

Paul Koop

Algorithmisch Rekursive
Sequenzanalyse

Algorithmischer Strukturalismus:

Formalisierung genetischer
Strukturalismus:

Ein Versuch, dazu beizutragen, den
genetischen Strukturalismus
falsifizierbar zu machen

Die Algorithmisch Rekursive Sequenzanalyse (Aachen 1994) ist eine Methode zur Analyse endlicher diskreter Zeichenketten.

Ndiaye, Alassane (Rollenübernahme als Benutzermodellierungsmethode : globale Antizipation in einem transmutierbaren Dialogsystem 1998) und Krauß, C. C., & Krueger, F.R. (Unbekannte Signale 2002) veröffentlichten äquivalente Methoden. Genial ist, etwas Einfaches einfach zu denken.

Seit Anfang des 21. Jahrhunderts wird die Konstruktion von Grammatiken aus gegebenen empirischen Zeichenketten in der Computerlinguistik unter dem Stichwort Grammatikinduktion diskutiert (Alpaydin, E. (2008): Maschinelles Lernen, Shen, Chunze(2013): Effiziente Grammatikinduktion) oder (Dehmer 2005 Strukturelle Analyse)(Krempel 2016 Netze, Karten, Irrgärten). Mit [sequitur](#) definieren Nevill-Manning Witten (Identifying Hierarchical Structure in Sequences: A linear-time algorithm 1999) eine Grammar Induktion zur Komprimierung von Zeichenketten. Graphen, Grammatiken und Transformationsregeln sind natürlich erst der Anfang. Denn eine Sequenzanalyse ist erst abgeschlossen, wenn, wie bei der algorithmisch rekursiven Sequenzanalyse mindestens eine Grammatik angegeben werden kann für die ein Parser die Sequenz als wohlgeformt identifiziert, mit der ein Transduktor künstliche Protokolle erzeugen kann, die äquivalent zur untersuchten empirischen Sequenz sind und zu der ein Induktor mindestens eine äquivalente Grammatik erzeugen kann. Gold (1967) formulierte das Problem in Antwort auf Chomsky (1965).

Der algorithmische Strukturalismus ist widerspruchsfrei, empirisch bewährt, galileisch, naturalistisch, darwinisch und ein Ärgernis für tiefenhermeneutische, konstruktivistische, postmodernistische und (post)strukturalistische Sozialphilosophen. Ich freue mich über Erben, die die Arbeit fortsetzen oder sich inspirieren lassen.

Eine soziale Handlung ist ein Ereignis im Möglichkeitsraum aller sozialen Handlungen. Der Sinn einer sozialen Handlung ist die Menge aller möglichen Anschlusshandlungen und ihrer Auftrittswahrscheinlichkeit. Der Sinn muss nicht deutend verstanden werden, sondern kann empirisch rekonstruiert werden. Die Rekonstruktion kann durch Bewährungsversuche an empirischen Protokollen bewährt oder falsifiziert werden.

Seit Mitte der 1970er bis heute finden irrationalistische oder antirationalistische Ideen unter akademischen Soziologen in Amerika, Frankreich, Großbritannien und Deutschland zunehmend Verbreitung. Die Ideen werden als Dekonstruktionismus, Tiefenhermeneutik, Wissenssoziologie, Sozialkonstruktivismus, Konstruktivismus oder Wissenschafts- und Technologieforschung bezeichnet. Der Oberbegriff für diese Bewegungen ist (Post)strukturalismus oder Postmodernismus. Alle Formen des Postmodernismus sind antiwissenschaftlich, antiphilosophisch, antistrukturalistisch, antinaturalistisch, antigalileisch, antidarwinisch und allgemein antirational. Die Sicht der Wissenschaft als eine Suche nach Wahrheiten (oder annähernden Wahrheiten) über die Welt wird abgelehnt. Die natürliche Welt spielt eine kleine oder gar keine Rolle bei der Konstruktion wissenschaftlichen Wissens. Die Wissenschaft ist

nur eine andere soziale Praxis, die Erzählungen und Mythen hervorbringt, die nicht mehr Gültigkeit haben als die Mythen vorwissenschaftlicher Epochen.

Man kann den Gegenstand der Sozialwissenschaften so beobachten, wie die Astronomie ihren Gegenstand beobachtet. Wenn sich der Gegenstand der Sozialwissenschaften dem direkten Zugang oder Laborexperiment so entzieht, wie Himmelsobjekte (Gerichtsverhandlung, Verkaufsgespräch, Vorstandssitzung, et cetera), bleibt nur, ihn interpretationsfrei rein physikalisch zu beobachten und die Beobachtungen rein physikalisch zu protokollieren. Die Protokolle könnte man dann natürlich auch ohne Rückbindung an Physik, Chemie, Biologie, Evolutionsbiologie, Zoologie, Primatenforschung und Lifescience interpretieren. Diese überprüfungsfreie Interpretation nennt man bei der Himmelsbeobachtung dann Astrologie. In den Sozialwissenschaften nennt man auch diese überprüfungsfreie Interpretation Soziologie. Beispiele sind Konstruktivismus (Luhmann), systemische Heilslehren, Postmodernismus, Poststrukturalismus, oder Theorie kommunikativen Handelns (Habermas). Regelbasierte Agentenmodelle arbeiten deshalb bisher mit heuristischen Regelsystemen. Diese Regelsysteme sind nicht empirisch bewährt. Wie auch in der Astrologie könnte man damit natürlich auch in der Soziologie Computermodele erstellen, die ebenso wie astrologische Modelle wenig empirischen Erklärungsgehalt hätten. Einige nennen das Sozionik. Man kann aber auch die Protokolle unter Beachtung von Physik, Chemie, Biologie, Evolutionsbiologie, Zoologie, Primatenforschung und Lifescience interpretieren und auf empirische Gültigkeit überprüfen. Die Beobachtung von Himmelsobjekten nennt

man dann Astronomie. In den Sozialwissenschaften könnte man von Sozionomie oder Soziomatik sprechen. Das eigentlich ist Soziologie. Heraus kämen dabei keine grossen Weltanschauungen, sondern wie in der Astronomie Modelle mit begrenzter Reichweite, die empirisch überprüfbar sind und an Evolutionsbiologie, Zoologie, Primatenforschung und Lifescience anschlussfähig sind. Diese Modelle (Differentialgleichungen, formale Sprachen, Zellularautomaten, etc) liessen die Deduktion empirisch überprüfbarer Hypothesen zu, wären also falsifizierbar. Eine solche Sozionomie oder Soziomatik gibt es noch nicht. Ich würde formale Sprachen als Modellsprachen für empirisch bewährte Regelsysteme bevorzugen. Denn Regelsysteme für Gerichtsverhandlungen oder Verkaufsgespräche z.B. (Modelle mit begrenzter Reichweite, Multiagentensysteme, zelluläre Automaten) lassen sich eher mit formalen Sprachen als mit Differentialgleichungen modellieren.

Der Algorithmische Strukturalismus ist ein Versuch, dazu beizutragen, den genetischen Strukturalismus (ohne Auslassung und ohne Hinzufügung) in eine falsifizierbare Form zu übersetzen und empirisch bewährte Regelsysteme zu ermöglichen. Die Algorithmisch Rekursive Sequenzanalyse ist der erste systematische Versuch, einer naturalistischen und informatischen Ausformulierung des genetischen Strukturalismus als memetisches und evolutionäres Modell. Die Methodologie der Algorithmisch Rekursiven Sequenzanalyse ist der Algorithmische Strukturalismus. Der Algorithmische Strukturalismus ist eine Formalisierung des genetischen Strukturalismus. Der genetische Strukturalismus (Oevermann) unterstellt einen intentionsfreien, apsychischen Möglichkeitsraum algorithmischer Regeln, die die

Pragmatik wohlgeformter Ereignisketten textförmig strukturieren (Chomsky, McCarthy, Papert, Solomon, Lévi-Strauss, de Saussure, Austin, Searle). Der Algorithmische Strukturalismus ist der Versuch den genetischen Strukturalismus falsifizierbar zu machen. Der Algorithmische Strukturalismus ist galileisch und an Habermas und Luhmann so wenig anschlussfähig, wie Galilei an Aristoteles. Natürlich kann man sich bemühen, an Luhmann oder Habermas anschlussfähig zu bleiben und Luhmann oder Habermas zu algorithmisieren. Algorithmisieren kann man alle Artefakte, zum Beispiel die Astrologie oder das Schachspiel. Und man kann normative Agenten verteilter künstlicher Intelligenz, Zelluläre Automaten, neuronale Netze und andere Modelle mit heuristischen Protokollsprachen und Regeln modellieren. Das ist zweifellos theoretisch wertvoll. So wird es keinen soziologischen Theoriefortschritt geben. Gesucht ist eine neue Soziologie, die die Replikation, Variation und Selektion sozialer Replikatoren, gespeichert in Artefakten und neuralen Mustern, modelliert. Diese neue Soziologie wird an Habermas oder Luhmann ebenso wenig anschlussfähig sein wie Galilei an Aristoteles. Und ihre basalen Sätze werden so einfach sein wie die newtonschen Gesetze. So wie Newton die Begriffe Bewegung, Beschleunigung, Kraft, Körper und Masse operational definierte, so wird diese Theorie die sozialen Replikatoren, ihre materiellen Substrate, ihre Replikation, Variation und Selektion algorithmisch und operational definieren und sequenzanalytisch sichern. Soziale Strukturen sind sprachlich codiert und basieren auf einem digitalen Code. Gesucht sind syntaktische Strukturen einer Kultur codierenden Sprache. Aber dies wird keine philosophische Sprache sein, sondern eine Sprache, die Gesellschaft codiert und erschafft. Diese Sprache codiert die Replikation, Variation

und Selektion kultureller Replikatoren. Auf dieser Basis werden dann normative Agenten verteilter künstlicher Intelligenz, Zelluläre Automaten, neuronale Netze und andere Modelle andere als heuristische Protokollsprachen und Regelsysteme nutzen können, um die Evolution kultureller Replikatoren zu simulieren.

Thematisch bewegt sich der Algorithmische Strukturalismus im Grenzgebiet zwischen Informatik und Soziologie. Die Algorithmische Strukturalismus unterstellt, dass die soziale Wirklichkeit selbst (Weltware, Welt 2) nicht kalkülfähig ist. Die soziale Wirklichkeit hinterlässt bei ihrer Reproduktion und Transformation rein physikalisch semantisch unspezifische Spuren (Protokolle, Hardware, Welt 1). Diese Spuren können als Texte (diskrete endliche Zeichenketten, Software, Welt 3) verstanden werden. Es wird dann gezeigt, dass eine Approximation der Transformationsregeln der sozialen Wirklichkeit (latente Sinnstrukturen, Regeln im Sinne von Algorithmen) durch Konstruktion formaler Sprachen (Welt 3, Software) möglich ist. Diese Methode ist die Algorithmisch Rekursive Sequenzanalyse. Diese linguistische Struktur ist der Motor der memetischen Reproduktion kultureller Replikatoren. Diese algorithmisch rekursive Struktur ist natürlich nicht (sic!) an Habermas und Luhmann angeschlossen. Galilei ist ja auch nicht an Aristoteles angeschlossen!

Durch Lesartenproduktion und Lesartenfalsifikation wird Sequenzstelle für Sequenzstelle informell das Regelsystem erzeugt. Das informelle Regelsystem wird in ein K-System übersetzt. Mit dem K-System wird dann eine Simulation durchgeführt. Das Ergebnis der Simulation, eine terminale endliche Zeichenkette, wird mit der empirisch gesicherten Spur statistisch verglichen.

Das bedeutet nicht, dass Subjekte in irgendeinem Sinne von Bedeutung Regeln im Sinne von Algorithmen folgen. Die soziale Wirklichkeit ist unmittelbar nur sich selbst zugänglich. Völlig unzugänglich sind die inneren Zustände der Subjekte. Aussagen über diese inneren Zustände von Subjekten sind Derivate aus den gefundenen latenten Sinnstrukturen, Regeln im Sinne von Algorithmen. Bevor eine Annahme über den inneren Zustand eines Subjektes formuliert werden kann, müssen zuerst diese latenten Sinnstrukturen, Regeln im Sinne von Algorithmen, als Möglichkeitsraum von Sinn und Bedeutung gültig bestimmt werden. Sinn meint nicht ein ethisch gutes, ästhetisch schönes oder empathisch nachvollzogenes Leben, sondern einen intelligiblen Zusammenhang, Regeln im Sinne von Algorithmen.

Die latenten Sinnstrukturen, Regeln im Sinne von Algorithmen, erzeugen diachronisch eine Kette von Selektionsknoten (Parameter I), wobei sie synchronisch zum Zeitpunkt t aus dem Selektionsknoten t den Selektionsknoten $t+1$ erzeugen (Parameter II). Dem entspricht eine kontextfreie formale Sprache (K-Systeme), die aus dem Selektionsknoten zum Zeitpunkt t durch Anwendung von Produktionsregeln den Selektionsknoten $t+1$ erzeugt.

Dabei ist jeder Selektionsknoten ein Zeiger auf rekursiv ineinander verschachtelte K-Systeme. So kann wie mit einem Mikroskop in die Fallstruktur hineingezoomt werden. Die Menge der K-Systeme bilden eine Case Structure Modelling Language "CSML".

Die Approximation lässt sich beliebig nahe an die Transformation der sozialen Wirklichkeit annähern. Dabei werden den Produktionen Maße zugeordnet, die ihrer empirischen gesicherten Pragmatik/Semantik entsprechen. Sie bilden topologisch ein rekursives Transitionsnetz diskreter nichtmetrischer Ereignismengen über denen ein algorithmisches Regelsystem arbeitet.

K- Systeme K sind formal durch ein Alphabet ($A := \{a_1, a_2, \dots, a_n\}$), alle Worte über dem Alphabet (A^*), Produktionsregeln (p) dem Auftrittsmaß h (Pragmatik/Semantik) und einer axiomatischen ersten Zeichenkette ($k_0 k_1 k_2 \dots$) definiert:

$$\begin{aligned} K &:= (A, P, k_0) \\ A &:= \{a_1, a_2, \dots, a_n\} \\ p &:= A \rightarrow A \\ p(a_i) &\in P \\ p &:= A \times H \times A \\ H &:= \{h \mid 0 \leq h \leq 100 \wedge h \in \mathbb{N}\} \\ k_0 &\in A^* \wedge k_i \in A \end{aligned}$$

Das Auftrittsmaß h lässt sich spieltheoretisch expandieren (vgl. Diekmann).

Vom Axiom k_0 ausgehend, erzeugt ein K-System eine Zeichenkette $k_0 k_1 k_2 \dots$ indem die Produktionsregel p auf das Zeichen i einer Kette angewendet wird:

$$a_{i+1} := P(a_i)$$

$$k_i := a_{i-2}a_{i-1}a_i$$

$$k_{(i+1)} := a_{i-2}a_{i-1}a_i P(a_i)$$

Ein strenges Maß für die Zuverlässigkeit der Zuordnung der Interakte zu den Kategorien (vorläufige Formative da im Prinzip ad infinitum zu approximieren) ist die Anzahl der von allen Interpreten übereinstimmend vorgenommene Zuordnung (vgl. MAYRING 1990, S. 94ff, LISCH/KRIZ 1978, S. 84ff). Diese Zahl muss dann noch durch Relativierung um die Anzahl der Interpreten normalisiert werden. Dieser Koeffizient ist dann definiert mit:

$$R_{ars} := \frac{N * Z}{\sum_{i=1}^N I_i}$$

N:= Anzahl der Interpreten

Z:= Anzahl der total übereinstimmenden Zuordnungen

li:=Anzahl der Zuordnungen des Interpreten li

Eine Beispielsitzung unter clisp mit dem K-system für Verkaufsgespräche:

Das Beispiel ist Ergebnis umfangreicher Sequenzanalysen von Verkaufsgesprächen in den Jahren 1994, 1995 und 1996. Dabei wurden in großem Umfang Spuren von Verkaufs- und Kaufsinterakten gesichert: Tonbandprotokolle von Interakten im Einzelhandel und auf Märkten. Eine Auswahl dieser Protokolle wurden transkribiert und einer umfangreichen objektiv hermeneutischen Interpretation unterworfen. Aus dieser Auswahl wurde dann ein Transkript einer aufwendigen vollständigen Algorithmisch Rekursiven Sequenzanalyse unterworfen. Alle Arbeiten wurden umfangreich dokumentiert und vollständig zusammengefasst. (Die Dokumente werden auf Anfrage gerne vollständig zur Verfügung gestellt.)

```
[3]> (s vkg)
((KKG VBG) (((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAA VAA)))
 ((KBBB UBBB) (KBA UBA)) ((KBBB UBBB) (KBA UBA)) ((KBBB UBBB) (KBA UBA))
 ((KAE VAE) (KAA VAA)))
(KAV UAU))
[4]> (s vkg)
((KKG VBG)
 (((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAE VAE) (KAE VAE) (KAE VAE) (KAA VAA)))
 (((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAE VAE) (KAA VAA)))
 (((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAA VAA)))
 (((KBBB UBBB) (KBA UBA)) ((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAA VAA)))
 (KAV UAU))
[5]> (s vkg)
((KKG VBG)
 (((KBBB UBBB) (KBA UBA)) ((KBBB UBBB) (KBA UBA))
 ((KAE VAE) (KAE VAE) (KAA VAA)))
 (((KBBB UBBB) (KBA UBA)) ((KBBB UBBB) (KBA UBA)) ((KBBB UBBB) (KBA UBA))
 ((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAA VAA)))
 (((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAA VAA)))
 (((KBBB UBBB) (KBA UBA)) ((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAA VAA)))
 (((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAE VAE) (KAA VAA)))
 (((KBBB UBBB) (KBA UBA)) ((KBBB UBBB) (KBA UBA)) ((KAE VAE) (KAA VAA)))
 (KAV UAU))
[6]> _
```

Paul Koop K-System VKG Transduktor Verkaufsgespräch in Lisp

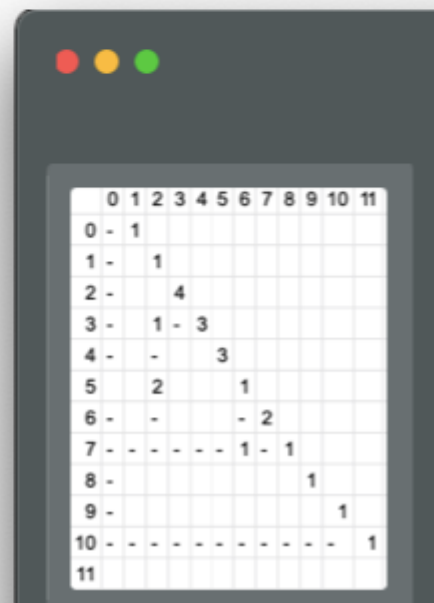
```

;; Korpus
(define korpus (list 'KBG 'VBG 'KBBd 'VBBd 'KBA 'VBA 'KBBd 'VBBd
                    'KBBd 'VBBd 'KBA 'VBA 'KBBd 'VBBd 'KBA 'VBA 'KAE
                    'VAE 'KAE 'VAE 'KAA 'VAA 'KAV 'VAV));

;; Lexikon
(define lexikon (vector 'KBG 'VBG 'KBBd 'VBBd 'KBA 'VBA 'KAE 'VAE
                       'KAA 'VAA 'KAV 'VAV)) ;; 0 - 12

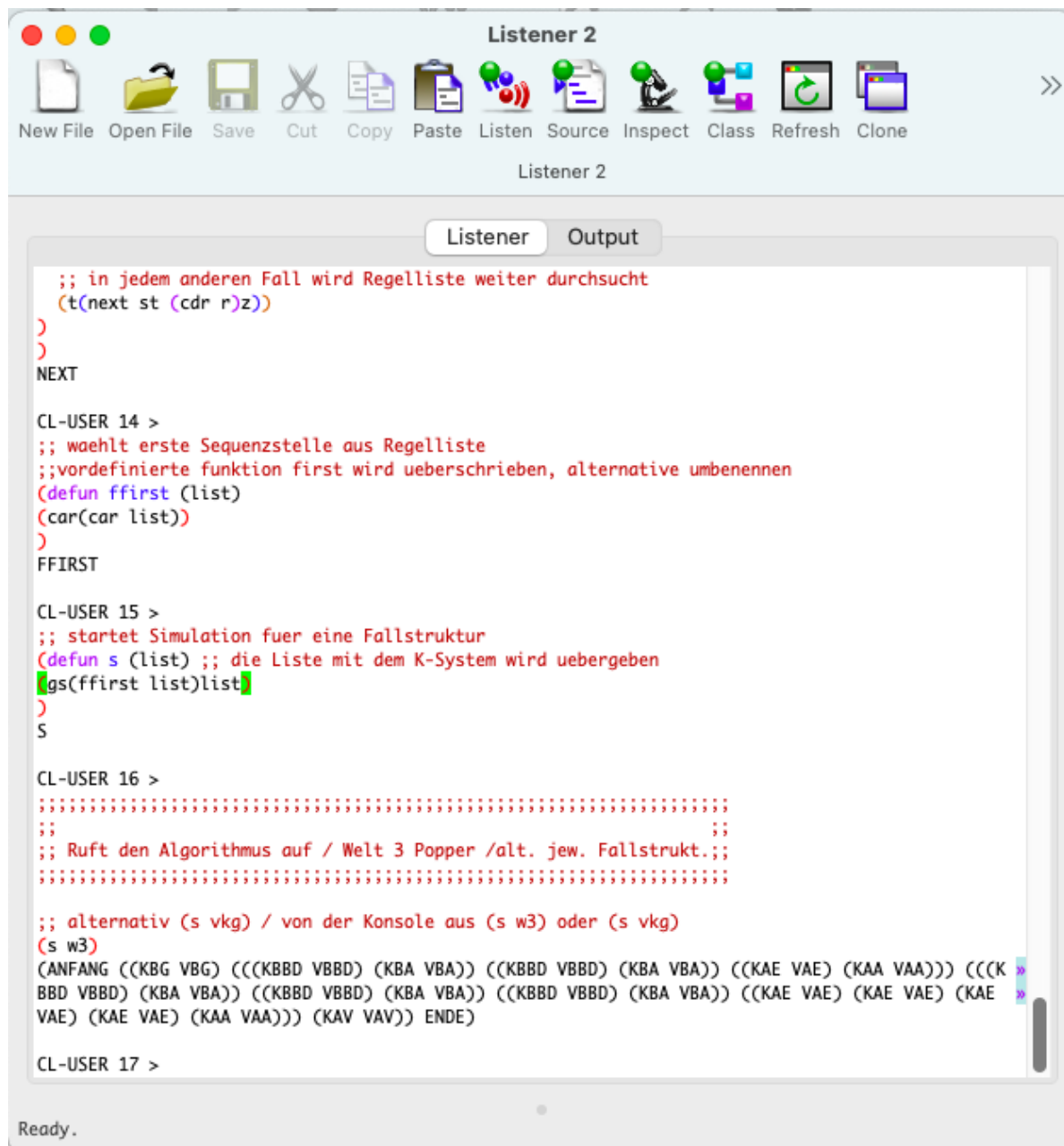
> (transformationenZaehlen korpus)
> (grammatikerstellen matrix)
(KBG -> . VBG)
(VBG -> . KBBd)
(KBBd -> . VBBd)
(VBBd -> . KBBd)(VBBd -> . KBA)
(KBA -> . VBA)
(VBA -> . KBBd)(VBA -> . KAE)
(KAE -> . VAE)
(VAE -> . KAE)(VAE -> . KAA)
(KAA -> . VAA)
(VAA -> . KAV)
(KAV -> . VAV)
> (matrixausgeben matrix)
010000000000
001000000000
000400000000
001030000000
000003000000
002000100000
000000020000
000000101000
000000000100
000000000010
000000000001
000000000000
>

```



	0	1	2	3	4	5	6	7	8	9	10	11
0 -	1											
1 -		1										
2 -			4									
3 -		1		3								
4 -					3							
5 -		2				1						
6 -							2					
7 -						1		1				
8 -									1			
9 -										1		
10 -											1	
11												

Paul Koop K-System VKG Induktor Sitzung Scheme



Paul Koop K-System VKG Transduktor Sitzung mit Lisp

```
Auswählen Eingabeaufforderung
Demo-Parser Chart-Parser Version 1.0(c)1992 by Paul Koop
-----> KBG VBG KBBD VBBD KBA VBA KAE VAE KAA VAA KAV VAV
-----> KBG VBG KBBD VBBD KBA VBA KAE VAE KAA VAA KAV VAV
VKG
  BG
  KBG
  ----> KBG
  VBG
  ----> VBG
  VT
  B
  BBD      KBBD
  ----> KBBD
  VBBD----> VBBD
  BA
  KBA
  ----> KBA
  VBA ----> VBA
  A      AE      KAE ----> KAE
  VAE
  ----> VAE
  AA      KAA ----> KAA
  VAA ----> VAA
  AV      KAV ----> KAV
  VAV ----> VAV

C:\Users\User\Documents\VKGPARSER>
```

Paul Koop K-System VKG PARSER Sitzung an der Konsole (Erstellt mit Object Pascal)

Die Zeichen der Zeichenkette sind ohne vordefinierte Bedeutung. Theoretisch relevant ist allein die Syntax ihrer Verknüpfung. Sie definiert die Fallstruktur. Die semantische Interpretation der Zeichen ist alleine eine Interpretationsleistung eines menschlichen Lesers. Im Prinzip ist auch eine visuelle Interpretation (die animiert werden kann) etwa zur automatischen Synthese von Filmsequenzen möglich.:

Ein menschlicher Leser kann die Zeichen interpretieren:

Verkaufsgespräche	VKG
Verkaufstätigkeit	VT
Bedarfsteil	B
Abschlußteil	A
Begrüßung	BG
Bedarf	Bd
Bedarfsargumentation	BA
Abschlußeinwände	AE
Verkaufsabschluss	AA
Verabschiedung	AV
vorangestelltes K	Kunde
vorangestelltes V	Verkäufer

1	(setq vkg '(((s bg)100(s vt)) ((s vt)50(s vt)) ((s vt)100(s av))))	Parameter II
2	(setq av '((kav 100 vav)))	Parameter II
3	(setq bg '((kbg 100 vbg)))	Parameter II
4	(setq vt '(((s b)50(s b)) ((s b)100(s a))))	Parameter II

5	(setq a '(((s ae)50(s ae)) ((s ae)100(s aa))))	Parameter II
6	(setq b '(((s bbd) 100 (s ba))))	Parameter II
7	(setq aa '((kaa 100 vaa)))	Parameter II
8	(setq ae '((kae 100 vae)))	Parameter II
9	(setq ba '((kba 100 vba)))	Parameter II

10	<pre>(setq bbd ' (kbbd 100 vbdd)))</pre>	Parameter II
11	<pre>(defun gs (s r) (cond ((equal s nil)nil) ((atom s)(cons s(gs(next s r(random 100))r))) (t (cons(eval s)(gs(next s r(random 100))r)))))</pre>	Parameter I
12	<pre>(defun next (s r z) (cond ((equal r nil)nil) ((and(<=z(car(cdr(car r))))) (equal s(car(car r)))(car(reverse(car r)))) (t(next s (cdr r)z))))</pre>	Parameter I
13	<pre>(defun first (list) (car(car list)))</pre>	Parameter I
14	<pre>(defun s () (setq protokoll(gs(first vkg)vkg)))</pre>	Parameter I

Es wurde ein Reliabilitätskoeffizient von

$$R_{ars} = \frac{2 * 35}{118} = 0.59$$

Correlations				Test Statistics			
		Kodierer1	Kodierer2		Kodierer1	Kodierer2	
Kodierer1	Correlation		.59	Chi-Square	2.60	2.00	
	Sig.		.09	df	6	5	
Kodierer2	Correlation	.59		Asymp. Sig.	.86	.85	
	Sig.	.09					

gemessen.

Die soziale Wirklichkeit selbst aber ist nicht kalkülfähig und nur sich selbst im Moment der Transformation zugänglich.

Geisteswissenschaftliche, konstruktivistische und postmoderne Ansätze sind mir methodisch fremd. Mead, Parsons, Weber, Simmel, Mannheim/Scheler, Berger/Luckmann, Maturana, Varela, Habermas und Luhmann habe ich hinter mir gelassen. Albert, Axelrod, Esser, Diekmann, Troitzsch, Popper, Brezinka, Rössner, Dawkins, Dennett, Hofstadter, Rucker, Blackmore überzeugen mich mehr. Persönlich bevorzuge ich eine linguistische evolutionäre Perspektive und die damit verbundene Modellierung kultureller Replikatoren mit formalen Sprachen. Aus der diskreten Struktur der Materie geht die linguistische Struktur der biologischen Evolution und die linguistische Struktur der kulturellen Replikatoren hervor. Ich bevorzuge deshalb einen algorithmischen Strukturalismus.
