \in \mathbb{T}$ und $i \in \Phi_t$. Dieser Graph heißt *Erreichbarkeitsgraph* des betrachteten APA. Damit entsprechen den Schaltfolgen die Pfade im Erreichbarkeitsgraph, und die Schaltfolgensprache L ist eine reguläre Sprache, falls der Erreichbarkeitsgraph endlich ist (lokale Sprache /Eil74/). Die Schaltfolgensprache und der Erreichbarkeitsgraph entsprechen sich gegenseitig in eindeutiger Weise und sind beide eine vollständige Beschreibung des dynamischen Verhaltens eines APA.

Man kann sich jetzt leicht überlegen, daß der Erreichbarkeitsgraph des APA, welcher die Automatendarstellung eines Kooperationsproduktes $[F,G]_c$ beschreibt, isomorph zu dem Teil der Zustandsübergangsrelation Δ dieser Automatendarstellung ist, der vom Anfangszustand aus erreichbar ist. Der einzige Unterschied liegt darin, daß

die Kanten des Erreichbarkeitsgraphen mit Paaren (t,i) und nicht nur mit i beschriftet sind. Wegen der Disjunktheit von Φ und Γ gibt es aber in diesem Fall zu jedem solchen i nur genau ein $t \in \mathbb{T}$ mit $i \in \Phi_t$. Rein auf Sprachebene betrachtet ist nun jedes Kooperationsprodukt $[F,G]_C$ als homomorphes Bild einer Schaltfolgensprache dargestellt. Dieser Homomorphismus bildet jeden Schaltschritt $(p,(t,i),q)$ auf das entsprechende $i \in \Phi \cup \Gamma$ ab.

Wie schon oben bemerkt, dienen asynchrone Produktautomaten der Spezifikation kooperierender Systeme. Die Schaltfolgensprachen beschreiben dann das globale dynamische Verhalten der spezifizierten Systeme. Zur Verifikation solcher Spezifikationen müssen bestimmte homomorphe Bilder der entsprechenden Schaltfolgensprachen betrachtet werden. Genau wie bei den Kooperationsprodukten tritt auch bei den Schaltfolgensprachen bzw. Erreichbarkeitsgraphen das Problem der Zustandsexplosion auf. Es ist daher sinnvoll, Schaltfolgensprachen, oder wenigstens hinreichend feine homomorphe Bilder davon, als Kooperationsprodukte darzustellen, um die im vorhergehenden Kapitel entwickelten Methoden zur Komplexitätsreduktion auch hier nutzbar zu machen.

Wesentliche Voraussetzung dafür ist eine Strukturierung der APA in sogenannte *Module*. Dies geschieht durch eine Partitionierung der Menge \mathbb{T} . Die bei der Definition der APA erwähnte Verklebung von Elementarautomaten kann auch auf beliebige APA erweitert werden. Seien dazu $\mathbb{A}_k = ((Z_{\mathbb{S}_k})_{\mathbb{S}_k}, (\Phi_t, \Delta_t)_{t \in \mathbb{T}_k}, N_k, q_{k0})$ für $k \in \{1,2\}$ zwei APA mit $\mathbb{T}_1 \cap \mathbb{T}_2 = \emptyset$ sowie $Z_{1_s} = Z_{2_s}$ und $q_{1_{0s}} = q_{2_{0s}}$ für alle $s \in \mathbb{S}_1 \cap \mathbb{S}_2$. Die *direkte Summe* $\mathbb{A}_1 \oplus \mathbb{A}_2$ der beiden APA ist dann definiert durch $\mathbb{A}_1 \oplus \mathbb{A}_2 = ((Z_{\mathbb{S}})_{\mathbb{S}}, (\Phi_t, \Delta_t)_{t \in \mathbb{T}}, N, q_0)$ mit $\mathbb{S} = \mathbb{S}_1 \cup \mathbb{S}_2$, $\mathbb{T} = \mathbb{T}_1 \cup \mathbb{T}_2$, $Z_s = Z_{k_s}$ und $q_{0s} = q_{k_{0s}}$ für $s \in \mathbb{S}_k$, sowie $(\Phi_t, \Delta_t) = (\Phi_{k_t}, \Delta_{k_t})$ für $t \in \mathbb{T}_k$ und $N(t) = N_k(t)$, wobei wegen $\mathbb{T}_1 \cap \mathbb{T}_2 = \emptyset$ das jeweilige $k \in \{1,2\}$ eindeutig bestimmt ist. Man sagt auch, daß $\mathbb{A}_1 \oplus \mathbb{A}_2$ aus den Modulen \mathbb{A}_1 und \mathbb{A}_2 durch *Verkleben der gemeinsamen Zustandskomponenten* aus $\mathbb{S}_1 \cap \mathbb{S}_2$ entsteht. Mit dieser Definition ist jeder APA die direkte Summe seiner Elementarautomaten. Wie oben bemerkt, kann Abb. 18 als APA interpretiert werden. Mit einer entsprechenden Interpretation ist er dann die direkte Summe der in Abb. 19 und 20 dargestellten APA. Diese sind dabei an der gemeinsamen Zustandskomponente C verklebt.

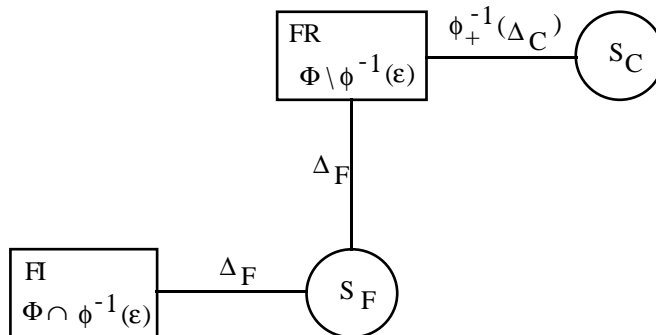


Abb. 19

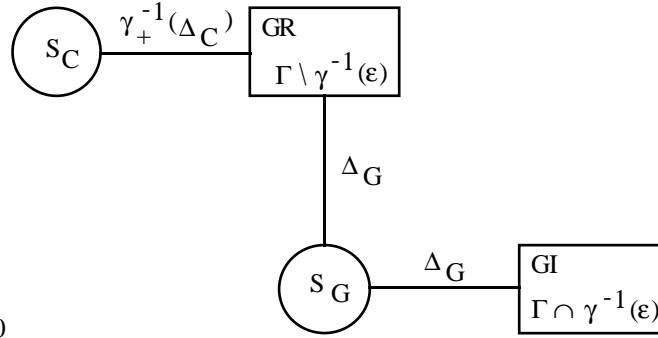


Abb. 20

Zur Beschreibung des dynamische Verhaltens eines Moduls in einem APA sind zwei spezielle Homomorphismen von Nutzen. Zu ihrer Definition sei für einen APA $((Z_s)_{s \in \mathbb{S}}, (\Phi_t, \Delta_t)_{t \in \mathbb{T}}, N, q_0)$ und eine Teilmenge $A \subset \mathbb{T}$ das Komplement $\mathbb{T} \setminus A$ mit A' bezeichnet, und $RDA = N(A) \cap N(A') \subset \mathbb{S}$. Für eine Teilmenge $Y \subset \mathbb{S}$ und einen Zustand $q = (q_s)_{s \in \mathbb{S}} \in \times_{s \in \mathbb{S}} (Z_s)$ sei $q|Y \in \times_{s \in Y} (Z_s)$ definiert durch $q|Y = (q_s)_{s \in Y}$ und heißt der auf Y eingeschränkte Zustand q . Es sei $\Psi_{RA} = \{ (p|RDA, q|RDA) \mid \text{es existiert } (p, (t, i), q) \in \Psi \text{ mit } t \in A \text{ und } N(t) \cap RDA \neq \emptyset \}$ und $\Psi_{MA} = \{ (p|N(A), (t, i), q|N(A)) \mid (p, (t, i), q) \in \Psi, t \in A \text{ und } N(t) \cap RDA \neq \emptyset \} \cup \{ (p|N(A) \setminus RDA, (t, i), q|N(A) \setminus RDA) \mid (p, (t, i), q) \in \Psi, t \in A \text{ und } N(t) \cap RDA = \emptyset \}$.

Durch $\mu_A((p, (t, i), q)) = (p|N(A), (t, i), q|N(A))$ falls $t \in A$ und $N(t) \cap RDA \neq \emptyset$, $\mu_A((p, (t, i), q)) = (p|N(A) \setminus RDA, (t, i), q|N(A) \setminus RDA)$ falls $t \in A$ und $N(t) \cap RDA = \emptyset$ und $\mu_{A'}((p, (t, i), q)) = \varepsilon$ falls $t \in A'$ ist ein Sprachhomomorphismus $\mu_A : \Psi^* \rightarrow \Psi_{MA}^*$ definiert, welcher *Modulhomomorphismus* auf Ψ^* bezüglich A genannt wird; durch $\rho_A((p, (t, i), q)) = (p|RDA, q|RDA)$ falls $t \in A$ und $N(t) \cap RDA \neq \emptyset$ sowie $\rho_A((p, (t, i), q)) = \varepsilon$ falls $t \in A'$ oder $N(t) \cap RDA = \emptyset$ ist ein Sprachhomomorphismus $\rho_A : \Psi^* \rightarrow \Psi_{RA}^*$ definiert, welcher *Randhomomorphismus* auf Ψ^* bezüglich A genannt wird. Die homomorphen Bilder $\rho_A(L)$ bzw. $\mu_A(L)$ der Schaltfolgensprache $L \subset \Psi^*$ beschreiben jetzt das “Randverhalten” bzw. das “vollständige Verhalten” des Moduls A im Zusammenspiel mit dem Modul A' . Der Modulhomomorphismus ist feiner als der Randhomomorphismus, d.h. es existiert ein Homomorphismus $\kappa_A : \Psi_{MA}^* \rightarrow \Psi_{RA}^*$ mit $\rho_A = \kappa_A \circ \mu_A$.

Entscheidend für den Rest dieses Kapitels ist die “Lokalität” von Schaltschritten, die bedeutet, daß sich die Zustandsveränderung nur in der Nachbarschaft des entsprechenden Elementarautomaten abspielt. Ist $(p, (t, i), q)$ ein Schaltschritt eines APA, dann gilt nach Definition $q|\mathbb{S} \setminus N(t) = p|\mathbb{S} \setminus N(t)$. Ist p' ein weiterer Zustand mit $p'|N(t) = p|N(t)$, dann ist auch $(p', (t, i), q')$ ein Schaltschritt, wobei q' durch $q'|N(t) = q|N(t)$ und $q'|\mathbb{S} \setminus N(t) = p'|\mathbb{S} \setminus N(t)$ gegeben ist. Eine entsprechende Aussage gilt auch noch dann, wenn der betrachtete Elementarautomat einschließlich seiner benachbarten Zustandskomponenten in verschiedene APA “eingebettet” ist.

Mögen die beiden direkten Summen $A \oplus B$ und $A \oplus C$ die gleiche Menge RDA gemeinsamer Zustandskomponenten besitzen. Die Indexmenge der Elementarautomaten von A sei A . $L \subset \Psi^*$ bzw. $L' \subset \Psi'^*$ sei die Schaltfolgensprache von $A \oplus B$ bzw. $A \oplus C$ und μ_A bzw. μ'_A der Modulhomomorphismus auf Ψ^* bzw. Ψ'^* bezüglich A . Aus den obigen Überlegungen zur “Lokalität” von

Schaltsschritten folgt dann (vollständige Induktion), daß für $x \in L$ und $x' \in L'$ mit $\mu_A(x) = \mu'_A(x')$ die Gleichung $Z(x)|N(A)\backslash RDA = Z(x')|N(A)\backslash RDA$ gilt. Dabei bezeichnet $Z(x)$ den Zielzustand einer Schaltfolge x ($Z(\varepsilon) = \text{Anfangszustand des APA}$).

Ein APA $A = ((Z_s)_{s \in \mathbb{S}}, (\Phi_t, \Delta_t)_{t \in \mathbb{T}}, N, q_0)$ kann sich im Zusammenspiel mit anderen APA sehr unterschiedlich verhalten. Er zeigt aber mindestens das Verhalten, welches durch seine eigene Schaltfolgensprache beschrieben ist. Betrachtet man nur solche Partner von A , welche eine fest ausgewählte Menge $RDA \subset \mathbb{S}$ von gemeinsamen Zustandskomponenten mit A haben, dann existiert ein "maximales" Verhalten, welches A im Zusammenspiel mit solchen Partnern zeigen kann. Dieses "maximale" Verhalten tritt dann auf, wenn A mit einem APA verklebt wird, der in jedem Schaltschritt alle möglichen Zustandsveränderungen auf $\times_{s \in RDA} (Z_s)$ durchführen kann. Ein spezieller APA $A_R = ((ZR_s)_{s \in \mathbb{S}_R}, (\Phi_{R_t}, \Delta_{R_t})_{t \in \mathbb{T}_R}, NR, qR_0)$, der dieses leistet, ist durch $\mathbb{T}_R = \{\#\}$ mit $\# \notin \mathbb{T}$, $\mathbb{S}_R = RDA$, $NR(\#) = RDA$, $(ZR_s)_{s \in RDA} = (Z_s)_{s \in RDA}$, $qR_0 = (q_{0s})_{s \in RDA}$, $\Phi_{R\#} = \{\#\}$ und $\Delta_{R\#} = \times_{s \in RDA} (Z_s) \times \{\#\} \times \times_{s \in RDA} (Z_s)$ gegeben.

Zur Präzisierung dieser Aussage sei $A1 \oplus A2$ die direkte Summe zweier APA $Ak = ((Zk_s)_{s \in \mathbb{S}_k}, (\Phi_{k_t}, \Delta_{k_t})_{t \in \mathbb{T}_k}, Nk, qk_0)$ ($k \in \{1,2\}$) mit $\mathbb{S}_1 \cap \mathbb{S}_2 = RDA$. Die Schaltfolgensprache von $A1 \oplus A2$ sei $L \subset \Psi^*$ und $\mu_{\mathbb{T}_1}$ bzw. $\mu_{\mathbb{T}_2}$ sei der Modulhomomorphismus auf Ψ^* bezüglich \mathbb{T}_1 bzw. \mathbb{T}_2 . Die APA $A1R$ und $A2R$ seien wie oben definiert. $L1R \subset \Psi1R^*$ bzw. $L2R \subset \Psi2R^*$ sei die Schaltfolgensprache von $A1 \oplus A1R$ bzw. $A2R \oplus A2$, und $\mu1_{\mathbb{T}_1}$ bzw. $\mu2_{\mathbb{T}_2}$ sei der Modulhomomorphismus auf $\Psi1R^*$ bzw. $\Psi2R^*$ bezüglich \mathbb{T}_1 bzw. \mathbb{T}_2 . $\rho1_{\mathbb{T}_1}$ und $\rho2_{\mathbb{T}_2}$ seien die entsprechenden Randhomomorphismen. Wie oben bemerkt, gibt es dann Homomorphismen $\kappa1_{\mathbb{T}_1}$ und $\kappa2_{\mathbb{T}_2}$ mit $\rho1_{\mathbb{T}_1} = \kappa1_{\mathbb{T}_1} \circ \mu1_{\mathbb{T}_1}$ und $\rho2_{\mathbb{T}_2} = \kappa2_{\mathbb{T}_2} \circ \mu2_{\mathbb{T}_2}$. Zur Formulierung des nachfolgenden Satzes wird noch die Kooperationsform $r = (\mu1_{\mathbb{T}_1}(\Psi1R), \mu2_{\mathbb{T}_2}(\Psi2R), \Sigma, \kappa1_{\mathbb{T}_1}, \kappa2_{\mathbb{T}_2}, R)$ benötigt, wobei $\Sigma = \times_{s \in RDA} (Z_s) \times \times_{s \in RDA} (Z_s)$ und $R = \{ x \in \Sigma^* \mid \text{für } 1 \leq i \leq |x| \text{ existieren } p_i \in \times_{s \in RDA} (Z_s) \text{ mit } x = a_1 \dots a_{|x|} \text{ s}_i \in \mathbb{S}, a_i = (p_{i-1}, p_i) \text{ und } p_0 = (q_{0s})_{s \in RDA} \}$. Mit diesen Bezeichnungen gilt

Satz 3: $[\mu_{\mathbb{T}_1}, \mu_{\mathbb{T}_2}](L) = [\mu1_{\mathbb{T}_1}(L1R), \mu2_{\mathbb{T}_2}(L2R)]_r$

Zusätzlich gilt noch $\mu_{\mathbb{T}_1}(\Psi) \subset \mu1_{\mathbb{T}_1}(\Psi1R)$ sowie $\mu_{\mathbb{T}_2}(\Psi) \subset \mu2_{\mathbb{T}_2}(\Psi2R)$.

Beweis:

Zum Beweis der Gleichung genügt es, die beiden folgenden Aussagen I und II zu zeigen. I: Zu jedem $x \in L$ existieren $y_1 \in L1R$ und $y_2 \in L2R$ mit $[\mu_{\mathbb{T}_1}, \mu_{\mathbb{T}_2}](x) \in [\mu1_{\mathbb{T}_1}(\{y_1\}), \mu2_{\mathbb{T}_2}(\{y_2\})]_r$. II: Zu jedem $y \in [\mu1_{\mathbb{T}_1}(L1R), \mu2_{\mathbb{T}_2}(L2R)]_r$ existiert ein $x \in L$ mit $y = [\mu_{\mathbb{T}_1}, \mu_{\mathbb{T}_2}](x)$.

Beweis von I durch vollständige Induktion über die Länge von x :

Ist $x = \varepsilon$, dann erfüllen $y_1 = y_2 = \varepsilon$ die Behauptung. Sei jetzt $x = x'(p, (t, i), q) \in L$ mit $(p, (t, i), q) \in \Psi$.

Nach Induktionsvoraussetzung gibt es $y_1' \in L1R$ und $y_2' \in L2R$ mit $[\mu_{T1}, \mu_{T2}](x') \in [\mu_{T1}(\{y_1'\}), \mu_{T2}(\{y_2'\})]_T$. Daraus folgt $\mu_{T1}(x') = \mu_{T1}(y_1')$ und $\mu_{T2}(x') = \mu_{T2}(y_2')$, also auch $Z(x')|S1 \setminus RDA = Z(y_1')|S1 \setminus RDA$ und $Z(x')|S2 \setminus RDA = Z(y_2')|S2 \setminus RDA$. O. B. d. A. sei $t \in T1$, also $\mu_{T2}(x) = \mu_{T2}(x') = \mu_{T2}(y_2')$. Wegen $(Z(y_1'), (\#, \#), Z(x')|S1) \in \Psi1R$ ist $y_1 = y_1'(Z(y_1'), (\#, \#), Z(x')|S1)(p|S1, (t, i), q|S1) \in L1R$ mit $\mu_{T1}(x) = \mu_{T1}(y_1)$. Wegen $[\kappa_{T1}, \kappa_{T2}]([\mu_{T1}, \mu_{T2}](x') \mu_{T1}((Z(y_1'), (\#, \#), Z(x')|S1)(p|S1, (t, i), q|S1))) = [\kappa_{T1}, \kappa_{T2}]([\mu_{T1}, \mu_{T2}](x') \mu_{T1}((p, (t, i), q))) = [\rho_{T1}, \rho_{T2}](x) \in R$ folgt daraus $[\mu_{T1}, \mu_{T2}](x) \in [\mu_{T1}(\{y_1'\}), \mu_{T2}(\{y_2'\})]_T$.

Aus diesem Induktionsbeweis folgen auch noch die beiden Inklusionen bezüglich der betrachteten Bildalphabeten.

Beweis von II durch vollständige Induktion über die Länge von y :

Ist $y = \varepsilon$, dann erfüllt $x = \varepsilon$ die Behauptung. Sei jetzt $y = y'a \in [\mu_{T1}(L1R), \mu_{T2}(L2R)]_T$ mit $a \in \mu_{T1}(\Psi1R) \cup \mu_{T2}(\Psi2R)$; o. B. d. A. $a \in \mu_{T1}(\Psi1R)$. Dann existieren $y_1'(p', (t, i), q') \in L1R$ und $y_2' \in L2R$ mit $(p', (t, i), q') \in \Psi1R$, $\mu_{T1}((p', (t, i), q')) = a$ und $y \in [\mu_{T1}(\{y_1'(p', (t, i), q')\}), \mu_{T2}(\{y_2'\})]_T$. Damit ist $y' \in [\mu_{T1}(\{y_1'\}), \mu_{T2}(\{y_2'\})]_T$, und nach Induktionsvoraussetzung existiert ein $x' \in L$ mit $y' = [\mu_{T1}, \mu_{T2}](x')$. Insbesondere ist dann $\mu_{T1}(x') = \mu_{T1}(y_1')$ und damit $Z(x')|S1 \setminus RDA = Z(y_1')|S1 \setminus RDA$. Es sind jetzt die beiden Fälle $N1(t) \cap RDA = \emptyset$ und $N1(t) \cap RDA \neq \emptyset$ zu betrachten. Im Fall $N1(t) \cap RDA = \emptyset$ läßt sich jetzt $(p', (t, i), q') \in \Psi1R$ zu einem Schaltschritt $(p, (t, i), q) \in \Psi$ "ausdehnen", sodaß $\mu_{T1}((p', (t, i), q')) = \mu_{T1}((p, (t, i), q))$ und $x'(p, (t, i), q) \in L$. Daraus folgt dann $y = y'a = [\mu_{T1}, \mu_{T2}](x'(p, (t, i), q))$. Im Fall $N1(t) \cap RDA \neq \emptyset$ ist $\rho_{T1}((p', (t, i), q')) = \kappa_{T1}(\mu_{T1}((p', (t, i), q'))) \neq \varepsilon$. Wegen $[\kappa_{T1}, \kappa_{T2}](y) = [\kappa_{T1}, \kappa_{T2}](y')(p'|RDA, q'|RDA) \in R$ und $[\kappa_{T1}, \kappa_{T2}](y') = [\kappa_{T1}, \kappa_{T2}]([\mu_{T1}, \mu_{T2}](x')) = [\rho_{T1}, \rho_{T2}](x')$ folgt dann $p'|RDA = Z(x')|RDA$. Damit läßt sich auch in diesem Fall $(p', (t, i), q') \in \Psi1R$ zu einem Schaltschritt $(p, (t, i), q) \in \Psi$ "ausdehnen", sodaß $\mu_{T1}((p', (t, i), q')) = \mu_{T1}((p, (t, i), q))$ und $x'(p, (t, i), q) \in L$.

Mit Satz 3 wird $[\mu_{T1}, \mu_{T2}](L)$ durch ein Kooperationsprodukt dargestellt. Dieses homomorphe Bild beschreibt ebenso wie L selbst vollständig das dynamische Verhalten von $A1 \oplus A2$, da es alle "wesentliche" Information über die einzelnen Schaltschritte enthält. Zur Präzisierung dieser Aussage betrachten wir die Schaltfolgensprache $L \subset \Psi^*$ eines beliebigen APA. Sei $v : \Psi^* \rightarrow \Psi'^*$ der durch $v((p, (t, i), q)) = (p|N(t), (t, i), q|N(t))$ definierte Homomorphismus, wobei $\Psi' = \{(p|N(t), (t, i), q|N(t)) \mid (p, (t, i), q) \in \Psi\}$. v ist injektiv auf L , denn ausgehend vom Anfangszustand des APA können die Urbilder von Elementen aus $v(L)$ sequentiell berechnet werden. Dann ist mit den Voraussetzungen von Satz 3 auch der Homomorphismus $[\mu_{T1}, \mu_{T2}]$ injektiv auf L , da er feiner als v ist.

3 Austauschätze

Die im vorhergehenden Kapitel angegebene allgemeine Automatendarstellung eines Kooperationsproduktes als APA liefert für das dort betrachtete Verbindungsauf- und -abbauprotokoll einen APA \mathbb{A} , der aus den beiden Elementarautomaten (Φ_{FR}, Δ_{FR}) und (Φ_{GR}, Δ_{GR}) sowie aus den drei Zustandsmengen S_F , S_G und S_C besteht. Seine Nachbarschaftsbeziehung N ist durch $N(FR) = \{F, C\}$ und $N(GR) = \{G, C\}$ gegeben. Damit besitzt \mathbb{A} eine Darstellung $\mathbb{A} = \mathbb{F} \oplus \mathbb{G}$, wobei \mathbb{F} und \mathbb{G} jeweils nur aus einem Elementarautomaten bestehen. Die einzige gemeinsame Zustandskomponente ist C . Betrachtet man gemäß der im vorhergehenden Kapitel gegebenen Definition die APA $\mathbb{F}R$ und $\mathbb{G}R$, dann besitzen die Schaltfolgensprachen von $\mathbb{F} \oplus \mathbb{F}R$ und $\mathbb{G} \oplus \mathbb{G}R$ sowie die davon in Satz 3 benötigten homomorphen Bilder unendliche Alphabete, wohingegen die Schaltfolgensprache von $\mathbb{F} \oplus \mathbb{G}$ sogar regulär ist. Das bedeutet, daß die am Ende des vorherigen Kapitels angegebene Beziehung zwischen Kooperationsprodukten und direkten Summen von APA prinzipieller Natur ist, aber so nicht unbedingt "praktisch anwendbar" ist. Um sie dennoch auch in den oben geschilderten Situationen nutzen zu können, wird noch ein gewisser "Austauschsatz" (Satz 5) benötigt.

Aus Satz 1 und den Überlegungen zu Beginn des vorhergehenden Kapitels folgt unmittelbar

Korollar 1:
$$[f, g]([F, G]_C) = [[f, E_{\Gamma}](F, G)_C], [E_{\Phi}, g]([F, G]_C)]_C$$

Dabei sind die Bezeichnungen wie in Satz 1 gewählt. Mit Hilfe dieses Satzes kann $[I_{\Phi}, E_{\Gamma}](F, G)_C$ ebenso wie $[E_{\Phi}, I_{\Gamma}](F, G)_C$ aus "einfacheren" Sprachen als $[F, G]_C$ bestimmt werden. Dazu setzen wir zusätzlich $\Phi \cap \Gamma' = \emptyset$ voraus. Dann gilt nämlich $[I_{\Phi}, E_{\Gamma}] = [I_{\Phi}, E_{\Gamma'}] \circ [I_{\Phi}, g]$ also $[I_{\Phi}, E_{\Gamma}](F, G)_C = [I_{\Phi}, E_{\Gamma'}](I_{\Phi}, g]([F, G]_C)) = [I_{\Phi}, E_{\Gamma'}](F, g(G))_d$, wobei $d = (\Phi, \Gamma', \Sigma, \phi, \gamma', C)$. Die "einfachste" dieser Darstellungen erhält man unter der Voraussetzung $\Phi \cap \Sigma = \emptyset$ mit $g = \gamma$ und $\Gamma' = \Sigma$, also $[I_{\Phi}, E_{\Gamma}](F, G)_C = [I_{\Phi}, E_{\Sigma}](F, \gamma(G))_c$, wobei $e = (\Phi, \Sigma, \Sigma, \phi, I_{\Sigma}, C)$. Mit $c^\circ = (\Phi, \Gamma^\circ, \Sigma, \phi, \gamma^\circ, C)$ folgt aus dieser Darstellung unmittelbar

Korollar 2:
$$[I_{\Phi}, E_{\Gamma}](F, G)_C = [I_{\Phi}, E_{\Gamma^\circ}](F, G^\circ)_{c^\circ} \text{ falls } \gamma(G) = \gamma^\circ(G^\circ)$$

Für den nächsten Satz werden ebenfalls die Voraussetzungen von Satz 1 benötigt. Auf die Bedingung $g \prec \gamma$ kann allerdings verzichtet werden. Zusätzlich wird noch $\Phi \cap \Gamma' = \emptyset = \Phi' \cap \Gamma$ vorausgesetzt. Mit $c^\wedge = (\Phi', \Gamma, \Sigma, \phi', \gamma, C)$ gilt dann

Satz 4:
$$[I_{\Phi}, g]([F, G]_C) = [f, I_{\Gamma'}]^{-1}([I_{\Phi'}, g]([f(F), G]_{c^\wedge})) \cap [F, \Gamma'^*]_\emptyset$$

Beweis:

Nach Satz 1 gilt $[f, I_{\Gamma'}]^{-1}([I_{\Phi'}, g]([f(F), G]_{c^\wedge})) = [f, I_{\Gamma'}]^{-1}([I_{\Phi'}, g]([f, I_{\Gamma'}](F, G)_C))) = [f, I_{\Gamma'}]^{-1}([f, I_{\Gamma'}]([I_{\Phi}, g]([F, G]_C))) \supset [I_{\Phi}, g]([F, G]_C)$. Wegen $[F, \Gamma'^*]_\emptyset \supset [I_{\Phi}, g]([F, G]_C)$ gilt dann $[I_{\Phi}, g]([F, G]_C) \subset [f, I_{\Gamma'}]^{-1}([I_{\Phi'}, g]([f(F), G]_{c^\wedge})) \cap [F, \Gamma'^*]_\emptyset$. Sei jetzt $x \in [f, I_{\Gamma'}]^{-1}([I_{\Phi'}, g]([f(F), G]_{c^\wedge})) \cap [F, \Gamma'^*]_\emptyset$, dann existiert ein $y \in [F, G]_C$ mit $[f, I_{\Gamma'}](x) = [I_{\Phi'}, g]([f, I_{\Gamma'}](y)) = [f, g](y)$. Dann gibt es Zerlegungen $x = u_1 v_1 u_2 v_2 \dots u_n v_n$ und $y = r_1 s_1 r_2 s_2 \dots r_n s_n$ mit $n \geq 1$ und $u_i \in \Phi^*$, $v_i \in \Gamma'^*$, $r_i \in \Phi^*$, $s_i \in \Gamma^*$, $f(u_i) = f(r_i)$ sowie $v_i = g(s_i)$ für $1 \leq i \leq n$. Sei $z = u_1 s_1 u_2 s_2 \dots u_n s_n$, dann ist $z \in [F, G]_\emptyset$ und $[f, I_{\Gamma'}](z) = [f, I_{\Gamma'}](y)$ sowie $[I_{\Phi}, g](z) = x$. Wegen $[f, I_{\Gamma'}] \prec [\phi, \gamma]$ folgt daraus $z \in [F, G]_C$. Damit gilt

$x \in [I_{\Phi, g}]([F, G]_{\mathcal{C}})$, was den Beweis abschließt.

Wir werden diesen Satz insbesondere für den Fall $g = E_{\Gamma}$ benutzen. Mit $d' = (\Phi', \Gamma, \Sigma, \phi', \gamma, C)$ und $d^{\circ} = (\Phi', \Gamma^{\circ}, \Sigma, \phi', \gamma^{\circ}, C)$ folgt dann unmittelbar

Korollar 3: $[I_{\Phi, E_{\Gamma}}]([F, G]_{\mathcal{C}}) = [I_{\Phi, E_{\Gamma^{\circ}}}]([F, G^{\circ}]_{\mathcal{C}^{\circ}})$ falls $[I_{\Phi', E_{\Gamma}}]([f(F), G]_{d'}) = [I_{\Phi', E_{\Gamma^{\circ}}}]([f(F), G^{\circ}]_{d^{\circ}})$

Die Sätze 1 und 4 liefern zusammen mit ihren Korollaren Beziehungen zwischen den Bildern von $[F, G]_{\mathcal{C}}$ bezüglich der Homomorphismen $[f, I_{\Gamma}]$, $[I_{\Phi, g}]$ und $[f, g]$. Der folgende Satz 5 ermöglicht es, diese Bilder aus einfacheren Sprachen als $[F, G]_{\mathcal{C}}$ zu bestimmen.

Lemma 1:

Sei $F1 \subset \Phi^*$, $F2 \subset \Phi^*$, $G1 \subset \Gamma^*$, $G2 \subset \Gamma^*$ und $c = (\Phi, \Gamma, \Sigma, \phi, \gamma, C)$. Aus $[F1, G1]_{\mathcal{C}} = [F2, G2]_{\mathcal{C}}$ folgt $[F1, G2]_{\mathcal{C}} = [F2, G1]_{\mathcal{C}} = [F1, G1]_{\mathcal{C}} = [F2, G2]_{\mathcal{C}}$.

Beweis:

Sei $x \in [F1, G1]_{\mathcal{C}} = [F2, G2]_{\mathcal{C}}$. Wegen $[I_{\Phi, E_{\Gamma}}](x) \in F1$, $[I_{\Phi, E_{\Gamma}}](x) \in F2$, $[E_{\Phi, I_{\Gamma}}](x) \in G1$, $[E_{\Phi, I_{\Gamma}}](x) \in G2$ und $[\phi, \gamma](x) \in C$ folgt $x \in [F1, G2]_{\mathcal{C}}$ und $x \in [F2, G1]_{\mathcal{C}}$, also $[F1, G1]_{\mathcal{C}} = [F2, G2]_{\mathcal{C}} \subset [F1, G2]_{\mathcal{C}}$ und $[F1, G1]_{\mathcal{C}} = [F2, G2]_{\mathcal{C}} \subset [F2, G1]_{\mathcal{C}}$. Wir zeigen jetzt durch Induktion über die Länge von x , daß $x \in [F1, G1]_{\mathcal{C}} = [F2, G2]_{\mathcal{C}}$ aus $x \in [F1, G2]_{\mathcal{C}}$ folgt. Für $x = \varepsilon$ stimmt die Behauptung. Sei $x = ya$ mit $a \in \Phi \cup \Gamma$, o.B.d.A. $a \in \Phi$. Nach Induktionsvoraussetzung ist dann $y \in [F1, G1]_{\mathcal{C}}$. Wegen $[I_{\Phi, E_{\Gamma}}](ya) \in F1$, $[\phi, \gamma](ya) \in C$ und $[E_{\Phi, I_{\Gamma}}](ya) = [E_{\Phi, I_{\Gamma}}](y) \in G1$ ist dann auch $x = ya \in [F1, G1]_{\mathcal{C}} = [F2, G2]_{\mathcal{C}}$. Damit ist $[F1, G2]_{\mathcal{C}} \subset [F1, G1]_{\mathcal{C}} = [F2, G2]_{\mathcal{C}}$ gezeigt. Der Beweis für $[F2, G1]_{\mathcal{C}} \subset [F1, G1]_{\mathcal{C}} = [F2, G2]_{\mathcal{C}}$ verläuft entsprechend.

Satz 5:

Unter den Voraussetzungen von Lemma 1 gilt $[I_{\Phi, E_{\Gamma}}]([F1, G2]_{\mathcal{C}}) = [I_{\Phi, E_{\Gamma}}]([F1, G1]_{\mathcal{C}})$ und $[E_{\Phi, I_{\Gamma}}]([F1, G2]_{\mathcal{C}}) = [E_{\Phi, I_{\Gamma}}]([F2, G2]_{\mathcal{C}})$ falls $[\phi, E_{\Gamma}]([F1, G1]_{\mathcal{C}}) = [\phi, E_{\Gamma}]([F2, G2]_{\mathcal{C}})$ und $[E_{\Phi, \gamma}]([F1, G1]_{\mathcal{C}}) = [E_{\Phi, \gamma}]([F2, G2]_{\mathcal{C}})$.

Beweis:

Damit alle benötigten Disjunktheitsbedingungen erfüllt sind wählen wir Faktorisierungen $\phi = \phi' \circ f$ und $\gamma = \gamma' \circ g$ mit $f : \Phi^* \rightarrow \Phi'^*$ und $g : \Gamma^* \rightarrow \Gamma'^*$ derart, daß $\phi' : \Phi'^* \rightarrow \Sigma^*$ und $\gamma' : \Gamma'^* \rightarrow \Sigma^*$ Isomorphismen sind, und daß außerdem $\Phi' \cap \Gamma' = \Phi \cap \Gamma = \Phi' \cap \Gamma = \emptyset$. Es gilt dann natürlich $[f, E_{\Gamma}]([F1, G1]_{\mathcal{C}}) = [f, E_{\Gamma}]([F2, G2]_{\mathcal{C}})$ und $[E_{\Phi, g}]([F1, G1]_{\mathcal{C}}) = [E_{\Phi, g}]([F2, G2]_{\mathcal{C}})$. Neben den Kooperationsformen $c = (\Phi, \Gamma, \Sigma, \phi, \gamma, C)$ und $c' = (\Phi', \Gamma', \Sigma, \phi', \gamma', C)$ benötigen wir noch $c^{\circ} = (\Phi, \Gamma', \Sigma, \phi, \gamma', C)$. Aus Satz 1 und Korollar 1 folgt jetzt $[f(F1), g(G1)]_{\mathcal{C}'} = [[f, E_{\Gamma}]([F1, G1]_{\mathcal{C}}), [E_{\Phi, g}]([F1, G1]_{\mathcal{C}})]_{\mathcal{C}'}$, $[[f, E_{\Gamma}]([F2, G2]_{\mathcal{C}}), [E_{\Phi, g}]([F2, G2]_{\mathcal{C}})]_{\mathcal{C}'}$, $= [f(F2), g(G2)]_{\mathcal{C}'}$. Nach Lemma 1 gilt dann $[f(F1), g(G2)]_{\mathcal{C}'} = [f(F2), g(G1)]_{\mathcal{C}'}$, $= [f(F1), g(G1)]_{\mathcal{C}'}$, $= [f(F2), g(G2)]_{\mathcal{C}'}$, woraus mit Korollar 2 und 3 $[I_{\Phi, E_{\Gamma}}]([F1, G2]_{\mathcal{C}}) = [I_{\Phi, E_{\Gamma'}}]([F1, g(G2)]_{\mathcal{C}^{\circ}}) = [I_{\Phi, E_{\Gamma'}}]([F1, g(G1)]_{\mathcal{C}^{\circ}}) = [I_{\Phi, E_{\Gamma}}]([F1, G1]_{\mathcal{C}})$ folgt. Die andere Gleichung der Behauptung läßt sich entsprechend zeigen.

Unter Benutzung von Satz 3 kann Satz 5 auch auf APA angewendet werden. Die wesentliche Bedeutung von Satz 3 liegt darin, das $[\mu_{\mathbb{T}1}, \mu_{\mathbb{T}2}]$ -Bild der Schaltfolgensprache L einer direkten Summe $A1 \oplus A2$ zweier

APA als ein Kooperationsprodukt $[L1, L2]_{\Gamma}$ von Sprachen $L1 \subset \Psi 1^*$ und $L2 \subset \Psi 2^*$ darzustellen, wobei $L1$ bzw. $L2$ nur von $A1$ bzw. $A2$ und den gemeinsamen Zustandskomponenten von $A1$ und $A2$ abhängen. Darüber hinaus folgt aus Satz 3 auch noch $\mu_{T1}(L) \subset L1$ sowie $\mu_{T2}(L) \subset L2$. Es gilt also $[\mu_{T1}, \mu_{T2}](L) = [L1, L2]_{\Gamma} = [[I_{\Psi 1}, E_{\Psi 2}](\mu_{T1}, \mu_{T2})(L), [E_{\Psi 1}, I_{\Psi 2}](\mu_{T1}, \mu_{T2})(L)]_{\Gamma} = [\mu_{T1}(L), \mu_{T2}(L)]_{\Gamma}$.

Mittels Satz 1 kann jetzt die Bestimmung gewisser homomorpher Bilder von $[\mu_{T1}, \mu_{T2}](L)$ auf die Bestimmung entsprechender Bilder von $\mu_{T1}(L)$ und $\mu_{T2}(L)$ zurückgeführt werden. Aus Komplexitätsgründen (Zustandsexplosion) ist es daher interessant, $\mu_{T1}(L)$ und $\mu_{T2}(L)$ zu bestimmen, ohne eine Darstellung von L (Erreichbarkeitsgraph von $A1 \oplus A2$) zu benutzen. Dazu liefert jetzt Satz 5 hinreichende Bedingungen.

Wir betrachten jetzt vier APA $F1, F2, G1$ und $G2$, welche den entsprechenden Bedingungen genügen, damit die direkten Sumen $F1 \oplus G1$, $F1 \oplus G2$ und $F2 \oplus G2$ definiert sind und in allen drei direkten Summen die jeweils beteiligten APA die gleichen gemeinsamen Zustandskomponenten besitzen. Es seien $L11 \subset \Psi 11^*$, $L12 \subset \Psi 12^*$ und $L22 \subset \Psi 22^*$ die Schaltfolgensprachen von $F1 \oplus G1$, $F1 \oplus G2$ und $F2 \oplus G2$. Mit diesen Voraussetzungen impliziert Satz 3 die folgenden drei Aussagen:

Es existieren Alphabete $\Phi 1, \Phi 2, \Gamma 1$ sowie $\Gamma 2$ und Homomorphismen $\mu_{F1} : (\Psi 11 \cup \Psi 12)^* \rightarrow \Phi 1^*$, $\mu_{G1} : \Psi 11^* \rightarrow \Gamma 1^*$, $\mu_{F2} : \Psi 22^* \rightarrow \Phi 2^*$ sowie $\mu_{G2} : (\Psi 12 \cup \Psi 22)^* \rightarrow \Gamma 2^*$, welche bezüglich $F1 \oplus G1$, $F1 \oplus G2$ und $F2 \oplus G2$ die entsprechenden Modulhomomorphismen darstellen.

Es existiert ein Alphabet Σ und Homomorphismen $\kappa_F : \Phi^* \rightarrow \Sigma^*$ sowie $\kappa_G : \Gamma^* \rightarrow \Sigma^*$ mit $\Phi = \Phi 1 \cup \Phi 2$ und $\Gamma = \Gamma 1 \cup \Gamma 2$, sodaß $\rho_{F1} = \kappa_F \circ \mu_{F1}$, $\rho_{F2} = \kappa_F \circ \mu_{F2}$, $\rho_{G1} = \kappa_G \circ \mu_{G1}$ sowie $\rho_{G2} = \kappa_G \circ \mu_{G2}$ bezüglich $F1 \oplus G1$, $F1 \oplus G2$ und $F2 \oplus G2$ die entsprechenden Randhomomorphismen darstellen.

Es existieren Sprachen $R \subset \Sigma^*$, $F1 \subset \Phi 1^*$, $F2 \subset \Phi 2^*$, $G1 \subset \Gamma 1^*$ sowie $G2 \subset \Gamma 2^*$, sodaß $[\mu_{F1}, \mu_{G1}](L11) = [F1, G1]_{\Gamma}$, $[\mu_{F1}, \mu_{G2}](L12) = [F1, G2]_{\Gamma}$ und $[\mu_{F2}, \mu_{G2}](L22) = [F2, G2]_{\Gamma}$, wobei die Kooperationsform r durch $r = (\Phi, \Gamma, \Sigma, \kappa_F, \kappa_G, R)$ gegeben ist.

Satz 6:

Mit den obigen Bezeichnungen gilt $\mu_{F1}(L12) = \mu_{F1}(L11)$ und $\mu_{G2}(L12) = \mu_{G2}(L22)$ falls $\rho_{F1}(L11) = \rho_{F2}(L22)$ und $\rho_{G1}(L11) = \rho_{G2}(L22)$.

Beweis:

Unter diesen Voraussetzungen gilt $[\kappa_F, E_{\Gamma}]([F1, G1]_{\Gamma}) = [\kappa_F, E_{\Gamma}]([\mu_{F1}, \mu_{G1}](L11)) = (\kappa_F \circ \mu_{F1})(L11) = \rho_{F1}(L11) = \rho_{F2}(L22) = (\kappa_F \circ \mu_{F2})(L22) = [\kappa_F, E_{\Gamma}]([\mu_{F2}, \mu_{G2}](L22)) = [\kappa_F, E_{\Gamma}]([F2, G2]_{\Gamma})$. Entsprechend läßt sich $[E_{\Phi}, \kappa_G]([F1, G1]_{\Gamma}) = [E_{\Phi}, \kappa_G]([F2, G2]_{\Gamma})$ zeigen. Damit sind alle Voraussetzungen von Satz 5

erfüllt, und es gilt $\mu_{F_1}(L12) = [I_{\Phi}, E_{\Gamma}](\mu_{F_1}, \mu_{G_1})(L12) = [I_{\Phi}, E_{\Gamma}](F1, G2)_{\Gamma} = [I_{\Phi}, E_{\Gamma}](F1, G1)_{\Gamma} = [I_{\Phi}, E_{\Gamma}](\mu_{F_1}, \mu_{G_1})(L11) = \mu_{F_1}(L11)$. Die zweite Gleichung des Satzes läßt sich genauso beweisen.

Dieser Satz ist für solche Situationen von Interesse, wo die Erreichbarkeitsgraphen von $F_1 \oplus G_1$ und $F_2 \oplus G_2$ kleiner als der Erreichbarkeitsgraph von $F_1 \oplus G_2$ sind. Erreichbarkeitsgraphen von direkten Summen von APA werden dann besonders groß, wenn beide Komponenten viele "interne Aktionen" besitzen. Zur Nutzung von Satz 6 sollten also in G_1 bzw. F_2 weniger "interne Aktionen" als in G_2 bzw. F_1 auftreten. Unter diesen Voraussetzungen kann dann Satz 6 für eine "divide et impera"- Strategie benutzt werden. In [Och94b, Och95] wurde an einem umfangreichen Beispiel demonstriert, daß dieses Vorgehen bei "wohlstrukturierten" Spezifikationen zur drastischen Komplexitätsreduktion bei der Berechnung gewisser homomorpher Bilder von Schaltfolgensprachen führen kann und ähnlich wie im ersten Kapitel auch für Induktionsbeweise eingesetzt werden kann.

4 Schlichtheit und Kooperativität

Bezüglich des dynamischen Verhaltens von Systemen unterscheidet man üblicherweise /AS85/ zwei Arten von Eigenschaften, nämlich *Sicherheits-* und *Lebendigkeitseigenschaften*. Salopp gesprochen sind Sicherheitseigenschaften vom Typ “es geschieht nichts Falsches” und Lebendigkeitseigenschaften vom Typ “immer mal wieder geschieht etwas Gewünschtes”. Bezogen auf unser einführendes Server-Resource-Beispiel ist die Tatsache, daß während der Bearbeitung eines Auftrags die Resource für den Server reserviert ist (Abb. 6), eine typische Sicherheitseigenschaft. Es ist unschwer einzusehen, daß Sicherheitseigenschaften eines homomorphen Bildes in entsprechender Form auch für das ursprüngliche Systemverhalten gelten.

Neben der obigen Sicherheitseigenschaft besitzt der Automat in Abb. 6 auch die Lebendigkeitseigenschaft, daß immer wieder ein Auftrag bearbeitet wird. Für die Verifikation kooperierender Systeme, welche ja i. allg. ein nichtdeterministisches Systemverhalten zeigen, müssen allerdings Lebendigkeitseigenschaften in einer etwas abgeschwächteren Weise, als oben angeführt, betrachtet werden. Unser Beispielsystem $[F1, G1]_{c1}$ besitzt zwar die Eigenschaft, daß immer wieder ein Auftrag *bearbeitet werden kann*, aber nicht die Eigenschaft, daß immer wieder ein Auftrag *tatsächlich bearbeitet wird*. Letztere Eigenschaft wird nur von dem homomorphen Bild in Abb. 6 erfüllt. Solche “Möglichkeiten von Verhaltensweisen” wurden in /NO95/ mit dem Begriff des *approximativen Erfüllens von Eigenschaften* präzisiert.

Selbst derartig abgeschwächte Lebendigkeitseigenschaften lassen sich aber nicht ohne weiteres von einem homomorphen Bild auf das ursprüngliche Systemverhalten übertragen. Man denke sich dazu eine Modifikation von $G1$ (Abb. 8) zu $G1'$, wobei in $G1'$ die Aktion b_i fehlt. Die Sprache $[F1, G1']_{c1}$ besitzt dann zwar bezüglich des betrachteten Homomorphismus v_1 das gleiche Bild wie $[F1, G1]_{c1}$, aber nicht mehr die oben beschriebene Lebendigkeitseigenschaft. Geschieht nämlich einmal die Aktion b_n , dann kann danach nie mehr ein Auftrag bearbeitet werden.

Fordert man allerdings zusätzliche Eigenschaften der Homomorphismen, dann lassen sich auch Lebendigkeitseigenschaften eines homomorphen Bildes auf das ursprüngliche Systemverhalten übertragen. Dazu wurde in /Och92, Och94a/ der Begriff der *Schlichtheit* von Homomorphismen eingeführt. Ist $L \subset \Sigma^*$ eine präfixstabile Sprache, welche das dynamische Verhalten eines Systems beschreibt, dann sind gewissermaßen die Linksquotienten $x^{-1}(L)$ verantwortlich für die Lebendigkeitseigenschaften des Systems. Die Tatsache, daß $h(x^{-1}(L)) \subset h(x)^{-1}(h(L))$ für einen Homomorphismus $h : \Sigma^* \rightarrow \Sigma'^*$, und daß diese Inklusion i. allg. echt ist, ist der eigentliche Grund dafür, daß Lebendigkeitseigenschaften eines homomorphen Bildes nicht ohne weiteres auf das ursprüngliche Systemverhalten übertragen werden können.

In der Definition der Schlichtheit wird deshalb gefordert, daß die obige Inklusion nicht “allzu echt” ist: Der Homomorphismus h heißt *schlicht auf L* , falls es zu jedem $x \in L$ ein $w \in h(x)^{-1}(h(L))$ mit $w^{-1}(h(x^{-1}(L))) = (h(x)w)^{-1}(h(L))$ gibt. Eine genauere Diskussion dieser Definition findet sich in /Och92, Och94a, NO95, OP95/; einige wichtige Eigenschaften schlichter Homomorphismen sind gegen Ende dieses Kapitels für Betrachtungen an unserem Client-Server-Beispiel zusammengestellt.

In /NO95/ wurde gezeigt, daß der Begriff des approximativen Erfüllens von Eigenschaften (zumindest für reguläre Sprachen) genau mit dem Begriff der Schlichtheit von Homomorphismen zusammenpaßt: Die Schlichtheit von Homomorphismen auf regulären Sprachen ist dadurch charakterisiert, daß alle Eigenschaften (Sicherheits- und Lebendigkeitseigenschaften), welche von homomorphen Bildern approximativ erfüllt werden, in entsprechender Form auch von den ursprünglichen Sprachen erfüllt werden.

Es ist also äußerst sinnvoll, Methoden anzugeben, welche neben der Bestimmung homomorpher Bilder von Kooperationsprodukten auch die Überprüfung der Schlichtheit dieser Homomorphismen erlauben. Dazu kann man sich leicht überlegen, daß $x^{-1}([F,G]_C) = [([I_\Phi, E_\Gamma](x))^{-1}(F), ([E_\Phi, I_\Gamma](x))^{-1}(G)]_{cx}$ für jedes $x \in [F,G]_C$, wobei $cx = (\Phi, \Gamma, \Sigma, \phi, \gamma, ([\phi, \gamma](x))^{-1}(C))$. Dies ermöglicht es, aus den Sätzen 1 und 4 nützliche Kriterien für die Schlichtheit von $[f, I_\Gamma]$, $[I_\Phi, g]$ und $[f, g]$ auf $[F,G]_C$ abzuleiten. Zum Beweis entsprechender Sätze werden noch zwei Lemmata benötigt.

Lemma 2:

Sind f und g zwei Homomorphismen, die schlicht auf einer formalen Sprache L sind, dann gibt es zu jedem $x \in L$ ein $y \in x^{-1}(L)$ mit $f(y)^{-1}(f(x^{-1}(L))) = f(xy)^{-1}(f(L))$ und $g(y)^{-1}(g(x^{-1}(L))) = g(xy)^{-1}(g(L))$.

Beweis:

Wegen der Schlichtheit von f gibt es zu jedem $x \in L$ ein $u \in x^{-1}(L)$ mit $f(u)^{-1}(f(x^{-1}(L))) = f(xu)^{-1}(f(L))$. Wegen der Schlichtheit von g gibt es zu $xu \in L$ ein $v \in (xu)^{-1}(L)$ mit $g(v)^{-1}(g((xu)^{-1}(L))) = g(xuv)^{-1}(g(L))$. $y = uv$ erfüllt die Behauptung des Lemmas, denn $f(y)^{-1}(f(x^{-1}(L))) = f(v)^{-1}(f(u)^{-1}(f(x^{-1}(L)))) = f(v)^{-1}(f(xu)^{-1}(f(L))) = f(xuv)^{-1}(f(L))$ und $g(y)^{-1}(g(x^{-1}(L))) = g(v)^{-1}(g(u)^{-1}(g(x^{-1}(L)))) \supseteq g(v)^{-1}(g((xu)^{-1}(L))) = g(xuv)^{-1}(g(L))$. Wegen $g(x^{-1}(L)) \subset g(x)^{-1}(g(L))$ gilt dann auch $g(y)^{-1}(g(x^{-1}(L))) = g(xuv)^{-1}(g(L))$.

Lemma 3:

Sind $l : \Phi^* \rightarrow \Gamma^*$, $m : \Gamma^* \rightarrow \Delta^*$ und $n : \Phi^* \rightarrow \Delta^*$ drei Homomorphismen mit $n = m \circ l$ und $L \subset \Delta^*$, dann gilt $m^{-1}(n(x)^{-1}(L)) = l(x)^{-1}(m^{-1}(L))$ für jedes $x \in \Phi^*$.

Beweis:

Es gilt $y \in m^{-1}(n(x)^{-1}(L)) \Leftrightarrow m(y) \in n(x)^{-1}(L) \Leftrightarrow n(x)m(y) \in L \Leftrightarrow m(l(x)m(y)) \in L \Leftrightarrow m(l(xy)) \in L \Leftrightarrow l(x)y \in m^{-1}(L) \Leftrightarrow y \in l(x)^{-1}(m^{-1}(L))$.

Satz 7:

Sind unter den Voraussetzungen von Satz 1 $[f, E_\Gamma]$ und $[E_\Phi, g]$ schlicht auf $[F,G]_C$, dann ist auch $[f, g]$ schlicht auf $[F,G]_C$.

Beweis:

Mit der Bemerkung vor Lemma 2 folgt aus Korollar 1 $[f, g](x^{-1}([F,G]_C)) = [[f, E_\Gamma]([I_\Phi, E_\Gamma](x)^{-1}(F), [E_\Phi, I_\Gamma](x)^{-1}(G))]_{cx}, [E_\Phi, g]([I_\Phi, E_\Gamma](x)^{-1}(F), [E_\Phi, I_\Gamma](x)^{-1}(G))]_{cx}$, wobei $cx' =$

$(\Phi', \Gamma', \Sigma, \phi', \gamma', [\phi, \gamma](x)^{-1}(C))$. Damit gilt $[f, g](x^{-1}([F, G]_c)) = [[f, E_\Gamma](x^{-1}([F, G]_c)), [E_{\Phi, g}(x^{-1}([F, G]_c))]_{c_X}$,
 $= [I_{\Phi', E_\Gamma}]^{-1}([f, E_\Gamma](x^{-1}([F, G]_c))) \cap [E_{\Phi', I_\Gamma}]^{-1}([E_{\Phi, g}(x^{-1}([F, G]_c))]) \cap [\phi', \gamma']^{-1}([\phi, \gamma](x)^{-1}(C))$.

Nach Lemma 2 gibt es zu jedem $x \in [F, G]_c$ ein $y \in x^{-1}([F, G]_c)$ mit $[f, E_\Gamma](y)^{-1}([f, E_\Gamma](x^{-1}([F, G]_c))) = [f, E_\Gamma](xy)^{-1}([f, E_\Gamma]([F, G]_c))$ und $[E_{\Phi, g}(y)^{-1}([E_{\Phi, g}(x^{-1}([F, G]_c))]) = [E_{\Phi, g}(xy)^{-1}([E_{\Phi, g}([F, G]_c))])$. Für dieses y gilt jetzt $[f, g](y)^{-1}([f, g](x^{-1}([F, G]_c))) = [f, g](y)^{-1}([I_{\Phi', E_\Gamma}]^{-1}([f, E_\Gamma](x^{-1}([F, G]_c)))) \cap [f, g](y)^{-1}([E_{\Phi', I_\Gamma}]^{-1}([E_{\Phi, g}(x^{-1}([F, G]_c))])) \cap [f, g](y)^{-1}([\phi', \gamma']^{-1}([\phi, \gamma](x)^{-1}(C)))$. Beachtet man, daß $[f, E_\Gamma] = [I_{\Phi', E_\Gamma}] \circ [f, g]$, $[E_{\Phi, g}] = [E_{\Phi', I_\Gamma}] \circ [f, g]$ und $[\phi, \gamma] = [\phi', \gamma'] \circ [f, g]$, dann folgt daraus mit Lemma 3 $[f, g](y)^{-1}([f, g](x^{-1}([F, G]_c))) = [I_{\Phi', E_\Gamma}]^{-1}([f, E_\Gamma](y)^{-1}([f, E_\Gamma](x^{-1}([F, G]_c)))) \cap [E_{\Phi', I_\Gamma}]^{-1}([E_{\Phi, g}(y)^{-1}([E_{\Phi, g}(x^{-1}([F, G]_c))])) \cap [f, g](y)^{-1}([\phi', \gamma']^{-1}(C)) = [I_{\Phi', E_\Gamma}]^{-1}([f, E_\Gamma](xy)^{-1}([f, E_\Gamma]([F, G]_c))) \cap [E_{\Phi', I_\Gamma}]^{-1}([E_{\Phi, g}(xy)^{-1}([E_{\Phi, g}([F, G]_c))])) \cap [f, g](xy)^{-1}([\phi', \gamma']^{-1}(C)) = [f, g](xy)^{-1}([I_{\Phi', E_\Gamma}]^{-1}([f, E_\Gamma]([F, G]_c))) \cap [f, g](xy)^{-1}([E_{\Phi', I_\Gamma}]^{-1}([E_{\Phi, g}([F, G]_c))])) \cap [f, g](xy)^{-1}([\phi', \gamma']^{-1}(C)) = [f, g](xy)^{-1}([I_{\Phi', E_\Gamma}]^{-1}([f, E_\Gamma]([F, G]_c))) \cap [E_{\Phi', I_\Gamma}]^{-1}([E_{\Phi, g}([F, G]_c))) \cap [\phi', \gamma']^{-1}(C) = [f, g](xy)^{-1}([f, E_\Gamma]([F, G]_c), [E_{\Phi, g}([F, G]_c)]) = [f, g](xy)^{-1}([f, g]([F, G]_c))$, was die Schlichtheit von $[f, g]$ auf $[F, G]_c$ beweist.

Satz 8:

Ist unter den Voraussetzungen von Satz 4 $[f, g]$ schlicht auf $[F, G]_c$, dann ist auch $[I_{\Phi, g}]$ schlicht auf $[F, G]_c$.

Beweis:

Aus Satz 4 und der Bemerkung vor Lemma 2 folgt $[I_{\Phi, g}(x^{-1}([F, G]_c))] = [I_{\Phi, g}([I_{\Phi, E_\Gamma}(x)^{-1}(F), [E_{\Phi, I_\Gamma}(x)^{-1}(G)]_{c_X})] = [f, I_\Gamma]^{-1}([f, g]([I_{\Phi, E_\Gamma}(x)^{-1}(F), [E_{\Phi, I_\Gamma}(x)^{-1}(G)]_{c_X})]) \cap [I_{\Phi, E_\Gamma}(x)^{-1}(F), \Gamma^*]_\diamond = [f, I_\Gamma]^{-1}([f, g](x^{-1}([F, G]_c))) \cap [I_{\Phi, g}(x)^{-1}([F, \Gamma^*]_\diamond)]$ für jedes $x \in [F, G]_c$. Wegen der Schlichtheit von $[f, g]$ auf $[F, G]_c$ gibt es zu jedem $x \in [F, G]_c$ ein $y \in x^{-1}([F, G]_c)$ mit $[f, g](y)^{-1}([f, g](x^{-1}([F, G]_c))) = [f, g](xy)^{-1}([f, g]([F, G]_c))$. Unter Benutzung von Lemma 3 folgt dann $[I_{\Phi, g}(y)^{-1}([I_{\Phi, g}(x^{-1}([F, G]_c))]) = [I_{\Phi, g}(y)^{-1}([f, I_\Gamma]^{-1}([f, g](x^{-1}([F, G]_c)))] \cap [I_{\Phi, g}(y)^{-1}([I_{\Phi, g}(x)^{-1}([F, \Gamma^*]_\diamond))]) = [f, I_\Gamma]^{-1}([f, g](y)^{-1}([f, g](x^{-1}([F, G]_c))))) \cap [I_{\Phi, g}(xy)^{-1}([F, \Gamma^*]_\diamond)] = [f, I_\Gamma]^{-1}([f, g](xy)^{-1}([f, g]([F, G]_c)))) \cap [I_{\Phi, g}(xy)^{-1}([F, \Gamma^*]_\diamond)] = [I_{\Phi, g}(xy)^{-1}([f, I_\Gamma]^{-1}([f, g]([F, G]_c)))] \cap [I_{\Phi, g}(xy)^{-1}([F, \Gamma^*]_\diamond)] = [I_{\Phi, g}(xy)^{-1}([f, I_\Gamma]^{-1}([f, g]([F, G]_c)))] \cap [F, \Gamma^*]_\diamond = [I_{\Phi, g}(xy)^{-1}([I_{\Phi, g}([F, G]_c)])]$, was die Schlichtheit von $[I_{\Phi, g}]$ auf $[F, G]_c$ zeigt.

Zur Zurückführung der Schlichtheit von $[f, g]$ auf $[F, G]_c$ auf die Schlichtheit entsprechender Homomorphismen auf "einfacheren" Sprachen wird noch ein zusätzlicher Begriff benötigt. Dazu sei $L \subset (\Phi \cup \Gamma)^*$, $f: \Phi^* \rightarrow \Phi^*$ und $\Phi \cap \Gamma = \Phi' \cap \Gamma = \emptyset$. Für eine formale Sprache H sei $\text{pre}(H)$ die Menge aller Präfixe von Worten aus H und $\text{max}(H)$ die Menge aller Worte aus H , die nicht Präfix eines anderen Wortes aus H sind ($\text{max}(H) \subset H \subset \text{pre}(H)$).

Die Sprache $[I_{\Phi}, E_{\Gamma}](L)$ heißt *kooperativ in L bezüglich f*, falls es zu jedem $x \in L$ eine Teilmenge $H \subset [f, I_{\Gamma}](x)^{-1}([f, I_{\Gamma}](L))$ gibt, welche die folgenden Eigenschaften (1) - (3) besitzt:

- (1) $H \neq \emptyset$, $H = \text{pre}(\max(H))$ und $[I_{\Phi}, E_{\Gamma}](H)$ ist endlich.
- (2) Für $u \in \max(H)$ gilt $u^{-1}([f, I_{\Gamma}](x)^{-1}(L)) = ([f, I_{\Gamma}](x)u)^{-1}([f, I_{\Gamma}](L))$.
- (3) Für $u \in H \setminus \max(H)$ gilt
 - (i) $u^{-1}(H) \cap \Gamma = ([f, I_{\Gamma}](x)u)^{-1}([f, I_{\Gamma}](L)) \cap \Gamma$ sowie
 - (ii) $u^{-1}(H) \cap \Phi' \neq \emptyset$ falls $([f, I_{\Gamma}](x)u)^{-1}([f, I_{\Gamma}](L)) \cap \Phi' \neq \emptyset$.

Wegen $H \neq \emptyset$ und (2) impliziert diese Kooperativitätseigenschaft die Schlichtheit von $[f, I_{\Gamma}]$ auf L , und es gilt $H \subset [f, I_{\Gamma}](x)^{-1}(L)$. Die tiefere Bedeutung der Definition wird mit dem nachfolgenden Satzes 9 klar, zu dessen Beweis noch zwei Lemmata benötigt werden. Für den Beweis von Lemma 4 sei noch bemerkt, daß für jedes $u \in \max(H)$ auch die Menge $H \cup u((([f, I_{\Gamma}](x)u)^{-1}([f, I_{\Gamma}](L)) \cap \Gamma) \cup N)$ die Eigenschaften (1) - (3) besitzt, falls $N \subset ([f, I_{\Gamma}](x)u)^{-1}([f, I_{\Gamma}](L)) \cap \Phi'$ und $N \neq \emptyset$ falls $([f, I_{\Gamma}](x)u)^{-1}([f, I_{\Gamma}](L)) \cap \Phi' \neq \emptyset$.

Für viele Anwendungen reicht die folgende leichter handhabbare verschärfte Version der obigen Definition aus: Die Sprache $[I_{\Phi}, E_{\Gamma}](L)$ heißt *strikt kooperativ in L bezüglich f*, falls sie kooperativ in L bezüglich f ist und falls in der Bedingung (1) die Endlichkeit von $[I_{\Phi}, E_{\Gamma}](H)$ durch die Endlichkeit von H ersetzt wird.

Die beiden Definitionen unterscheiden sich im wesentlichen darin, daß von der allgemeineren Version auch Situationen erfaßt werden können, bei denen ein Kooperationspartner beliebig viele "Aktionen an der Schnittstelle" durchführen kann, ohne daß der andere reagiert. Da ein solches Kooperationsverhalten eher untypisch ist, kommt man bei praktischen Anwendungen in aller Regel mit der einfacheren Definition der strikten Kooperativität aus.

Lemma 4:

Ist unter den Voraussetzungen des Satzes 4 und der zusätzlichen Bedingung $g \langle \gamma \ [E_{\Phi'}, I_{\Gamma}](f(F), G)_{c^{\wedge}}$ kooperativ in $[f(F), G]_{c^{\wedge}}$ bezüglich g und $[I_{\Phi}, E_{\Gamma'}](f(F), g(G))_{c^{\circ}}$ kooperativ in $[F, g(G)]_{c^{\circ}}$ bezüglich f , wobei $c^{\wedge} = (\Phi', \Gamma, \Sigma, \phi', \gamma, C)$ und $c^{\circ} = (\Phi, \Gamma', \Sigma, \phi, \gamma', C)$, dann gibt es zu jedem $w \in [F, G]_{c^{\circ}}$ ein $p \in [f, g](w)^{-1}([f(F), g(G)]_{c^{\circ}})$ mit $p^{-1}([f, I_{\Gamma'}](I_{\Phi}, g](w)^{-1}([F, g(G)]_{c^{\circ}}))) = ([f, g](w)p)^{-1}([f(F), g(G)]_{c^{\circ}}) = p^{-1}([I_{\Phi}, g](f, I_{\Gamma'}](w)^{-1}([f(F), G]_{c^{\wedge}})))$.

Beweis:

Nach Voraussetzung gibt es zu $[f, I_{\Gamma'}](w) \in [f(F), G]_{c^{\wedge}}$ und $[I_{\Phi}, g](w) \in [F, g(G)]_{c^{\circ}}$ jeweils eine Teilmenge $H_G \subset [f, g](w)^{-1}([f(F), g(G)]_{c^{\circ}})$ sowie $H_F \subset [f, g](w)^{-1}([f(F), g(G)]_{c^{\circ}})$, die beide entsprechende Kooperativitätseigenschaften (1) - (3) besitzen. Wegen (1) ist $\varepsilon \in H_G \cap H_F$, und wegen (3) ist $u^{-1}(H_G) \cap u^{-1}(H_F) \cap (\Phi' \cup \Gamma') \neq \emptyset$ für jedes $u \in (H_G \setminus \max(H_G)) \cap (H_F \setminus \max(H_F))$. Damit kann schrittweise ein $u \in H_G \cap H_F$ konstruiert werden, welches wegen der Endlichkeitsbedingung in (1) auch ein Element von $\max(H_G) \cup \max(H_F)$ ist. O.B.d.A. sei $u \in \max(H_G) \cap H_F$. Da $H_F = \text{pre}(\max(H_F))$ gibt es ein $v \in u^{-1}(H_F)$ mit $uv \in \max(H_F)$. Mit der obigen Bemerkung zur Kooperativitätsdefinition kann H_G

derart zu einer Menge H_G' erweitert werden, daß $uv \in \max(H_G')$ und daß H_G' die Eigenschaften (1) - (3) besitzt. Wegen (2) erfüllt jetzt $p = uv$ die Behauptung des Lemmas.

Lemma 5:

Sind unter den Voraussetzungen des Satzes 4 und der zusätzlichen Bedingung $g \langle \gamma \ uv \in [F,G]_C$, $st \in [F,G]_C$ sowie $w \in [F,G]_C$ mit $[I_{\Phi},g](u) = [I_{\Phi},g](w)$, $[f,I_{\Gamma}](s) = [f,I_{\Gamma}](w)$ und $[f,g](v) = [f,g](t)$, dann existiert ein $z \in w^{-1}([F,G]_C)$ mit $[I_{\Phi},g](z) = [I_{\Phi},g](v)$ und $[f,I_{\Gamma}](z) = [f,I_{\Gamma}](t)$.

Beweis: Induktion über die Länge von $[f,g](v) = [f,g](t)$

Wesentlich für den Beweis ist die Tatsache, daß $[f,g] \langle [\phi, \gamma]$. Ist $[f,g](v) = [f,g](t) = \varepsilon$, dann erfüllt $z = [I_{\Phi},E_{\Gamma}](v)[E_{\Phi},I_{\Gamma}](t)$ die Behauptung des Lemmas. Im Fall $[f,g](v) = [f,g](t) \neq \varepsilon$ gibt es Zerlegungen $v = v^{\circ}v^{\wedge}$ und $t = t^{\circ}t^{\wedge}$ mit $[f,g](v^{\circ}) = [f,g](t^{\circ})$, $[f,g](v^{\wedge}) = [f,g](t^{\wedge}) = \varepsilon$. Nach Induktionsvoraussetzung gibt es zu $uv' \in [F,G]_C$ und $st' \in [F,G]_C$ ein $z' \in w^{-1}([F,G]_C)$, das die entsprechende Aussage des Lemmas erfüllt. Für $uv^{\circ}v^{\wedge} \in [F,G]_C$ und $st^{\circ}t^{\wedge} \in [F,G]_C$ erfüllt jetzt $z = z'[I_{\Phi},E_{\Gamma}](v^{\circ})[E_{\Phi},I_{\Gamma}](t^{\circ})[I_{\Phi},E_{\Gamma}](v^{\wedge})[E_{\Phi},I_{\Gamma}](t^{\wedge})$ die Behauptung des Lemmas.

Satz 9: Unter den Voraussetzungen von Lemma 4 ist $[f,g]$ schlicht auf $[F,G]_C$.

Beweis:

Es genügt, $p^{-1}([f,g](w^{-1}([F,G]_C))) \supset ([f,g](w)p)^{-1}([f,g]([F,G]_C))$ für jedes $w \in [F,G]_C$ und ein geeignetes $p \in [f,g](w)^{-1}([f,g]([F,G]_C))$ zu zeigen. Nach Lemma 4 gibt es zu jedem $w \in [F,G]_C$ ein $p \in [f,g](w)^{-1}([f,g]([F,G]_C))$ mit $p^{-1}([f,I_{\Gamma}]([I_{\Phi},g](w)^{-1}([I_{\Phi},g]([F,G]_C)))) = ([f,g](w)p)^{-1}([f,g]([F,G]_C)) = p^{-1}([I_{\Phi},g]([f,I_{\Gamma}](w)^{-1}([f,I_{\Gamma}]([F,G]_C))))$. Zu jedem $q \in ([f,g](w)p)^{-1}([f,g]([F,G]_C))$ existiert demnach $uv^{\circ}v' \in [F,G]_C$ und $st^{\circ}t' \in [F,G]_C$ mit $[I_{\Phi},g](u) = [I_{\Phi},g](w)$, $[f,I_{\Gamma}](s) = [f,I_{\Gamma}](w)$, $[f,g](v^{\circ}) = [f,g](t^{\circ}) = p$ und $[f,g](v') = [f,g](t') = q$. Aus Lemma 5 folgt dann die Existenz eines $z \in w^{-1}([F,G]_C)$ mit $[f,g](z) = pq$. Damit gilt $q \in p^{-1}([f,g](w^{-1}([F,G]_C)))$, was die gewünschte Inklusion beweist.

Die Sätze 7, 8 und 9 können bei unserem Server-Resource-Beispiel aus dem ersten Kapitel dazu benutzt werden, die Schlichtheit des Homomorphismus v_1 auf $[F_1, G_1]_{C_1}$ nachzuweisen, ohne dazu $[F_1, G_1]_{C_1}$ explizit zu benutzen. Für das Beispiel folgt dann aus dieser Schlichtheit, daß immer wieder ein Auftrag vom Server bearbeitet werden kann. Zur Anwendung obiger Sätze werden noch einige Fakten über schlichte Homomorphismen benötigt /Och92,Och94a,OP95/.

Für eine Sprache $L \subset \Sigma^*$ und einen Homomorphismus $h : \Sigma^* \rightarrow \Sigma'^*$ sei die *Deadlocksprache* DL durch $DL = \{ u \in h(L) \mid \text{es existiert } x \in L \text{ mit } u = h(x) \}$, und $h(x^{-1}(L)) = \{\varepsilon\}$ und die *Terminierungssprache* TL durch $TL = \{ u \in h(L) \mid u^{-1}(h(L)) = \{\varepsilon\} \}$ definiert. Ist der Homomorphismus h schlicht auf L , dann gilt $TL = DL$.

Für zwei Homomorphismen $h : \Sigma_1^* \rightarrow \Sigma_2^*$ und $g : \Sigma_2^* \rightarrow \Sigma_3^*$ gilt: Ist h schlicht auf $L \subset \Sigma_1^*$ und g schlicht auf $h(L) \subset \Sigma_2^*$, dann ist $g \circ h$ schlicht auf L ; ist $g \circ h$ schlicht auf L , dann ist g schlicht auf $h(L)$.

Aus der Graphentheorie ist bekannt, daß sich in jedem gerichteten Graphen die Knotenmenge in sogenannte starke Zusammenhangskomponenten partitionieren läßt. Eine starke Zusammenhangskomponente nennen wir *tot*, wenn aus ihr keine Kante herausführt. Sei L eine Sprache, die von einem endlichen Automaten \mathbb{A} erkannt wird und h ein Homomorphismus auf L . Falls es zu jedem $x \in L$ eine Fortsetzung $y \in L$ gibt (also $x \in \text{pre}(y)$), die im Automaten \mathbb{A} in eine tote Zusammenhangskomponente führt, so daß jedes $y' \in L$ mit $h(y) = h(y')$ in \mathbb{A} ebenfalls in diese tote Zusammenhangskomponente führt, dann ist h schlicht auf L . Insbesondere ist h schlicht, falls \mathbb{A} streng zusammenhängend ist.

Im ersten Kapitel wurde gezeigt, daß $v_1 = v \circ [f_1, g_1]$ und daß $[f_1, g_1]([F_1, G_1]_{c_1}) = [F, G]_c$. Da der Minimalautomat von $[F, G]_c$ (Abb. 5) streng zusammenhängend ist, ist v schlicht auf $[f_1, g_1]([F_1, G_1]_{c_1})$. Zum Nachweis der Schlichtheit von v_1 auf $[F_1, G_1]_{c_1}$ muß also noch die Schlichtheit von $[f_1, g_1]$ auf $[F_1, G_1]_{c_1}$ gezeigt werden. Nach Satz 7 genügt es jetzt, die Schlichtheit von $[f_1, E_{\Gamma_1}]$ sowie $[E_{\Phi_1}, g_1]$ auf $[F_1, G_1]_{c_1}$ zu beweisen. Satz 8 und 9 ermöglichen es, diesen Nachweis durch Schlichtheitsuntersuchungen auf "einfacheren" Sprachen als $[F_1, G_1]_{c_1}$ zu führen.

Sei dazu $\Phi' = \{a?, a+, a-, a!\}$, $\Gamma' = \{b?, b+, b-, b!\}$, $f' : \Phi_1^* \rightarrow \Phi'^*$ mit $f' = [I_{\Phi'}, E_{\Phi_1 \setminus \Phi'}]$, $g' : \Gamma_1^* \rightarrow \Gamma'^*$ mit $g' = [I_{\Gamma'}, E_{\Gamma_1 \setminus \Gamma'}]$, $\phi' : \Phi'^* \rightarrow \Sigma^*$ mit $\phi'(a?) = s?$, $\phi'(a+) = r+$, $\phi'(a-) = r-$, $\phi'(a!) = s!$, $\gamma' : \Gamma'^* \rightarrow \Sigma^*$ mit $\gamma'(b?) = r?$, $\gamma'(b+) = s+$, $\gamma'(b-) = s-$, $\gamma'(b!) = r!$, $c' = (\Phi', \Gamma', \Sigma, \phi', \gamma', C)$, $c^\circ = (\Phi', \Gamma_1, \Sigma, \phi_1, \gamma_1, C)$ und $c^\wedge = (\Phi_1, \Gamma', \Sigma, \phi_1, \gamma', C)$. Die Minimalautomaten von $f'(F_1)$, $g'(G_1)$, $[f'(F_1), g'(G_1)]_{c'}$, $[f'(F_1), G_1]_{c^\circ}$ und $[F_1, g'(G_1)]_{c^\wedge}$ sind in den Abbildungen 21, 22, 23, 24 und 25 dargestellt.

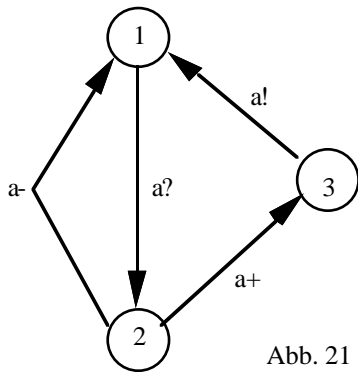


Abb. 21

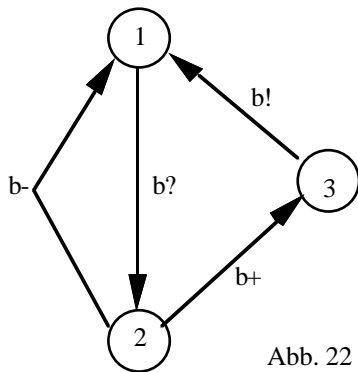


Abb. 22

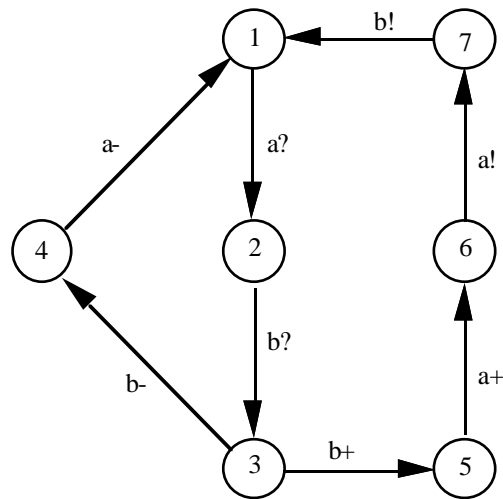


Abb. 23

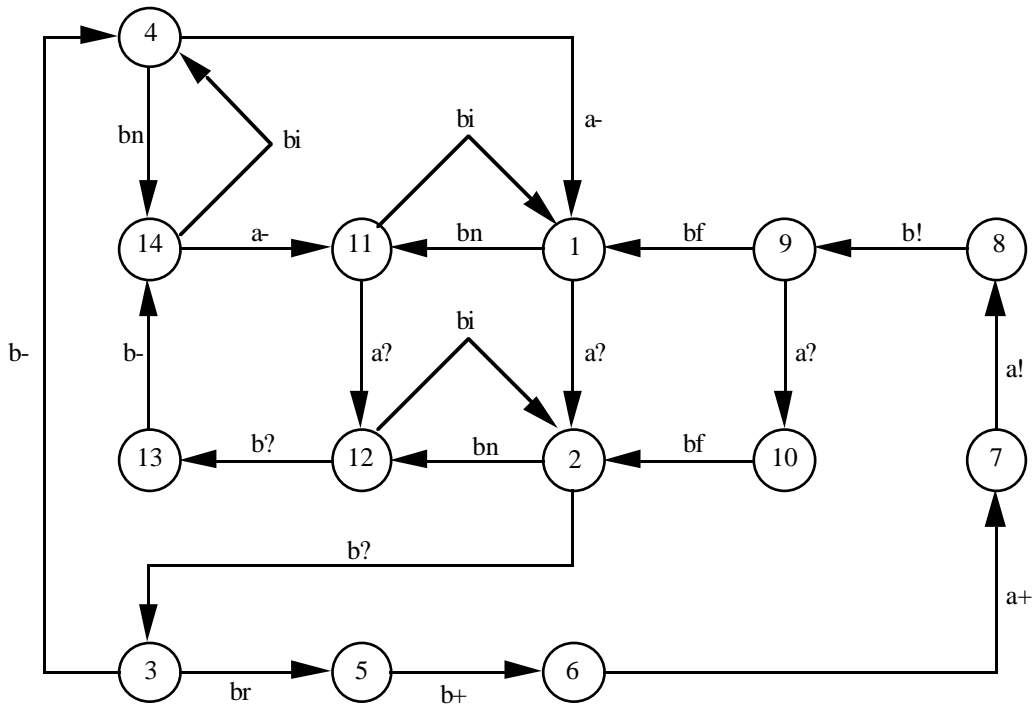


Abb. 24

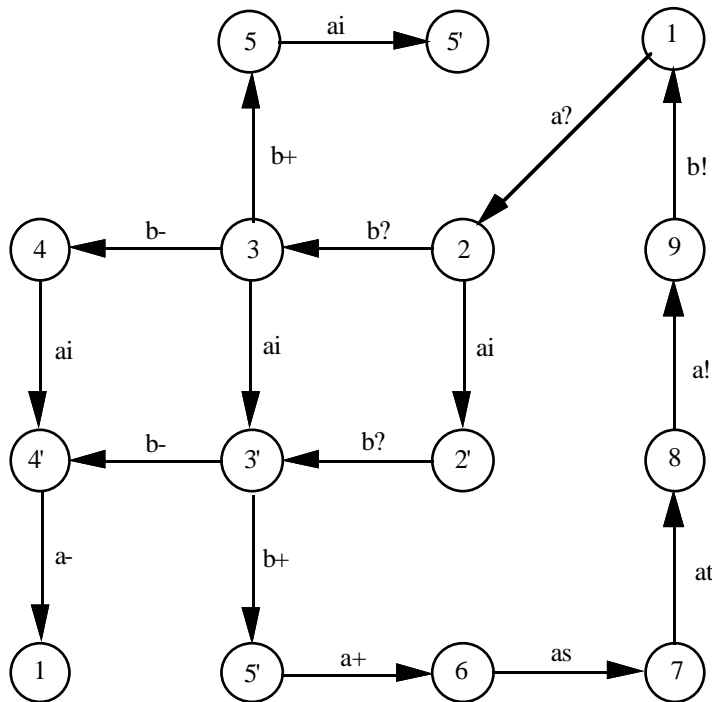


Abb. 25

Mit den Minimalautomaten von $[f'(F1),g'(G1)]_{\mathcal{C}}$, $[f'(F1),G1]_{\mathcal{C}^o}$ und $[F1,g'(G1)]_{\mathcal{C}^{\wedge}}$ läßt sich zeigen, daß $[E_{\Phi},I_{\Gamma}][f'(F1),G1]_{\mathcal{C}^o}$ strikt kooperativ in $[f'(F1),G1]_{\mathcal{C}^o}$ bezüglich g' ist und daß $[I_{\Phi},E_{\Gamma}][F1,g'(G1)]_{\mathcal{C}^{\wedge}}$ strikt kooperativ in $[F1,g'(G1)]_{\mathcal{C}^{\wedge}}$ bezüglich f' ist. Aus Satz 9 folgt dann, daß $[f',g']$ schlicht auf $[F1,G1]_{\mathcal{C}1}$ ist. Wegen Satz 8 sind dann auch $[I_{\Phi},g']$ und $[f',I_{\Gamma}]$ schlicht auf $[F1,G1]_{\mathcal{C}1}$. Es gilt $[f1,E_{\Gamma}] = [f1,E_{\Gamma}] \circ [I_{\Phi},g']$ und $[E_{\Phi},g1] = [E_{\Phi},g1] \circ [f',I_{\Gamma}]$. $[f1,E_{\Gamma}]$ bzw. $[E_{\Phi},g1]$ ist schlicht auf

$[I_{\Phi_1, g'}]([F1, G1]_{c1}) = [F1, g'(G1)]_{c1}$ bzw. $[f', I_{\Gamma_1}]([F1, G1]_{c1}) = [f'(F1), G1]_{c1}$, da der entsprechende Minimalautomat streng zusammenhängend ist. Damit ist dann auch die Schlichtheit von $[f1, E_{\Gamma_1}]$ und $[E_{\Phi_1, g1}]$ auf $[F1, G1]_{c1}$ gezeigt. Es sei nochmals bemerkt, daß zu diesen Untersuchungen nur die Automaten der Abbildungen 23, 24 und 25 benötigt wurden.

Betrachtet man, wie zu Beginn des Kapitels erwähnt, $G1'$ an Stelle von $G1$ (Aktion bi entfernen), dann läßt sich nicht mehr die strikte Kooperativität von $[E_{\Phi}, I_{\Gamma_1}]([f'(F1), G1']_{c1})$ in $[f'(F1), G1']_{c1}$ bezüglich g' zeigen. In der Tat ist der Homomorphismus $v1$ nicht schlicht auf $[F1, G1']_{c1}$. Auch dies kann gezeigt werden, ohne eine Darstellung für $[F1, G1']_{c1}$ zu kennen.

Dazu sei bemerkt, daß $v1([F1, G1']_{c1})$ genau so wie $v1([F1, G1]_{c1})$ mit der kompositionellen Methode des ersten Kapitels bestimmt werden kann, und daß $v1([F1, G1']_{c1}) = v1([F1, G1]_{c1})$. Da der Minimalautomat von $v1([F1, G1']_{c1})$ streng zusammenhängend ist (siehe Abb. 6), ist der Homomorphismus $[E_{\{as, at\}}, I_{\{br, bf\}}]$ schlicht auf $v1([F1, G1']_{c1})$. Wäre $v1$ schlicht auf $[F1, G1']_{c1}$, dann wäre also auch $[E_{\{as, at\}}, I_{\{br, bf\}}] \circ v1$ schlicht auf $[F1, G1']_{c1}$. Wegen $[E_{\{as, at\}}, I_{\{br, bf\}}] \circ v1 = [E_{\Phi' \cup \Gamma_1 \setminus \{br, bf\}}, I_{\{br, bf\}}] \circ [f', I_{\Gamma_1}]$, müßte dann auch $[E_{\Phi' \cup \Gamma_1 \setminus \{br, bf\}}, I_{\{br, bf\}}]$ schlicht auf $[f', I_{\Gamma_1}]([F1, G1']_{c1}) = [f'(F1), G1']_{c1}$ sein. Entfernt man in Abb 24 die drei mit bi beschrifteten Kanten, dann erhält man den Minimalautomaten von $[f'(F1), G1']_{c1}$. An diesem Automaten ist unschwer zu erkennen (tote Zusammenhangskomponente in den Zuständen 11, 12, 13 und 14), daß die Deadlocksprache von $[f'(F1), G1']_{c1}$ bezüglich des Homomorphismus $[E_{\Phi' \cup \Gamma_1 \setminus \{br, bf\}}, I_{\{br, bf\}}]$ beispielsweise das leere Wort enthält, wohingegen die entsprechende Terminierungssprache leer ist. Daraus folgt aber, daß $[E_{\Phi' \cup \Gamma_1 \setminus \{br, bf\}}, I_{\{br, bf\}}]$ nicht schlicht auf $[f'(F1), G1']_{c1}$ und damit auch $v1$ nicht schlicht auf $[F1, G1']_{c1}$ ist.

5 Komposition der Kooperativität

Das im vorigen Kapitel demonstrierte kompositionelle Vorgehen bei wohlstrukturierten Systemspezifikationen läßt sich auch iterieren und zu Induktionsbeweisen ausbauen. Dazu wird noch eine Kompositionseigenschaft der Kooperativität (Satz 10 und 11) benötigt. Wie schon bemerkt, reicht für praktische Anwendungen in aller Regel der Begriff der strikten Kooperativität aus. Der Einfachheit wegen wird deshalb in diesem Papier eine Kompositionseigenschaft auch nur für diesen eingeschränkten Begriff angegeben. Für ihren Beweis wird noch ein Lemma sowie ein zusätzlicher Begriff benötigt.

Lemma 6:

Unter den Voraussetzungen des Satzes 4, der zusätzlichen Bedingung $g \langle \gamma$ und mit den Kooperationsformen $c = (\Phi, \Gamma, \Sigma, \phi, \gamma, C)$ sowie $d = (\Phi, \Gamma', \Sigma, \phi, \gamma', C)$ gilt $[I_{\Phi, g}(x^{-1}([F, G]_c))] \cap \Phi = [I_{\Phi, g}(x)^{-1}([F, g(G)]_d)] \cap \Phi$ für jedes $x \in [F, G]_c$.

Beweis:

Mit den Bemerkungen vor Lemma 2 und Satz 1 gilt

$$\begin{aligned} [I_{\Phi, g}(x^{-1}([F, G]_c))] &= [I_{\Phi, g}([I_{\Phi, E_{\Gamma}}(x)^{-1}(F), [E_{\Phi, I_{\Gamma}}(x)^{-1}(G)]_{cx})] = [[I_{\Phi, E_{\Gamma}}(x)^{-1}(F), g([E_{\Phi, I_{\Gamma}}(x)^{-1}(G)])]_{dx}, \\ \text{wobei } cx &= (\Phi, \Gamma, \Sigma, \phi, \gamma, [\phi, \gamma](x)^{-1}(C)) \text{ und } dx = (\Phi, \Gamma', \Sigma, \phi, \gamma', [\phi, \gamma](x)^{-1}(C)). \text{ Entsprechend gilt} \\ [I_{\Phi, g}(x)^{-1}([F, g(G)]_d)] &= [[I_{\Phi, E_{\Gamma'}}([I_{\Phi, g}(x)^{-1}(F), [E_{\Phi, I_{\Gamma'}}([I_{\Phi, g}(x)^{-1}(g(G))])]_{dx})]_{dx} = \\ &= [[I_{\Phi, E_{\Gamma}}(x)^{-1}(F), [E_{\Phi, g}(x)^{-1}(g(G))]]_{dx}. \text{ Aus der Definition des Kooperationsproduktes folgt dann} \\ [I_{\Phi, g}(x^{-1}([F, G]_c))] \cap \Phi &= [I_{\Phi, E_{\Gamma}}(x)^{-1}(F) \cap \Phi \cap [\phi, \gamma']^{-1}([\phi, \gamma](x)^{-1}(C))] = [I_{\Phi, g}(x)^{-1}([F, g(G)]_d)] \cap \Phi. \end{aligned}$$

Zur Formulierung von Satz 10 seien $f : \Phi^* \rightarrow \Phi'^*$, $g : \Gamma^* \rightarrow \Gamma'^*$ sowie $q : \Delta^* \rightarrow \Delta'^*$ Homomorphismen und $F \subset \Phi^*$ sowie $G \subset (\Gamma \cup \Delta \cup \Pi)^*$. Die Alphabete $\Phi, \Gamma, \Delta, \Pi, \Phi', \Gamma'$ und Δ' seien paarweise disjunkt mit Ausnahme der Paarungen (Φ, Φ') , (Γ, Γ') sowie (Δ, Δ') , und $\phi : \Phi^* \rightarrow \Sigma^*$, $\gamma : (\Gamma \cup \Delta \cup \Pi)^* \rightarrow \Sigma^*$, $\phi' : \Phi'^* \rightarrow \Sigma^*$ sowie $\gamma' : \Gamma'^* \rightarrow \Sigma^*$ seien Homomorphismen mit $\phi = \phi' \circ f$ und $\gamma = \gamma' \circ [g, E_{\Delta \cup \Pi}]$. Mit $C \subset \Sigma^*$ seien die Kooperationsformen $c = (\Phi, \Gamma \cup \Delta \cup \Pi, \Sigma, \phi, \gamma, C)$, $d = (\Phi, \Gamma', \Sigma, \phi, \gamma', C)$ und $e = (\Phi', \Gamma' \cup \Delta \cup \Pi, \Sigma, \phi', \gamma', C)$ definiert.

Der Homomorphismus $[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi}}]$ heißt *balanciert auf* $[[g, q], I_{\Phi' \cup \Pi}]]([f(F), G]_e)$, falls es zu jedem $w \in [[g, q], I_{\Phi' \cup \Pi}]]([f(F), G]_e)$ mit $[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi}}](w^{-1}([[g, q], I_{\Phi' \cup \Pi}]]([f(F), G]_e))) \neq \{\varepsilon\}$ eine Teilmenge $K \subset w^{-1}([[g, q], I_{\Phi' \cup \Pi}]]([f(F), G]_e))$ mit folgenden Eigenschaften gibt:

(1) $K \neq \emptyset$, $K = \text{pre}(\max(K))$ und K ist endlich.

(2) Für $u \in \max(K)$ gilt

$$[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi}}](u) \neq \varepsilon \text{ oder } [I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi}}](w^{-1}([[g, q], I_{\Phi' \cup \Pi}]]([f(F), G]_e))) = \{\varepsilon\}.$$

(3) Für $u \in K \setminus \max(K)$ gilt

$$(i) \quad u^{-1}(K) \cap \Pi = (w^{-1}([[g, q], I_{\Phi' \cup \Pi}]]([f(F), G]_e))) \cap \Pi \text{ sowie}$$

$$(ii) \quad u^{-1}(K) \cap (\Phi' \cup \Gamma' \cup \Delta') \neq \emptyset \text{ falls } (w^{-1}([[g, q], I_{\Phi' \cup \Pi}]]([f(F), G]_e))) \cap (\Phi' \cup \Gamma' \cup \Delta') \neq \emptyset.$$

Satz 10: Unter den obigen Voraussetzungen gilt:

Ist $[I_{\Phi, E_{\Gamma}}](F, [g, E_{\Delta \cup \Pi}](G))_d$ strikt kooperativ in $[F, [g, E_{\Delta \cup \Pi}](G)]_d$ bezüglich $f, [I_{\Gamma \cup \Delta}, E_{\Phi', \cup \Pi}](f(F), G)_e$ strikt kooperativ in $[f(F), G]_e$ bezüglich $[g, q]$ und ist $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}]$ balanciert auf $[[g, q], I_{\Phi', \cup \Pi}](f(F), G)_e$, dann ist $[I_{\Phi \cup \Gamma \cup \Delta}, E_{\Pi}](F, G)_c$ strikt kooperativ in $[F, G]_c$ bezüglich $[[f, g], q]$.

Beweis:

Im folgenden wird mehrfach die Assoziativität der direkten Summe von Homomorphismen sowie Satz 1 benutzt, ohne dies jeweils besonders zu erwähnen. Zur Abkürzung der Schreibweise sei $L = [[f, g], [q, I_{\Pi}]](F, G)_c$ und $M = [[f, g], E_{\Delta \cup \Pi}](F, G)_c = [I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](L)$. Sei $x \in [F, G]_c$, $y = [[f, g], [q, I_{\Pi}]](x)$ und $z = [[f, g], E_{\Delta \cup \Pi}](x) = [I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](y)$. Laut Voraussetzung gibt es dann Teilmengen $H1 \subset z^{-1}(M)$ und $H2 \subset y^{-1}(L)$, welche entsprechende strikte Kooperativitätsbedingungen (1) - (3) erfüllen.

Mit Hilfe von H1 und H2 wird jetzt eine Teilmengen $H \subset H2$ bestimmt, welche die Kooperativität von $[I_{\Phi \cup \Gamma \cup \Delta}, E_{\Pi}](F, G)_c$ in $[F, G]_c$ bezüglich $[[f, g], q]$ nachweist. Zur Konstruktion von H sind folgende zwei Beobachtungen wichtig: $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](y^{-1}(L)) \cap \Gamma' \subset z^{-1}(M) \cap \Gamma'$ und $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](y^{-1}(L)) \cap \Phi' = z^{-1}(M) \cap \Phi'$. Die Inklusion folgt aus $M = [I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](L)$, und die Gleichung ist eine Konsequenz von Lemma 6, wenn man beachtet, daß wegen $L = [f(F), [g, [q, I_{\Pi}]]G]_c$ mit $[I_{\Gamma}, E_{\Delta', \cup \Pi}]$ an Stelle von g die entsprechenden Voraussetzungen des Lemmas erfüllt sind; $c' = (\Phi', \Gamma' \cup \Delta' \cup \Pi, \Sigma, \phi', \gamma' \circ [I_{\Gamma}, E_{\Delta', \cup \Pi}], C)$ Außerdem sei noch bemerkt, daß die für H zu zeigende Eigenschaft (3) schwächer als die entsprechende Eigenschaft (3) von H2 ist. Aus diesen Überlegungen folgt jetzt, daß gemäß der Bemerkung vor Lemma 4 H1 derart (erweitert) gewählt werden kann, daß $H = [I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}]^{-1}(H1) \cap H2$ die Eigenschaften (1) und (3) der zu zeigenden strikten Kooperativitätsbedingung besitzt und daß $\max(H) \subset \max(H2)$.

Zum Beweis von Teil (2) der Kooperativitätsbedingung für H wird noch eine Eigenschaft von $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](u)$ für $u \in \max(H)$ benötigt. Es sei bemerkt, daß $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](u)$ für $u \in \max(H)$ nicht maximal in H1 sein muß. Da aber $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}]$ balanciert auf L ist, kann gemäß der Bemerkung vor Lemma 4 H2 derart (erweitert) gewählt werden, daß weiterhin $\max(H) \subset \max(H2)$, und zusätzlich für $u \in \max(H)$ $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](u) \in \max(H1)$ oder $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](y^{-1}(L)) = \{\varepsilon\}$ und $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](u) \in H1$.

Für die so konstruierte Menge H kann jetzt auch Teil (2) der Kooperativitätsbedingung gezeigt werden. Sei also $u \in \max(H)$. Es genügt dann $(yu)^{-1}(L) \subset u^{-1}([f, g], [q, I_{\Pi}]](x^{-1}([F, G]_c))$ zu beweisen. Da auch $u \in \max(H2)$, gilt wegen der entsprechenden Eigenschaft (2) $(yu)^{-1}(L) \subset u^{-1}([I_{\Phi', \cup \Pi}, [g, q]](f, I_{\Gamma \cup \Delta \cup \Pi}](x)^{-1}([f(F), G]_e)))$. Ebenso folgt, falls $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](y^{-1}(L)) \neq \{\varepsilon\}$, aus $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](u) \in \max(H1)$ die Inklusion $(z[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](u))^{-1}(M) \subset [I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](u)^{-1}([f, I_{\Gamma'}][I_{\Phi}, [g, E_{\Delta \cup \Pi}]](x)^{-1}([F, [g, E_{\Delta \cup \Pi}]](G)]_d))$.

Für $v \in (yu)^{-1}(L)$ folgt aus der ersten dieser zwei Inklusionen $uv \in [I_{\Phi', \cup \Pi}, [g, q]](f, I_{\Gamma \cup \Delta \cup \Pi}](x)^{-1}([f(F), G]_e)$. Wegen $[I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](v) \in [I_{\Phi', \cup \Gamma}, E_{\Delta', \cup \Pi}](y^{-1}(L)) \subset$

$[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}](yu)^{-1}([I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}](L)) = (z[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}](u))^{-1}(M)$ folgt aus der zweiten Inklusion $[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}](uv) \in [f, I_{\Gamma'}]([I_{\Phi, [g, E_{\Delta \cup \Pi}}]](x)^{-1}([F, [g, E_{\Delta \cup \Pi}]](G)]_d)$, falls $[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}]((yu)^{-1}(L)) \neq \{\varepsilon\}$. Ist $[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}]((yu)^{-1}(L)) = \{\varepsilon\}$, dann gilt dies ebenfalls, da $[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}](uv) = [I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}](u) \in H_1 \subset [f, I_{\Gamma'}]([I_{\Phi, [g, E_{\Delta \cup \Pi}}]](x)^{-1}([F, [g, E_{\Delta \cup \Pi}]](G)]_d)$.

Damit existieren $s \in [f, I_{\Gamma \cup \Delta \cup \Pi}](x)^{-1}([f(F), G]_e)$ und $t \in [I_{\Phi, [g, E_{\Delta \cup \Pi}}]](x)^{-1}([F, [g, E_{\Delta \cup \Pi}]](G)]_d)$ mit $uv = [I_{\Phi' \cup \Pi, [g, q]}](s)$ und $[I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}](uv) = [f, I_{\Gamma'}](t)$. Wegen $[f, I_{\Gamma \cup \Delta \cup \Pi}](x)s \in [f(F), G]_e = [f, I_{\Gamma \cup \Delta \cup \Pi}](F, G]_e$ und $[I_{\Phi, [g, E_{\Delta \cup \Pi}}]](x)t \in [F, [g, E_{\Delta \cup \Pi}]](G)]_d = [I_{\Phi, [g, E_{\Delta \cup \Pi}}]](F, G]_d$ existieren $r, w \in [F, G]_c$ mit $[f, I_{\Gamma \cup \Delta \cup \Pi}](x)s = [f, I_{\Gamma \cup \Delta \cup \Pi}](r)$ und $[I_{\Phi, [g, E_{\Delta \cup \Pi}}]](x)t = [I_{\Phi, [g, E_{\Delta \cup \Pi}}]](w)$. Damit gibt es Zerlegungen $r = r' r^\circ$ und $w = w' w^\circ$ mit $[f, I_{\Gamma \cup \Delta \cup \Pi}](x) = [f, I_{\Gamma \cup \Delta \cup \Pi}](r')$, $s = [f, I_{\Gamma \cup \Delta \cup \Pi}](r^\circ)$, $[I_{\Phi, [g, E_{\Delta \cup \Pi}}]](x) = [I_{\Phi, [g, E_{\Delta \cup \Pi}}]](w')$ und $t = [I_{\Phi, [g, E_{\Delta \cup \Pi}}]](w^\circ)$. Beachtet man noch, daß $[f, [g, E_{\Delta \cup \Pi}]](r^\circ) = [I_{\Phi, [g, E_{\Delta \cup \Pi}}]](s) = [I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}]([I_{\Phi' \cup \Pi, [g, q]}](s)) = [I_{\Phi' \cup \Gamma', E_{\Delta' \cup \Pi'}}](uv) = [f, I_{\Gamma'}](t) = [f, I_{\Gamma'}]([I_{\Phi, [g, E_{\Delta \cup \Pi}}]](w^\circ)) = [f, [g, E_{\Delta \cup \Pi}]](w^\circ)$, dann folgt mit Lemma 5 die Existenz eines $p \in x^{-1}([F, G]_c)$ mit $[f, I_{\Gamma \cup \Delta \cup \Pi}](p) = [f, I_{\Gamma \cup \Delta \cup \Pi}](r^\circ) = s$. Daraus folgt $uv = [I_{\Phi' \cup \Pi, [g, q]}](s) = [I_{\Phi' \cup \Pi, [g, q]}]([f, I_{\Gamma \cup \Delta \cup \Pi}](p)) = [[f, g], [q, I_{\Pi}]](p) \in [[f, g], [q, I_{\Pi}]](x^{-1}([F, G]_c))$, also $v \in u^{-1}([[f, g], [q, I_{\Pi}]](x^{-1}([F, G]_c)))$. Damit ist die gewünschte Inklusion gezeigt und Satz 10 vollständig bewiesen.

Satz 11:

Sei $\Phi \cap \Gamma = \Phi' \cap \Gamma = \Delta \cap \Gamma = \emptyset$, $F \subset \Phi^*$, $G \subset \Gamma^*$ und $C \subset \Sigma^*$. Seien $f : \Phi^* \rightarrow \Phi'^*$, $h : \Phi'^* \rightarrow \Delta^*$, $\phi : \Delta^* \rightarrow \Sigma^*$ und $\gamma : \Gamma^* \rightarrow \Sigma^*$ Homomorphismen. Die Kooperationsformen c und d seien durch $c = (\Phi, \Gamma, \Sigma, \phi \circ h \circ f, \gamma, C)$ und $d = (\Phi', \Gamma, \Sigma, \phi \circ h, \gamma, C)$ definiert. Ist $[I_{\Phi, E_{\Gamma}}]([F, G]_c)$ strikt kooperativ in $[F, G]_c$ bezüglich f und ist $[I_{\Phi', E_{\Gamma}}]([f(F), G]_d)$ strikt kooperativ in $[f(F), G]_d$ bezüglich h , dann ist auch $[I_{\Phi, E_{\Gamma}}]([F, G]_c)$ strikt kooperativ in $[F, G]_c$ bezüglich $h \circ f$.

Beweis:

Nach Voraussetzung gibt es zu jedem $x \in [F, G]_c$ eine Teilmenge $H \subset [f, I_{\Gamma'}](x)^{-1}([f(F), G]_d)$ und zu jedem $u \in [f, I_{\Gamma'}](x)^{-1}([f(F), G]_d)$ eine Teilmenge $H_u \subset [h, I_{\Gamma'}]([f, I_{\Gamma'}](x)u)^{-1}([h, I_{\Gamma'}]([f(F), G]_d))$, welche jeweils die entsprechenden strikten Kooperativitätseigenschaften (1) - (3) besitzt. Ist $[h, I_{\Gamma'}]([f, I_{\Gamma'}](x)u)^{-1}([h, I_{\Gamma'}]([f(F), G]_d)) \neq \{\varepsilon\}$, dann kann (ggf. durch entsprechende Erweiterung) auch $H_u \neq \{\varepsilon\}$ gewählt werden. Die Menge $H^\circ = [h, I_{\Gamma'}](H) \cup \bigcup_{u \in \max(H)} [h, I_{\Gamma'}](u)H_u$ ist nicht leer, endlich und präfixstabil. Wegen Lemma 6 und der speziellen Wahl der H_u besitzt H° auch die Kooperativitätseigenschaft (3) bezüglich $h \circ f$.

Die maximalen Elemente in H° besitzen eine Darstellung $[h, I_{\Gamma'}](u)v$, wobei u maximal in H und v maximal in H_u ist. Zum Nachweis der Kooperativitätseigenschaft (2) von H° genügt es, $([h, I_{\Gamma'}](u)v)^{-1}([h \circ f, I_{\Gamma'}](x^{-1}([F, G]_c))) \supset ([h, I_{\Gamma'}]([f, I_{\Gamma'}](x)u)v)^{-1}([h \circ f, I_{\Gamma'}]([F, G]_c))$ zu zeigen. Wegen der entsprechenden Eigenschaft (2) von H bzw. H_u gilt $([h, I_{\Gamma'}]([f, I_{\Gamma'}](x)u)v)^{-1}([h \circ f, I_{\Gamma'}]([F, G]_c)) = v^{-1}([h, I_{\Gamma'}]([f, I_{\Gamma'}](x)u)^{-1}([f(F), G]_d)) = v^{-1}([h, I_{\Gamma'}](u^{-1}([f, I_{\Gamma'}](x^{-1}([F, G]_c)))) \subset$

$v^{-1}([h, I_{\Gamma}](u)^{-1}([h \circ f, I_{\Gamma}](x^{-1}([F, G]_{\mathcal{C}})))) = ([h, I_{\Gamma}](u)v)^{-1}([h \circ f, I_{\Gamma}](x^{-1}([F, G]_{\mathcal{C}})))$, womit die gewünschte Inklusion gezeigt ist.

Am Ende des ersten Kapitels wurde ein Induktionsbeweis vorgestellt, der für das System in Abb. 14 $(v1 \circ [I_{\Phi \cup \Gamma_1}, E_{\Pi_1 \cup \dots \cup \Pi_n}])([...[[F1, P1]_{d1}, P2]_{d2}, \dots, Pn]_{dn}, G1]_{en}) = v([F, G]_{\mathcal{C}})$ zeigt, wobei für $k \geq 1$ $\Pi_k = \{ck_n, cki, ck?, ck+, ck-, ck!, ck?!, ck?+, ck?-, ck?! \}$ das Alphabet von P_k ist und zusätzlich entsprechende Kooperationsformen dk und ek definiert sind. Mit den Sätzen dieses letzten Kapitels läßt sich nun auch die Schlichtheit von $v1 \circ [I_{\Phi \cup \Gamma_1}, E_{\Pi_1 \cup \dots \cup \Pi_n}]$ auf $[...[[F1, P1]_{d1}, P2]_{d2}, \dots, Pn]_{dn}, G1]_{en}$ zeigen. Dies geschieht durch mehrere Induktionsbeweise. Wir knüpfen dazu an die Bezeichnungen an, welche im vorigen Kapitel zum Nachweis der Schlichtheit von $v1$ auf $[F1, G1]_{\mathcal{C}_1}$ eingeführt wurden. Zur Vereinfachung der Schreibweise benutzen wir die Abkürzung $S_n = [...[[F1, P1]_{d1}, P2]_{d2}, \dots, Pn]_{dn}$ und $S_0 = F1$. Für $k \geq 1$ seien die Homomorphismen $fk' : \Pi_k^* \rightarrow \Phi^*$ durch $fk'(ck?) = a?$, $fk'(ck?-) = a-$, $fk'(ck?+) = a+$, $fk'(ck?!) = a!$ und $fk'(x) = \varepsilon$ für $x \in \{ck_n, cki, ck?, ck+, ck-, ck!\}$ definiert. Für $k \geq 1$ sei $fk^{\wedge} = fk' \circ [E_{\Phi \cup \Pi_1 \cup \dots \cup \Pi_{k-1}}, I_{\Pi_k}]$, und $f0^{\wedge} = f'$. Mit vollständiger Induktion kann jetzt $fn^{\wedge}(S_n) = f'(F1)$ für alle $n \geq 0$ gezeigt werden. Die Induktionsverankerung folgt direkt; für den Induktionsschluß wird der in Abb. 26 dargestellte Minimalautomat von $[f'(F1), P_k]_{dk'}$ benötigt, wobei dk' eine entsprechende Kooperationsform ist. An diesem Automaten läßt sich leicht $fk'([E_{\Phi}, I_{\Pi_k}](f'(F1), P_k]_{dk'})) = f'(F1)$ nachweisen. Mit Korollar 2 ergibt sich daraus $fn+1^{\wedge}(S_{n+1}) = fn+1'([E_{\Phi \cup \Pi_1 \cup \dots \cup \Pi_n}, I_{\Pi_{n+1}}](S_n, P_{n+1}]_{dn+1})) = fn+1'([E_{\Phi}, I_{\Pi_{n+1}}](fn^{\wedge}(S_n), P_{n+1}]_{dn+1'})) = fn+1'([E_{\Phi}, I_{\Pi_{n+1}}](f'(F1), P_{n+1}]_{dn+1'})) = f'(F1)$.

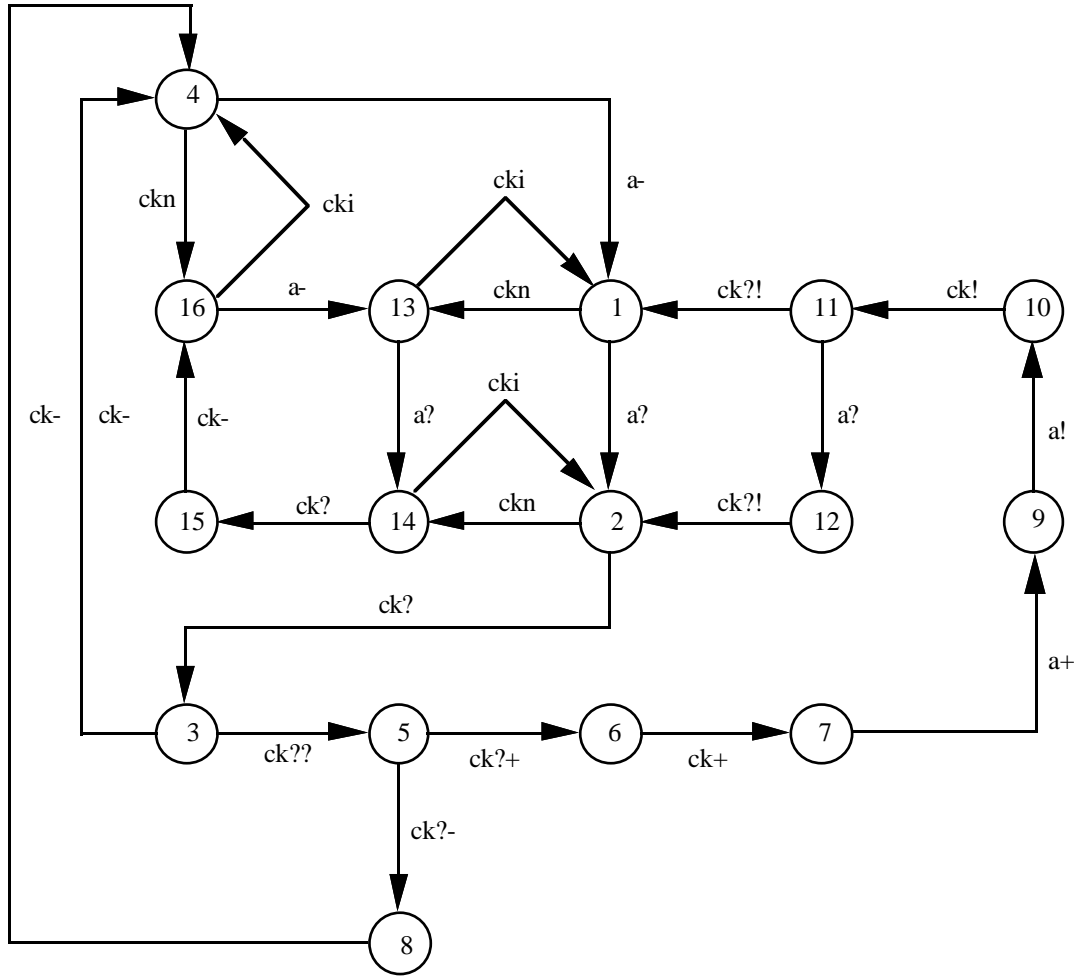


Abb. 26

Mit diesem Ergebnis sowie den Sätzen 10 und 11 wird jetzt mit vollständiger Induktion gezeigt, daß für $n \geq 0$ $[[\Phi_{1 \cup \Pi_1 \cup \dots \cup \Pi_n}, E_{\Gamma'}]([S_n, g'(G1)]_{en})$ strikt kooperativ in $[S_n, g'(G1)]_{en}$ bezüglich fn^{\wedge} ist. Die Induktionsverankerung wurde bereits im vorhergehenden Kapitel gezeigt.

Für den Induktionsschritt werden die in Abb. 27, 28, 29 und 23 dargestellten Minimalautomaten der Sprachen $[f'(F1), [Pk, g'(G1)]_{ek}]_{dk\sim}$, $[[\Phi' \cup \Gamma', [gk', qk]]([f'(F1), [Pk, g'(G1)]_{ek}]_{dk\sim})$, $[gk', E_{\Delta k \cup \Gamma'}]([Pk, g'(G1)]_{ek})$ und $[[E_{\Phi' \cup \Gamma_k}, hk], I_{\Gamma'}]([f'(F1), [Pk, g'(G1)]_{ek}]_{dk\sim})$ benötigt, wobei $dk\sim$ eine entsprechende Kooperationsform ist.

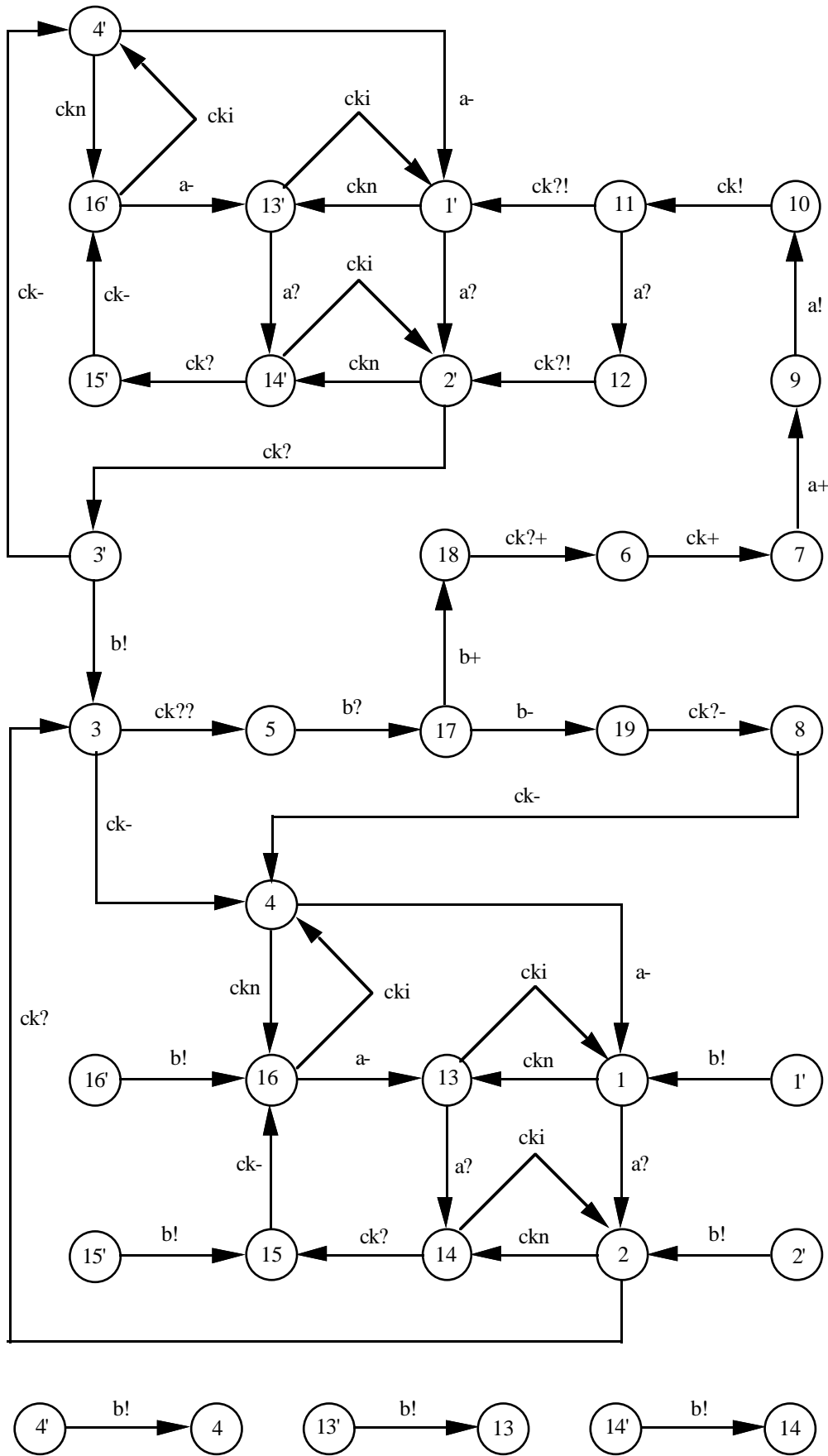


Abb. 27

$[I_{\Phi' \cup \Gamma_{n+1} \cup \Delta_{n+1}, E_{\Gamma'}}]([fn^{\wedge}, gn+1', qn+1, I_{\Gamma'}]([Sn+1, g'(G1)]_{en+1}))$ strikt kooperativ in $[fn^{\wedge}, gn+1', qn+1, E_{\Gamma'}]([Sn+1, g'(G1)]_{en+1})$ bezüglich $[E_{\Phi' \cup \Gamma_{n+1}, hn+1'}]$ ist. Aus Satz 11 folgt dann, daß $[I_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_{n+1}, E_{\Gamma'}}]([Sn+1, g'(G1)]_{en+1})$ strikt kooperativ in $[Sn+1, g'(G1)]_{en+1}$ bezüglich $[E_{\Phi' \cup \Gamma_{n+1}, hn+1'}] \circ [fn^{\wedge}, gn+1', qn+1] = fn+1^{\wedge}$ ist, was den Induktionsschritt vervollständigt.

Im vorhergehenden Kapitel wurde bereits gezeigt, daß $[E_{\Phi', I_{\Gamma_1}}]([f'(F1), G1]_{c^o})$ strikt kooperativ in $[f'(F1), G1]_{c^o}$ bezüglich g' ist. Daraus folgt jetzt mit dem obigen Ergebnis aus Satz 9, daß für $n \geq 0$ $[fn^{\wedge}, g']$ schlicht auf $[Sn, G1]_{en}$ ist. Wegen $fn^{\wedge}(Sn) = f'(F1)$ gilt mit Satz 1 $[fn^{\wedge}, g']([Sn, G1]_{en}) = [f'(F1), g'(G1)]_{c^o}$. Da der Minimalautomat von $[f'(F1), g'(G1)]_{c^o}$ streng zusammenhängend ist (Abb. 23), sind die Homomorphismen $[I_{\Phi', E_{\Gamma'}}]$ und $[E_{\Phi', I_{\Gamma_1}}]$ schlicht auf $[f'(F1), g'(G1)]_{c^o}$. Damit sind dann aber auch $[fn^{\wedge}, E_{\Gamma_1}] = [I_{\Phi', E_{\Gamma'}}] \circ [fn^{\wedge}, g']$ und $[E_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, g'}] = [E_{\Phi', I_{\Gamma_1}}] \circ [fn^{\wedge}, g']$ schlicht auf $[Sn, G1]_{en}$. Aus Satz 8 folgt daraus die Schlichtheit von $[I_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, E_{\Gamma_1}}]$ und von $[E_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, I_{\Gamma_1}}]$ auf $[Sn, G1]_{en}$.

Mit Hilfe von Abb. 23 läßt sich $[I_{\Phi', E_{\Gamma'}}]([f'(F1), g'(G1)]_{c^o}) = f'(F1)$ zeigen. Aus Korollar 2 und Satz 4 folgt dann $[I_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, E_{\Gamma_1}}]([Sn, G1]_{en}) = [I_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, E_{\Gamma'}}]([Sn, g'(G1)]_{en}) = fn^{\wedge^{-1}}([I_{\Phi', E_{\Gamma'}}]([fn^{\wedge}(Sn), g'(G1)]_{c^o})) \cap Sn = fn^{\wedge^{-1}}(f'(F1)) \cap Sn = Sn$ wegen $fn^{\wedge}(Sn) = f'(F1)$. Entsprechend läßt sich $[E_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, I_{\Gamma_1}}]([Sn, G1]_{en}) = G1$ zeigen.

Zum Nachweis der Schlichtheit von $v1 \circ [I_{\Phi_1 \cup \Gamma_1, E_{\Pi_1 \cup \dots \cup \Pi_n}}]$ auf $[Sn, G1]_{en}$ seien für $n \geq 0$ die Homomorphismen $hn : (\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n)^* \rightarrow \Phi^*$ durch $hn = [[I_{\{as, at\}, E_{\Phi_1 \setminus \{as, at\}}}, [E_{\Pi_1 \cup \dots \cup \Pi_{n-1}, fn'}]]$ für $n > 0$ und $h0 = f1$ definiert. Mit dem Homomorphismen $g1 : \Gamma_1^* \rightarrow \Gamma^*$ und $v : (\Phi \cup \Gamma)^* \rightarrow \Xi^*$ (siehe Kapitel 1) gilt jetzt $v1 \circ [I_{\Phi_1 \cup \Gamma_1, E_{\Pi_1 \cup \dots \cup \Pi_n}}] = v \circ [hn, g1]$.

Da der Minimalautomat von $G1$ streng zusammenhängend ist (Abb. 8), ist $g1$ schlicht auf $G1$ und damit auch $[E_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, g1}] = g1 \circ [E_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, I_{\Gamma_1}}]$ schlicht auf $[Sn, G1]_{en}$. Zum Nachweis der Schlichtheit von $[hn, E_{\Gamma_1}] = hn \circ [I_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, E_{\Gamma_1}}]$ auf $[Sn, G1]_{en}$ muß noch die Schlichtheit von hn auf Sn gezeigt werden.

Da der Minimalautomat von $S0 = F1$ streng zusammenhängend ist (Abb. 7), ist $h0$ schlicht auf $S0$; außerdem gilt $h0(S0) = f1(F1) = F$. Als Induktionsvoraussetzung sei jetzt hn schlicht auf Sn und $hn(Sn) = F$. Für $k \geq 1$ seien die Homomorphismen $hk^{\wedge} : \Pi k^* \rightarrow \Gamma^*$ durch $hk^{\wedge}(ck?) = b?$, $hk^{\wedge}(ck-) = b-$, $hk^{\wedge}(ck+) = b+$, $hk^{\wedge}(ck!) = b!$ und $hk^{\wedge}(x) = \varepsilon$ für $x \in \{ckn, cki, ck?+, ck?-, ck?! \}$ definiert. Es gilt dann $hk^{\wedge}(Pk) = g'(G1)$.

Mit Hilfe von Abb. 26 und 23 läßt sich zeigen, daß $[E_{\Phi', I_{\Pi_{n+1}}}]([f'(F1), Pn+1]_{dn+1})$ strikt kooperativ in $[f'(F1), Pn+1]_{dn+1}$ bezüglich $hn+1^{\wedge}$ ist. Es sei daran erinnert, daß oben schon die strikte Kooperativität von $[I_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, E_{\Gamma'}}]([Sn, g'(G1)]_{en})$ in $[Sn, g'(G1)]_{en}$ bezüglich fn^{\wedge} für alle $n \geq 0$ gezeigt wurde. Wegen $fn^{\wedge}(Sn) = f'(F1)$ folgt jetzt aus Satz 9, daß $[fn^{\wedge}, hn+1^{\wedge}]$ schlicht auf $[Sn, Pn+1]_{dn+1}$ ist. Wie oben folgt daraus die Schlichtheit von $[I_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, E_{\Pi_{n+1}}}]$ und von $[E_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, I_{\Pi_{n+1}}}]$ auf $[Sn, Pn+1]_{dn+1}$. Ebenfalls wie oben läßt sich auch $[I_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, E_{\Pi_{n+1}}}]([Sn, Pn+1]_{dn+1}) = Sn$ und $[E_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n, I_{\Pi_{n+1}}}]([Sn, Pn+1]_{dn+1}) = Pn+1$ zeigen.

Da der Minimalautomat von $Pn+1$ streng zusammenhängend ist (Abb. 11), ist der Homomorphismus $[I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}, E_{\{cn+1i, cn+1n\}}}]$ schlicht auf $Pn+1$. Zusammen mit der Induktionsvoraussetzung und

dem oben gezeigt ist dann $[hn, E_{\Pi_{n+1}}]$ sowie $[E_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n}, [I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}}, E_{\{cn+1i, cn+1n\}}]]$ schlicht auf $[Sn, P_{n+1}]_{dn+1}$, woraus mit Satz 9 die Schlichtheit von $[hn, [I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}}, E_{\{cn+1i, cn+1n\}}]]$ auf $[Sn, P_{n+1}]_{dn+1} = Sn+1$ folgt.

Nach Satz 1 gilt jetzt $[hn, [I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}}, E_{\{cn+1i, cn+1n\}}]](Sn+1) = [hn(Sn), [I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}}, E_{\{cn+1i, cn+1n\}}]](P_{n+1})_{dn+1} = [F, [I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}}, E_{\{cn+1i, cn+1n\}}]](P_{n+1})_{dn+1}^\circ$, wobei $dn+1^\circ$ eine entsprechende Kooperationsform ist. Bis auf Isomorphie ist der Minimalautomat von $[I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}}, E_{\{cn+1i, cn+1n\}}]](P_{n+1})$ bzw. von $[F, [I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}}, E_{\{cn+1i, cn+1n\}}]](P_{n+1})_{dn+1}^\circ$ gleich dem in Abb. 12 bzw. 13 dargestellten Automaten; der entsprechende Isomorphismus besteht darin, in den Kantenanschriften den Buchstaben c durch die Zeichenkette cn+1 auszutauschen.

Da der Automat in Abb. 13 streng zusammenhängend ist, folgt die Schlichtheit von $hn+1$ auf $Sn+1$ aus $hn+1(Sn+1) = ([I_{\{as, at\}}, E_{\Phi \setminus \{as, at\}}], [E_{\Gamma_{n+1}}, hn+1']) \circ [hn, [I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}}, E_{\{cn+1i, cn+1n\}}]](Sn+1)$. An diesem Automaten kann auch leicht $hn+1(Sn+1) = [I_{\{as, at\}}, E_{\Phi \setminus \{as, at\}}], [E_{\Gamma_{n+1}}, hn+1'])([F, [I_{\Pi_{n+1} \setminus \{cn+1i, cn+1n\}}, E_{\{cn+1i, cn+1n\}}]](P_{n+1})_{dn+1}^\circ) = F$ nachgeprüft werden.

Insgesamt folgt jetzt mit Satz 9 aus der Schlichtheit von $[hn, E_{\Gamma_1}]$ und $[E_{\Phi_1 \cup \Pi_1 \cup \dots \cup \Pi_n}, g_1]$ auf $[Sn, G_1]_{en}$ die Schlichtheit von $[hn, g_1]$ auf $[Sn, G_1]_{en}$. Nach Satz 1 gilt $[hn, g_1]([Sn, G_1]_{en}) = [hn(Sn), g_1(G_1)]_c = [F, G]_c$. Da der Minimalautomat von $[F, G]_c$ streng zusammenhängend ist (Abb. 5), ist der Homomorphismus v schlicht auf $[F, G]_c$ und damit auch $v \circ [I_{\Phi_1 \cup \Gamma_1}, E_{\Pi_1 \cup \dots \cup \Pi_n}] = v \circ [hn, g_1]$ schlicht auf $[Sn, G_1]_{en}$. Außerdem gilt noch $(v \circ [I_{\Phi_1 \cup \Gamma_1}, E_{\Pi_1 \cup \dots \cup \Pi_n}])([Sn, G_1]_{en}) = v([F, G]_c)$, was die Verifikation unseres parametrisierten Server-Resource-Beispiels vervollständigt.

Schlußbemerkung

Aus dem Gebiet der Prozessalgebren stammt ein Kopplungsmechanismus für LTS - Rendezvous genannt - welcher durch das "gleichzeitige Ausführen" einander zugeordneter Aktionen der beteiligten LTS definiert ist. Da hierbei die Aktionsfolgen des zusammengesetzten LTS keine Elemente des Shuffleproduktes der Aktionsfolgen der einzelnen LTS sind, läßt sich ein solches Rendezvous auch nicht in direkter Weise als Kooperationsprodukt darstellen. Mit Hilfe sogenannter Handshakes können aber Rendezvous derart durch Kooperationsprodukte simuliert werden, daß sie durch Bijektionen und schlichte Homomorphismen in Beziehung stehen. Dadurch lassen sich die hier bewiesenen Sätze über Kooperationsprodukte auf Rendezvous von LTS übertragen.

Zur Nutzung der hier vorgestellten Verifikationsmethode für komplexe Anwendungen existiert ein Werkzeug: das SH-Verification Tool /NOR96/. Zur Zeit wird darin die Systemspezifikation, welche die Menge der Aktionsfolgen des Systems festlegt, noch in Form von Produktnetzen /OP95/, einer Klasse höherer Petrinetze, unterstützt. Mit der Realisierung einer allgemeinen APA-Oberfläche wurde bereits begonnen.

Literatur

- /AS85/ B. Alpern, F.B. Schneider:
Defining Liveness
Information Processing Letters 24 1985
- /AH95/ R. Alur, T.A. Henzinger:
Local Liveness for Compositional Modeling of Fair Reactive Systems
Computer Aided Verification (CAV) '95 Liège 1995 LNCS 939
- /Arn94/ A. Arnold:
Finite Transition Systems
Prentice Hall 1994
- /BW90/ J.C.M. Baeten, W.P. Weijlan:
Process Algebra
Cambridge University Press 1990
- /BD87/ S. Budkowski, P. Dembinski:
An Introduction to Estelle
Computer Networks 14 1987
- /C&96a/ C. Capellmann, R. Demant, R. Galvez-Estrada, U. Nitsche, P. Ochsenschläger:
Case Study: Service Interaction Detection by Formal Verification under Behaviour Abstraction
International Workshop INTELLIGENT NETWORKS Passau 1996
- /C&96b/ C. Capellmann, R. Demant, F. Fatahi-Vanani, R. Galvez-Estrada, U. Nitsche, P. Ochsenschläger:
Verification by behavior abstraction:
A case study of service interaction detection in intelligent telephone networks
Computer Aided Verification (CAV) '96 New Brunswick 1996 LNCS 1102
- /CES86/ E.M. Clarke, E.A. Emerson, A.P. Sistla:
Automatic Verification of Finite-State Concurrent Systems using Temporal Logic Specifications
ACM TOPLAS 8 1986
- /Eil74/ S. Eilenberg:
Automata, Languages and Machines Vol. A
Academic Press 1974
- /Kur93/ R.P. Kurshan:
Automata-Theoretic Verification of Coordinating Processes
Princeton Univ. Press 1993
- /Mil90/ R. Milner:
Operational and Algebraic Semantics of Concurrent Processes
Handbook of Theoretical Computer Science Vol. B Elsevier 1990
- /Neb94/ M. Nebel:
Ein Produktnetz zur Verifikation von SmartCard-Anwendungen in der STARCOS-Umgebung
GMD-Studien 234 1994
- /Nit96a/ U. Nitsche:
Verification and behavior abstraction
towards a tractable verification technique for large distributed systems
Journal of Systems and Software
Special Issue on Software Engineering for Distributed Computing 1996

- /Nit96b/ U. Nitsche:
Verification of Co-Operating Systems and Behaviour Abstraction
als Dissertation im Fachbereich Informatik der Universität Frankfurt 1996 eingereicht
- /NO95/ U. Nitsche, P. Ochsenschläger:
Approximately Satisfied Properties of Systems and Simple Language Homomorphisms
In S. Gnesi, D. Latella:
ERCIM Workshop: Formal Methods for Industrial Critical Applications Oxford 1996
(erscheint in Information Processing Letters)
- /NOR96/ U. Nitsche, P. Ochsenschläger, J. Repp:
The SH-Verification Tool.
Arbeitspapiere der GMD 1007 1996
- /Och92/ P. Ochsenschläger:
Verifikation kooperierender Systeme mittels schlichter Homomorphismen
Arbeitspapiere der GMD 688 1992
- /Och94a/ P. Ochsenschläger:
Verification of cooperating systems by simple homomorphisms using the Product Net Machine
In J. Desel, A. Oberweis, W. Reisig:
Workshop: Algorithmen und Werkzeuge für Petrinetze Humboldt Universität Berlin 1994
- /Och94b/ P. Ochsenschläger:
Kompositionelle Verifikation kooperierender Systeme
Arbeitspapiere der GMD 885 1994
- /Och94c/ P. Ochsenschläger:
Verifikation von SmartCard-Anwendungen mit Produktnetzen
4. GMD-SmartCard Workshop Darmstadt 1994
- /Och95/ P. Ochsenschläger:
Compositional verification of cooperating systems using simple homomorphisms
In J. Desel, H. Fleischhack, A. Oberweis, M. Sonnenschein:
Workshop: Algorithmen und Werkzeuge für Petrinetze. C.v.O. Universität Oldenburg 1995
- /OP95/ P. Ochsenschläger, R. Prinoth:
Modellierung verteilter Systeme
Konzeption, Formale Spezifikation und Verifikation mit Produktnetzen.
Vieweg Verlag 1995
- /SSR89/ R. Saracco, J. R. W. Smith, R. Reed:
Telecommunications Systems Engineering using SDL
North Holland 1989
- /Zie89/ W. Zielonka:
Safe Executions of Recognizable Trace Languages by Asynchronous Automata
Logical Foundations of Computer Science 1989 LNCS 363 Springer Verlag