

#### 4. Anwendungen bei der Automatischen Zeichenerkennung

Wie bereits in der Einleitung angedeutet wurde, liegen bei praktischen Erkennungsproblemen, wie beispielsweise in der Automatischen Zeichenerkennung /1 - 5/, in der Regel keine Einklassen-Probleme <sup>+)</sup>  vor, bei denen die Entscheidungen mit Hilfe von Detektoren getroffen werden, sondern Mehrklassen-Probleme, wobei die Entscheidungsorgane als Klassifikatoren bezeichnet werden. Als Klassifikatoren insbesondere in der Automatischen Zeichenerkennung sind sogenannte bereichszuordnende Klassifikatoren /1/ dann zweckmäßig, wenn bei Abbildung der zu klassifizierenden Menge von Mustern in einem n-dimensionalen Nachrichtenraum sogenannte kompakte Klassenbereiche entstehen /1, 3, 6 - 9/. Unter Kompaktheit der Klassen wird in diesem Zusammenhang verstanden, daß die Bildpunkte im Nachrichtenraum auf einfache Weise durch Hyperflächen klassenweise separierbare Anordnung aufweisen /8/.

Die Anwendung bereichszuordnender Klassifikatoren ist jedoch nicht immer zweckmäßig und nicht in jedem Falle möglich. Ist eine Kompaktheit der zu separierenden Klassen nicht gegeben, was u.a. bei manchen Problemen der Automatischen Bildauswertung zu erwarten ist, so ist ein bereichszuordnender Klassifikator meist zu aufwendig /16/, sowohl hinsichtlich der Realisierung, als auch bezüglich des Entwurfes. In diesem Fall können sogenannte punktzuordnende Klassifikatoren /1/ sinnvoll eingesetzt werden. Ist jedoch eine Abbildung der zu klassifizierenden Menge von Mustern in einen n-dimensionalen Raum nicht möglich infolge variabler Kriterien-Anzahl, wie beispielsweise häufig bei Listenverarbeitungsprozessen /20/, so ist auch die Anwendung punktzuordnender Klassifikatoren nicht möglich. Bei solchen Problemen werden meist entweder triviale Vergleichsalgorithmen oder Algorithmen mit Entscheidungsbaumstruktur (z.B. /20/) verwendet. Für die Lösung solcher Klassifikationsprobleme kann der im folgenden Kapitel vorgeschlagene *Moore-Klassifikator* eingesetzt werden. Der

---

<sup>+)</sup>  Die Rückweisungsklasse ist nicht mitgerechnet.

Moore-Klassifikator ist ein endlicher Automat, dessen Theorie in Kapitel 4.1 auf die automatentheoretische Modellvorstellung des Akzeptors zurückgeführt wird unter Angabe einer Synthese-Methode. Auf der Grundlage des Moore-Klassifikators kann ein soft ware - System zur Klassifikation mit in einfacher Weise automatisierter Synthese des Klassifikators konzipiert werden, wie in Abschnitt 5 beschrieben wird. Die automatische Synthese gestattet beispielsweise bei Computereinsatz mit Dialogverkehr über Bildschirm-Ein- und -Ausgabegeräte mit Lichtgriffel /10-14/ die Aufstellung eines für den jeweiligen Benutzer individuellen Zeichensatzes /20/ zwecks späterer graphischer Eingabe über automatische Erkennung.

Wie beim Klassifikator-Entwurf, so ist in der Automatischen Zeichenerkennung auch beim Entwurf von Schaltungen oder Algorithmen zur Informationsreduktion keine ernstzunehmende Anwendung der Automatentheorie bekannt. Es sind nur wenige Anwendungen meist einfacher elektrischer Schaltwerke bekannt, die auf heuristische Weise entworfen wurden /27,29-32/. Die Anwendung eines Zustandsgraphen bei der Funktionsbeschreibung ist hierbei meist die einzige Brücke zur Automatentheorie. Darüber hinaus ist die Schaltwerks-Anwendung bei diesen Arbeiten meist auf einen funktionellen Ausbau des Lichtpunkt-Abtasters beschränkt /27,29,30/. In Kapitel 4.2 wird auf der theoretischen Grundlage des Akzeptor-Modelles ein Synthese-Verfahren zum Entwurf von Transduktoren für Zwecke der Informationsreduktion vorgeschlagen. Mit Hilfe eines Simulationsprogrammes wurde eine ausschließlich mittels Transduktoren durchgeführte Informationsreduktion für die Erkennung handgeschriebener Ziffern demonstriert /47/. Es handelt sich hierbei um eine aus vier Transduktoren bestehende Kaskade zur Ermittlung von Kriterien aus dem äußeren Konturverlauf von Linienmustern. Der erste Transduktor dieser Kaskade, der von einer Abtastung des Musters nach der Zeilendifferenz-Methode (vgl. Bild 2.1.1) ausgeht, wird in Kapitel 4.2.1 beschrieben.

In den folgenden Kapiteln wird eine Reihe neuer Akzeptortypen eingeführt. Es ergibt sich der in Bild 4.1 gezeigte hierarchische Zusammenhang zwischen allgemeineren und spezielleren Typen. Zunächst wird ein *allgemeiner Akzeptor* definiert, der sich vom bisher bekannten und nun als *gewöhnlicher Akzeptor* bezeichneten Typ dadurch unterscheidet, daß die Menge der Endzustände in Klassen unterteilt ist. Die Anzahl der Klassen sei mit  $k$  bezeichnet. Der gewöhnliche Akzeptor ist hierbei ein Spezialfall mit  $k = 1$ . Ein Akzeptor mit  $k > 1$  sei als *klassifizierender Akzeptor* bezeichnet.

Die graphische Darstellung des klassifizierenden Akzeptors in Form eines Zustandsgraphen bereitet Schwierigkeiten, da mit  $k + 2$  Kategorien von Zuständen mehr als 3 Typen von Knotensymbolen unterschieden werden müßten. Der für gewöhnliche Akzeptoren vereinbarte Vorrat an Knotensymbolen (Quadrat, Kreis und Doppelkreis) reicht hierbei nicht aus. Aus diesem Grunde wird zur Darstellung von allgemeinen Akzeptoren die Darstellungsweise von Moore-Automaten eingeführt. Ein Akzeptor mit  $k = 1$  wird in dieser Darstellung als *Moore-Akzeptor* und ein Akzeptor mit  $k > 1$  als *Moore-Klassifikator* bezeichnet (Bild 4.1). Da jeder Moore-Automat in einen äquivalenten Mealy-Automaten umgewandelt werden kann (z.B. /25/), werden entsprechende Modellvorstellungen unter der Bezeichnung *Mealy-Akzeptor* und *Mealy-Klassifikator* definiert.

Auf der Grundlage dieser eingeführten Akzeptormodelle werden formale Methoden zur Synthese von Automaten zur Lösung von Klassifikationsproblemen und von Problemen der Informationsreduktion angegeben. Diese Synthesemethoden sind Modifikationen der in den vorausgehenden Kapiteln beschriebenen Synthesemethode auf der Grundlage regulärer Ausdrücke.

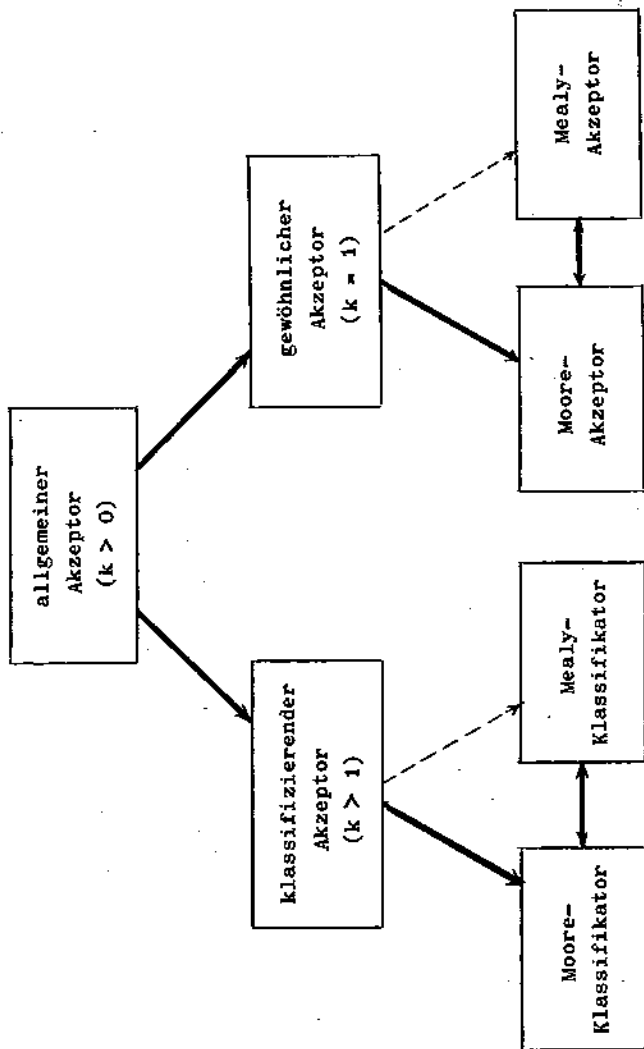


Bild 4.1 Hierarchie der Akzeptor-Typen (k = Anzahl der Endzustands-Klassen).

#### 4.1 Der Moore-Klassifikator und ein Verfahren zu seiner Synthese

Kapitel 2 und 3 der vorliegenden Arbeit basieren auf der abstrakten Modellvorstellung des *Akzeptors*, der auch unter der Bezeichnung *Detektor* oder *Erkennender Automat* bekannt ist /49/. Bild 4.1.1 a zeigt die black box-Darstellung des Akzeptors, der Eingabefolgen hinsichtlich ihrer Zugehörigkeit zu einer einzigen Klasse prüft. Diese Klasse ist gegeben durch das von diesem Akzeptor akzeptierte Ereignis  $\rho$ . Die Entscheidung des Akzeptors ist alternativ. Dies bedeutet, daß eine Eingabefolge einer einzigen gegebenen Klasse angehören kann oder nicht. Das Entscheidungskriterium ist hierbei die Untermenge  $Z'$  der Zustandsmenge  $Z$  gemäß Definition 2.2.3. Da in der Automatischen Zeichenerkennung Ein-Klassenprobleme jedoch nicht vorherrschend sind, wird untenstehend die Definition eines verallgemeinerten Akzeptors vorgeschlagen, mit dessen Hilfe Ein- und Mehr-Klassen-Probleme gelöst werden können. Dieser wie folgt definierte Akzeptor  $\mathcal{K}$  sei als *klassifizierender Akzeptor* bezeichnet.

Definition 4.1.1: Ein *klassifizierender Akzeptor*  $\mathcal{K}$  ist ein Quintupel  $\mathcal{K} = (X, Z, (z_0), \delta, Z_k^*)$ , wobei  $Z_k^*$  die Partition  $\Pi_k(Z') = \{Z_1^1, Z_2^1, Z_3^1, \dots, Z_i^1, \dots, Z_k^1\}$  ist.  $\mathcal{K}$  wird einfach genannt genau dann, wenn für alle  $i$  die Menge  $Z_i$  aus genau einem Zustand besteht, wobei Def. 2.2.6 sinngemäß eingehalten ist.

Definition 4.1.2: Ein *deterministischer klassifizierender Akzeptor* ist ein Akzeptor nach Definition 4.1.1, der nur einen einzigen Anfangszustand  $z_0$  besitzt, d.h. mit  $Z_0 = (z_0)$ .

Definition 4.1.3: Wenn ein Akzeptor  $\mathcal{K}$  das Ereignis  $\rho$  akzeptiert, ist  $\Pi_k(\rho) = \{\rho(i) \mid i = 1, 2, \dots, k\}$  die *Akzeptierungs-Partition* von  $\rho$ , wobei  $\rho(i) = \{s \mid \delta(z_0, s) \in Z_i^1\}$  ist, und  $i$  als Klasse bezeichnet wird.

Aus diesen Definitionen folgt für alle  $i$ , daß eine Folge  $s$  des Ereignis  $\rho(i) \subset \rho$  den klassifizierenden Akzeptor vom Anfangszustand  $z_0$  in einen Zustand  $z_f \in Z_i'$  überführt. Wird für jede Menge  $Z_i \subset Z_k''$  der Endzustände ein Indikator vorgesehen, so können mit dem klassifizierenden Akzeptor Eingabefolgen klassifiziert werden (Bild 4.1.1 b).

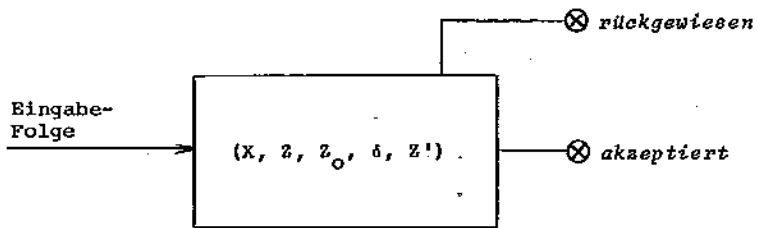
Der gewöhnliche Akzeptor gemäß Definition 2.2.3 ist ein Spezialfall des klassifizierenden Akzeptors mit der Klassen-Anzahl  $k = 1$ , weshalb  $Z_k'' = \Pi_k(Z') = \{Z_1'\}$  ist. Es liegt somit eine Verallgemeinerung des Akzeptormodelles vor.

Das Syntheseproblem des klassifizierenden Akzeptors ist lösbar für alle Ereignis-Mengen, deren Elemente Ereignisse sind, die sich paarweise nicht überschneiden. Im Falle von Überschneidungen ist das Problem wegen Widerspruches nicht lösbar.

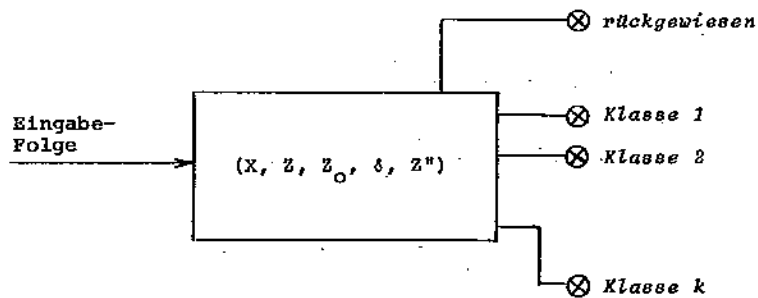
Ein gewöhnlicher oder ein klassifizierender Akzeptor kann als spezieller Moore-Automat aufgefaßt und dargestellt werden. Ein solcher spezieller Moore-Automat kann wie folgt definiert werden, wofür die Bezeichnung *Moore-Klassifikator* vorgeschlagen sei.

Definition 4.1.4: Ein *Moore-Klassifikator*  $\mathcal{R}$  ist ein 7-Tupel  $\mathcal{R} = (X, Y, Z, z_0, \delta, Z_k'', \lambda)$  mit den speziellen Alphabeten  $Y = \{0, y_1, y_2, \dots, y_n\}$  mit  $n = k$  (Klassen-Anzahl),  $Z_k'' = \Pi_k(Z')$ , und den Abbildungen  $\delta: X \times Z \rightarrow Z$ ,  $\lambda: Z_k'' \rightarrow Y$ , wobei  $y = 0$  die Bedeutung "kein Endzustand" hat. In Zustandsgraphen von Moore-Klassifikatoren werden Endzustände durch Doppelkreise bezeichnet, in welche Zustandsnummer und Ausgabe-Symbol eingetragen werden (Beispiele in Bild 4.1.2).

Bei dieser Definition ist gewährleistet, daß in diesem speziellen Fall alle Relationen und Transformationen, die in Ka-



a) Black box-Darstellung des gewöhnlichen Akzeptors



b) Black box-Darstellung des klassifizierenden Akzeptors

Bild 4.1.1 Gewöhnlicher und klassifizierender Akzeptor.

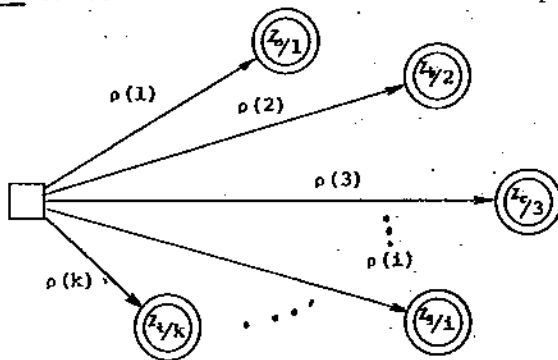


Bild 4.1.2 Uneigentlicher Zustandsgraph eines Moore-Klassifikators (2. Version einer Problemformulierung).

pitel 2 und 3 auf der Grundlage des gewöhnlichen Akzeptors beschrieben wurden, auch für die Darstellung nach Art eines Moore-Automaten gültig sind dadurch, daß beim so definierten Moore-Klassifikator nur solche Zustände ein Ausgabesignal haben, die im Sinne des Akzeptierungsbegriffes Endzustände sind. Von hier ab sei nicht mehr unterschieden zwischen klassifizierendem Akzeptor und Moore-Klassifikator und nur die letztere Bezeichnung benützt, da sich diese beiden Systeme in ihrer graphischen Darstellung prinzipiell nicht unterscheiden.

Satz 3.1.1 über die Äquivalenz von Zuständen muß für den Moore-Klassifikator modifiziert werden zur Berücksichtigung der Partition der Endzustände, woraus sich folgender Satz ergibt.

Satz 4.1.1: Zwei Zustände  $z_i$  und  $z_j$  eines Moore-Klassifikators  $\mathcal{R}$  sind äquivalent, wenn die Deskriptoren  $\rho_i$  und  $\rho_j$  der durch  $z_i$  und  $z_j$  akzeptierten Ereignisse gleich sind und gleichzeitig  $\lambda(\delta(z_i, \rho_i)) = \lambda(\delta(z_j, \rho_j))$  ist.

Für die Synthese eines Moore-Klassifikators wird folgendes Verfahren vorgeschlagen unter Anwendung der Transformationen für Graphen nach Kapitel 3 ff.

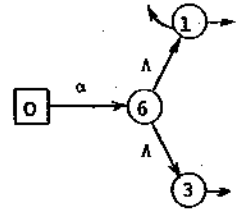
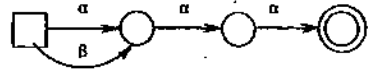
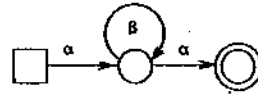
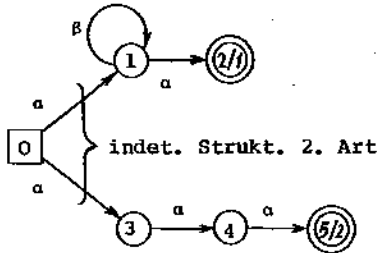
1. Man entwerfe für jedes Ereignis  $\rho(i)$  den Zustandsgraphen eines deterministischen Akzeptors mit  $\lambda(z_j) = y_i$  für alle  $z_j \in Z^n$  (nach dem Beispiel von Bild 4.1.3 a).
2. Man verschmelze alle Anfangsknoten dieser Graphen zu einem einzigen Anfangsknoten  $z_0$  (Beispiel: Bild 4.1.3 b).
3. Man wandle den so gewonnenen Graphen eines Moore-Klassifikators in einen deterministischen Graphen um.
4. Man prüfe, ob die Determiniertheit des Gesamt-Klassifikators erreichbar ist, ohne daß die Entscheidungsfunktion mehrdeutig wird. Ist dies nicht der Fall, so ist das Klassifikationsproblem wegen Widerspruchs nicht lösbar.



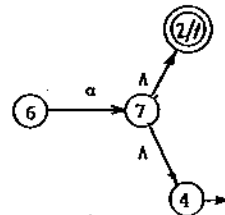
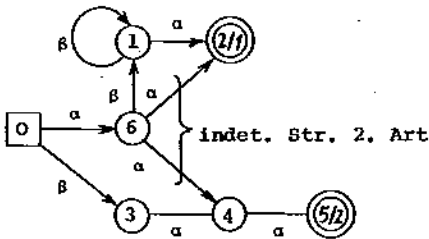
$$p(1) = \alpha \beta^* \alpha$$

$$p(2) = (\alpha \vee \beta) \alpha^2$$

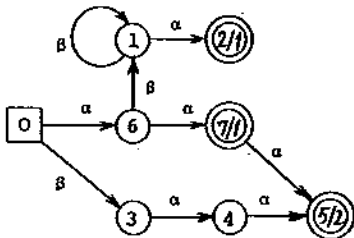
a) Problemformulierung nach Version 1



b) Transformation einer indet. Struktur 2. Art



c) Transformation einer indet. Folgestruktur 2. Art



d) Determinisierung beendet, da keine weitere indet. Strukt. 2. Art vorhanden

**Bild 4.1.3** Beispiel für die Lösung eines 2-Klassen-Problems durch Synthese eines Moore-Klassifikators

Bei Widerspruchsfreiheit der Problemstellung führt das Verfahren aus folgenden Gründen zu einer richtigen Lösung. Der durch Vereinigung der  $k$  einzelnen Akzeptoren gewonnene Gesamt-Akzeptor akzeptiert die Vereinigungsmenge  $\rho$  aller Ereignisse  $\rho(i)$  (vgl. Definition 2.2.3, wobei  $Z^i = \{z_i \mid i = 1, 2, \dots, k\}$ ). Wegen der Anwendbarkeit des graphischen Verfahrens nach Abschnitt 3 dieser Arbeit auf Akzeptoren bleibt bei Determinisierung des Akzeptors das akzeptierte Ereignis  $\rho$  unverändert. Bei Widerspruchsfreiheit bleibt die Akzeptierungs-Partition  $\Pi_k(\rho)$  unverändert infolge Satz 4.1.1. Die Verschmelzung rückwärts-äquivalenter Zustände ist zulässig, da bei einander fremden Ereignissen sich die Mengen ihrer Vorereignisse überschneiden können. Eine der wichtigsten graphischen Transformationen bei der Synthese von Moore-Klassifikatoren ist der in Bild 3.2.1.3 graphisch gezeigte Determinisierungs-Algorithmus. Bild 4.1.3 zeigt hierzu einen Lösungsweg für ein einfaches Synthesebeispiel für ein Zwei-Klassen-Problem. Die Anwendung des Moore-Klassifikators wurde durch zwei Simulationsprogramme demonstriert /44,47,59/. Die Simulationstechnik hierzu wird in Kapitel 5 ff. behandelt.

Eine Variante der vorgeschlagenen Methode zur Synthese von Moore-Klassifikatoren besteht darin, daß die Problemstellung sofort als uneigentlicher Zustandsgraph des gesuchten Moore-Akzeptors formuliert wird, der anfangs nur einen Anfangsknoten  $z_0$  und  $k$  Endknoten  $z_1$  bis  $z_k$  besitzt (Bild 4.1.2).

#### 4.2 Ein Verfahren zur Synthese von Transduktoren

In Kapitel 4.1 der vorliegenden Arbeit wurde gezeigt, daß das sequentielle Klassifikationsproblem (mehrere Klassen) über die Modellvorstellung des abstrakten Akzeptors auf das Erkennungsproblem (nur eine Klasse) zurückgeführt werden kann. Im Folgenden wird gezeigt, daß dies auch für das Problem der Informationsreduktion gilt, was insbesondere für die Automatische Zeichenerkennung und Bildauswertung von

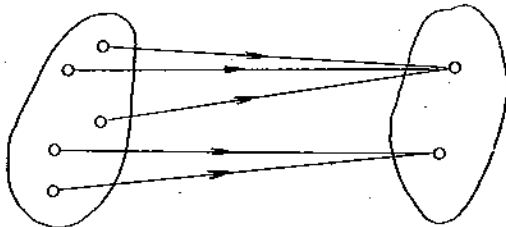
Bedeutung ist. Zur Technik der Informationsreduktion wurden auf diesem Gebiet bisher kaum praktisch verwendbare allgemeine Modellvorstellungen entwickelt, weshalb hier Experiment und Heuristik das Feld beherrschen /17-19/.

Für sequentielle Informationsreduktion bietet sich der *Transduktor* als Modellvorstellung an. Ein Transduktor /49/ ist ein abstrakter Automat mit Ausgangszuordner gemäß Bild 2.2.1/2, wobei aufgrund der Art des Ausgangszuordners Moore-Automaten und Mealy-Automaten unterschieden werden. Die Kenntnis dieser beiden Automaten-Modelle wird vorausgesetzt (z.B. nachlesbar in /21-26/). Ein Transduktor stellt funktionale Beziehungen her zwischen seinen Eingabe- und Ausgabevariablen und ist definiert gemäß Definition 2.2.1. Er bildet die Menge aller Silben des durch sein Überführungswerk dargestellten Ereignisses  $p_d$  in sein Ausgabe-Alphabet  $Y$  ab. Bei einem *deterministischen Transduktor mit Anfangszustand* ist diese Abbildung rechts-eindeutig (vgl. Bild 4.2.1). Ein solcher Transduktor ist auf spezielle Weise wie folgt definiert.

Definition 4.2.1: Ein *deterministischer Transduktor*  $\mathcal{C}$  mit *Anfangszustand* ist ein Sextupel  $\mathcal{C} = (X, Y, Z, (z_0), \delta, \lambda)$  mit  $z_0$  als Anfangszustand und den Alphabeten und Abbildungen gemäß Definition 2.2.1, wobei die Alphabete  $X$  und  $Y$  zusätzlich das Symbol  $\emptyset$  mit der Bedeutung "kein Signal" enthalten /49/.

Unter Berufung auf den folgenden Satz soll die Behandlung des Syntheseproblem es in diesem Abschnitt der Einfachheit halber im Prinzip auf den Moore-Automaten beschränkt bleiben. (Es werden nur Analogien zum Mealy-Automaten aufgezeigt.)

Satz 4.2.1: Zu jedem Moore-Automat  $\mathcal{M}$  existiert ein nicht-unterscheidbarer Mealy-Automat  $\mathcal{M}'$ , der nicht mehr Zustände hat als  $\mathcal{M}$  (z.B. /25/).



Vorbereich:  
Menge aller  
Silben von  $\rho$

Nachbereich:  
Menge der  
Ausgabesymbole

Bild 4.2.1 Abbildungsfunktion eines Transduktors

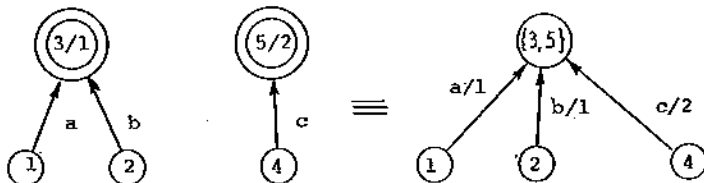


Bild 4.2.2 Veranschaulichung der Umwandlung eines Moore-Automaten in einen äquivalenten Mealy-Automaten.



Bild 4.2.3 Allgemeiner Graph eines Moore-Akzeptors und eines dazu äquivalenten Mealy-Akzeptors

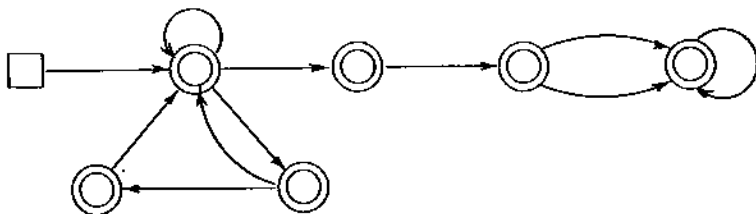
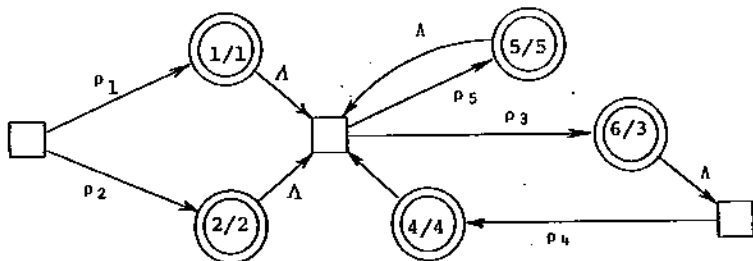


Bild 4.2.4 Beispiel des Residuengraphen eines Transduktors

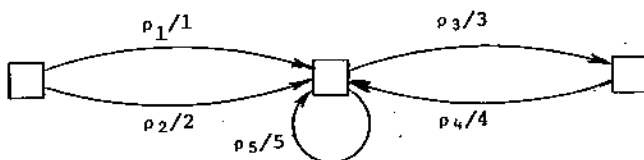
Entsprechende Umwandlungs-Algorithmen existieren hierzu, was durch Bild 4.2.2 am Beispiel eines Akzeptors veranschaulicht sei. Bild 4.2.3 zeigt die verallgemeinerte graphische Darstellung einer solchen Akzeptor-Umwandlung. Der dabei entstehende *Mealy-Akzeptor* sei gemäß folgendem Vorschlag definiert.

Definition 4.2.2: Ein *Mealy-Akzeptor*  $\mathcal{M}$  ist ein Sextupel  $\mathcal{M} = (X, Z, (z_0), \delta, (z_f), \delta')$ , wobei  $\delta' \subset \delta$  die Menge aller Übergänge in den Endzustand  $z_f$  ist. Als akzeptiert gilt die Menge aller Folgen, deren Endsymbole durch  $\delta'$  dargestellt sind.

Für die Automatische Zeichenerkennung ist ein sehr starkes Maß an Informationsreduktion charakteristisch. Eine typische Aufgabenstellung bei sequentieller Informationsreduktion hierbei ergibt bei Transduktor-Realisierung eine Abbildung der Eingabemengen in das Ausgabe-Alphabet derart, daß häufig längeren Silben von  $\rho$  nur ein einziges Ausgabe-symbol zugeordnet wird. Bei Empfang einer solchen Silbe durch einen Transduktor wird für jedes Symbol mit Ausnahme des letzten nur ein Ausgabesignal 0 ("kein Signal") ausgegeben. Bei Vorliegen eines Moore-Automaten bedeutet dies, daß nicht jeder Zustand  $z_i$  ein Ausgabesymbol ( $z_i \neq 0$ ) hat. Man kann einen uneigentlichen Zustandsgraphen eines solchen Moore-Automaten zeichnen, der nur für genau diejenigen Zustände Knoten enthält, die ein Ausgabesymbol haben (d.h. für genau diejenigen Zustände  $z_i$ , für die  $\lambda(z_i) \neq 0$  gilt). Ein nach dieser Vorschrift konstruierter Graph sei als *Residuen-Graph eines Transduktors* bezeichnet. Diese Konstruktion ist deshalb möglich, da sich für jeden Pfad von einem beliebigen Knoten zu einem beliebigen anderen Knoten ein regulärer Ausdruck finden läßt /34/. Der durch eine solche Konstruktion entstandene Graph kann als Komposition der Graphen einfacher Akzeptoren interpretiert werden nach Definition 2.2.6 (vgl. Regeln 2.3.1/2/3 b). Bild 4.2.4 zeigt einen solchen Residuen-Graphen eines Transduktors.



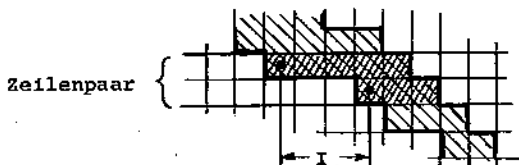
a) Komposition des Moore-Residuen-Graphen



b) Komposition des Mealy-Residuen-Graphen

**Bild 4.2.5** Zwei äquivalente Problemformulierungen als Beispiel für die Komposition des Residuen-Graphen eines Transduktors

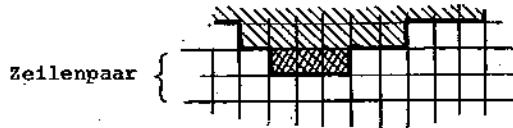
Ausgabe-Symbol	Bedeutung
0	$\Lambda$
1	$I = 0$
2	$I = 1$
3	$I = 2$
4	$I > 2$
5	$I < -2$
6	$I = -2$
7	$I = -1$
8	<i>Sprung</i>



c) Definition des Kontur-Inkrement (Beispiel für pos. Inkrement  $I = 3$ )

a) Ausgabe-Alphabet

Signal				
Code	1	2	3	4



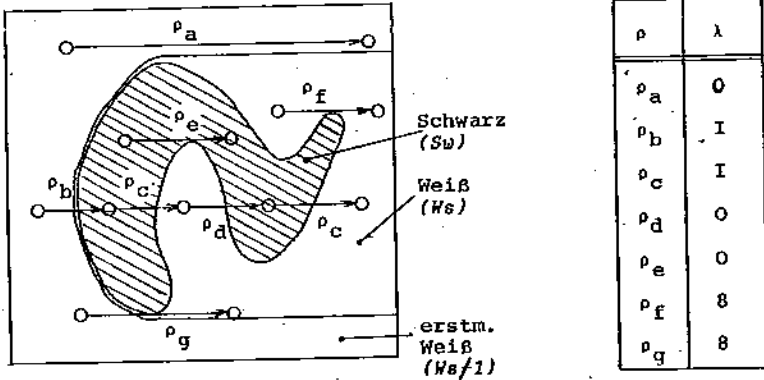
b) Eingabe-Alphabet d) Definition des *Sprung*

**Bild 4.2.6** Beispiel einer Aufgabenstellung aus der Zeichen-erkennung zur Synthese eines Transduktors

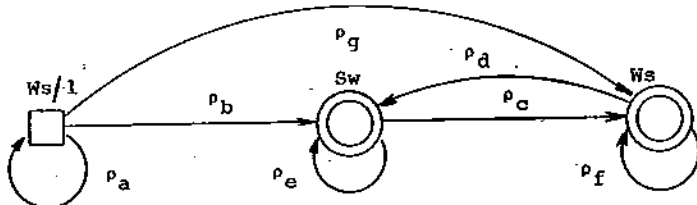
Auf der Grundlage des Transduktor-Residuen-Graphen sei folgende Synthese-Methode für Transduktoren vorgeschlagen. Die Methode besteht aus folgenden Schritten.

1. Man lege eine dem Problem angemessene Sprache und die Alphabete fest für Ein- und Ausgabe.
2. Man unterteile alle Eingabefolgen in Silben, die jeweils einer einzigen Symbol-Ausgabe zugeordnet werden sollen und ordne diese Silben entsprechend ihrer Folge nach Gleichheit der zugeordneten Ausgabesymbole. Man formuliere für jede dieser Gruppen einen regulären Ausdruck.
3. Man konstruiere gemäß der Folge dieser Ausdrücke einen Transduktor-Residuen-Graphen.
4. Man konstruiere aus dem Residuen-Graph einen deterministischen Zustandsgraphen nach dem für den Entwurf klassifizierender Akzeptoren vorgeschlagenen Verfahren (Kapitel 4.1).

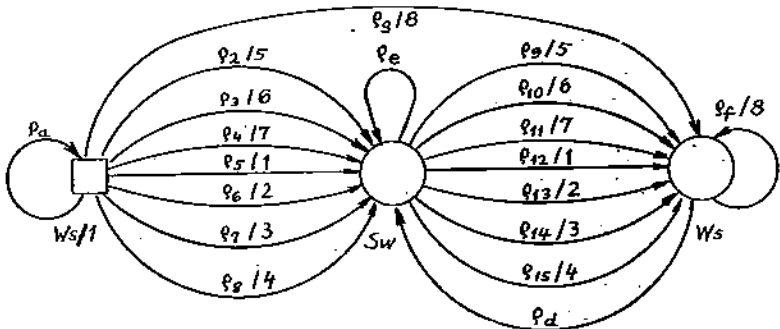
Zu den Schritten 1 bis 2 lassen sich kaum nähere Angaben machen, da diese sehr stark problemabhängig sind. Die für Schritt 2 nötigen Unterteilungen sind meist aufgrund derjenigen Schemata naheliegend, die in der Problemstellung enthalten sind. Zur Durchführung von Schritt 3 können zwei verschiedene Wege ausgewählt werden. Der erste Weg beginnt mit der graphischen Komposition des Problemes aus den Moore-Akzeptoren der Teilprobleme dadurch, daß Endknoten mit den problembedingt folgenden Anfangsknoten durch Leerzweige verknüpft werden (beispielsweise wie in Bild 4.2.5 a). Beim zweiten möglichen Weg wird davon ausgegangen, daß eine evtl. erwünschte Umwandlung in eine Mealy-Darstellung bereits vorweggenommen wird dadurch, daß die Unter-Akzeptoren vor der Komposition in Mealy-Akzeptoren umgewandelt werden (wie beispielsweise in Bild 4.2.5 b). Bei Schritt 4 bringt die Tatsache, daß nicht alle Knoten des Residuen-Graphen echte Endzustandsknoten der Gesamt-Struktur sind, keine besonderen Schwierigkeiten. Dieses Problem kommt bereits bei der Synthese minimaler gewöhnlicher Akzeptoren vor und ist von dort her gelöst.



a) Erkennung von Kontur-Inkrementen(I) und Sprüngen (8) bei Zeilendifferenz-Abtastung einer Binärmatrix



b) Gliederung der Aufgabenstellung nach Bild a



c) Residuengraph des gesuchten Transduktors

Bild 4.2.7 Problemformulierung zur Synthese eines Transduktors zur Erfassung von Kontur-Inkrementen eines Linienmusters



#### 4.2.1 Ein Synthesebeispiel aus der Zeichenerkennung

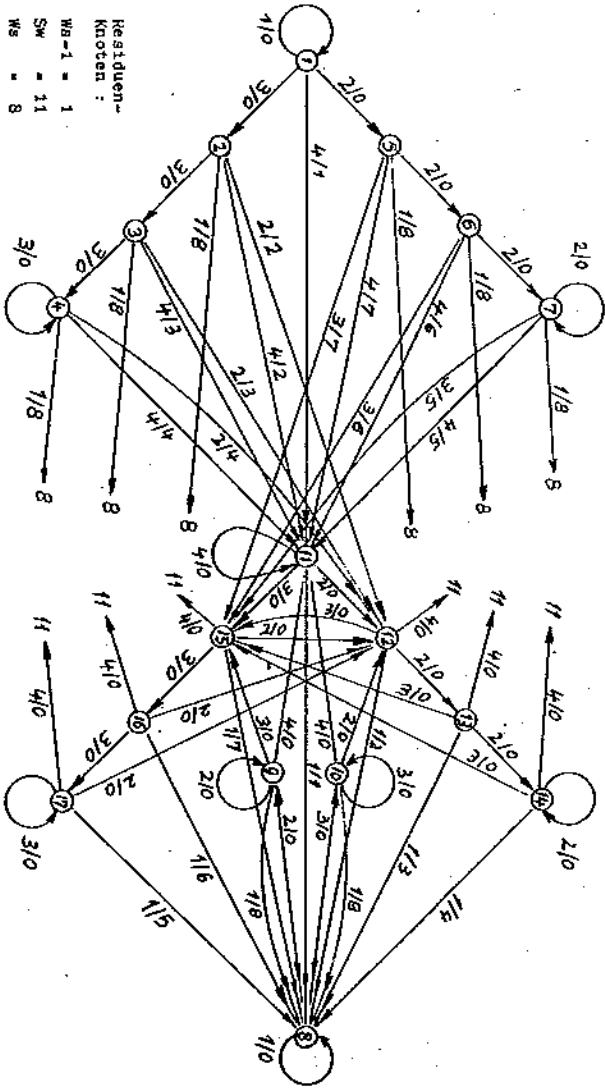
Zur Veranschaulichung des skizzierten Vorschlages sei ein Synthese-Beispiel aus der Automatischen Zeichenerkennung beschrieben. Ein als Binärmatrix gegebenes Linienmuster soll nach der Zeilendifferenz-Methode abgetastet werden (vgl. Bild 2.1.1). Gesucht wird ein Transduktor mit Anfangszustand, der für innerhalb einer jeden Doppelzeile erfaßte Konturen die *Inkrement*e gemäß Bild 4.2.6 a erkennt und an seinem Ausgang anzeigt. Als *Inkrement* wird die in Raster-Einheiten gemessene Koordinatendifferenz in Zeilenrichtung zwischen zwei benachbarten und zusammenhängenden Konturelementen bezeichnet, die aufgrund der Abtastung nach der Zeilendifferenz-Methode in einer Doppelzeile gemeinsam erfaßt werden (Bild 4.2.6 c).

Beim Vorliegen eines Linienanfanges oder eines Linienendes innerhalb einer abgetasteten Doppelzeile wird anstelle eines Inkrementes ein *Sprung* durch ein besonderes Ausgabesignal (8) angegeben (vgl. Bild 4.2.6 d).

Es sei die Aufgabe gestellt, daß nicht alle Konturen ein Ausgabesignal erzeugen sollen, sondern nur die in der Doppelzeile zuerst auftretende linke sowie alle dann auftretenden rechten Konturen. Die wesentliche Information zur Ermittlung der Kontur-Inkrementes steckt in den Konturen, d.h. in den Grenzübergängen zwischen weißer Fläche ( $W_s$  in Bild 4.2.7 a) und schwarzer Fläche ( $S_w$  in Bild 4.2.7 a) im Zuge der Abtastzeile. Solche Teile des Musters, in denen sich der abgetastete Zeilendifferenz-Code (Bild 4.2.6 b) nicht ändert, tragen keine relevante Information, weshalb eine Zustandsänderung des gesuchten Transduktors hier nicht sinnvoll ist. Es werden deshalb für den Residuen-Graph zwei Knoten mit der Kennzeichnung  $S_w$  und  $W_s$  definiert. Wegen der gewünschten unterschiedlichen Behandlung von erster Kontur und allen sonstigen Konturen muß unterschieden werden, ob eine vorliegen-

Unter-Ereignis $\rho$	$\lambda(\rho)$
$\rho_a = \rho_1 = 1^*$	0
$\rho_b = \rho_2 \cup \rho_3 \cup \rho_4 \cup \rho_5 \cup \rho_6 \cup \rho_7 \cup \rho_8$ $\rho_2 = 1^* 2^3 2^* 4$ $\rho_3 = 1^* 2^2 4$ $\rho_4 = 1^* 2^1 4$ $\rho_5 = 1^* 4$ $\rho_6 = 1^* 3^1 4$ $\rho_7 = 1^* 3^2 4$ $\rho_8 = 1^* 3^3 3^* 4$	5 6 7 1 2 3 4
$\rho_c = \rho_9 \cup \rho_{10} \cup \rho_{11} \cup \rho_{12} \cup \rho_{13} \cup \rho_{14} \cup \rho_{15}$ $\rho_9 = 4^* 3^3 3^* 1$ $\rho_{10} = 4^* 3^2 1$ $\rho_{11} = 4^* 3^1 1$ $\rho_{12} = 4^* 1$ $\rho_{13} = 4^* 2^1 1$ $\rho_{14} = 4^* 2^2 1$ $\rho_{15} = 4^* 2^3 2^* 1$	5 6 7 1 2 3 4
$\rho_d = 1^* (2^* \cup 3^*) 4$	0
$\rho_e = 4^* (2^* \cup 3^*) 4$	0
$\rho_f = 1^* (2^* \cup 3^*) 1$	8
$\rho_g = 1^* (2^* \cup 3^*) 1$	8

**Tafel 4.2.1** Abbildungsfunktion  $\lambda(\rho)$  des gesuchten Transduktors beim Syntheseproblem nach Bild 4.2.7



Residuen-  
Knoten :

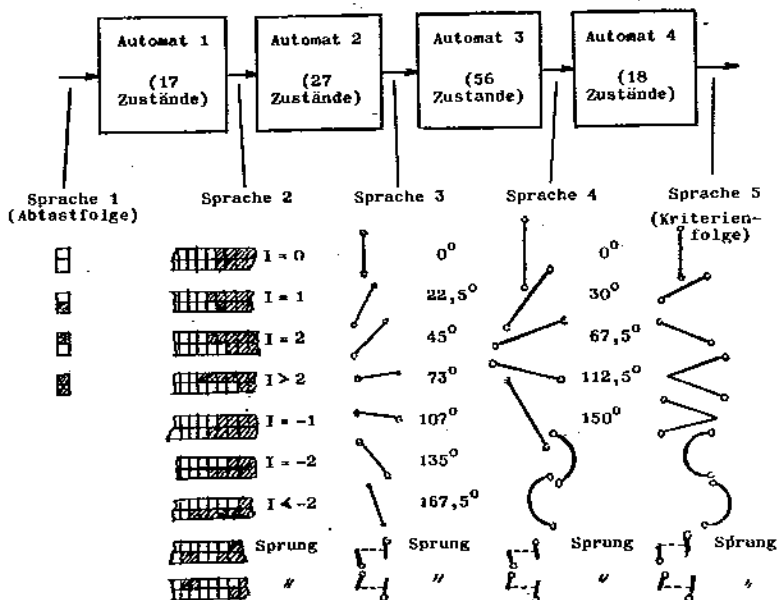
- W<sub>1</sub>-1 = 1
- SW = 11
- MS = 8

Bild 4.2.8 Minimallösung des Syntheseprogrammes nach Bild 4.2.7

de weiße Fläche innerhalb der Doppelzeile erstmalig aufgetreten ist, oder zwischenzeitlich "Schwarz" ( $S_w$ ) abgetastet worden ist. Aus diesem Grunde wird für den Residuen-Graphen ein besonderer Knoten mit der Kennzeichnung  $W_s/1$  für erstmaliges Weiß definiert. Bild 4.2.7 a zeigt schematisch am Bild eines Masters typische Teilstücke abgetasteter Doppelzeilen mit den zuzuordnenden Ausgabesignalen. Bild 4.2.7 b zeigt die darin sichtbare Syntax der Eingabesprache des gesuchten Transduktors in Form eines uneigentlichen Zustandsgraphen, wobei nur zwischen den syntaktischen Kategorien I (Inkrement), 8 (Sprung) und 0 (kein Ausgabesymbol) unterschieden wird. Bild 4.2.7 c zeigt den dazugehörigen Mealy-Residuen-Graphen. Die regulären Ausdrücke der einzelnen Zweige dieses Graphen sind in Tafel 4.2.1 zusammengefaßt.

Bild 4.2.8 zeigt den Zustandsgraphen der Lösung des Synthese-Problems. Dieser wurde aus dem Residuen-Graphen nach Bild 4.2.7 c auf graphischem Wege ermittelt, wobei die Transformationen für Strukturen mit Potenzen (vgl. Kapitel 3.3) von Wichtigkeit sind. Bild 4.2.6 a gibt das Ausgabe-Alphabet und Bild 4.2.6 b das Eingabe-Alphabet des ermittelten Transduktors an.

Bild 4.2.9 zeigt das Blockdiagramm einer aus 4 Transduktoren bestehenden Kaskade zur Gewinnung von Kriterien aus Konturen von Linienmustern /47/. Durch stufenweise Informationsreduktion erfolgt eine Darstellung in Sprachen zunehmender syntaktischer Einschränkung. Die Semantik der Alphabete dieser Sprachen sowie deren Syntax ist in Bild 1.2.9 schematisch wiedergegeben. Zur Simulation der Erkennung handgeschriebener Ziffern mit Hilfe dieser Transduktor-Kaskade wurde ein Simulationsprogramm in der Quellen-Sprache Algol geschrieben /47/. Als Klassifikator diente hierbei ein Moore-Klassifikator in Suchlistendarstellung /59/. Klassifikator-Synthese und Erkennung wurden im on-line-Betrieb über ein Lichtgriffel-Eingabegerät auf einem Digitalrechner CD 3300 simuliert.



Sprache	durchschnittliche Zahl von Symbolen je Kontur *)	Form der Kontur-Beschreibung
1	4032	Zeilendifferenz-Folgen (vgl. Bild 2.1.1)
2	< 63	Kontur-Inkrement-Folgen (s. Bild 4.2.6)
3	20	Polygonzug gleichlanger Geradenstücke
4	6	Kette aus längeren Geradenstücken und Kreisbogensegmenten *)
5	3	Kriterien-Folge

\*) bei handgeschriebenen Ziffern und 64:64-Matrix

\*) mit geglätteter Kontur unter Beseitigung von Flecken und Fehlstellen

**Bild 4.2.9** Beispiel für eine Transduktor-Kaskade zur Lösung eines Informationsreduktions-Problems aus der Zeichenerkennung: Ermittlung von Kontur-Kriterien bei Zeilendifferenz-Abtastung.