Komplexität und Messung von Komplexität

Autor:

M. Flückiger & M. Rauterberg

Technical Report IfAP/ETH/CC-01/95

Institut für Arbeitspsychologie

ETH Zürich

1995

Abstract

Die vorliegende Arbeit befasst sich mit Komplexität und der Messung von Komplexität. Es wird eine Sammlung mehr und minder gängiger Komplexitätsmasse vorgestellt. Die Definitionen der einzelnen Masse sind, was ihre Präzision, Anwendbarkeit und ihr Herkunftsgebiet betrifft, sehr unterschiedlich. Trotzdem erscheinen immer wieder gleiche Ideen und Muster dahinter. Die wichtigsten Masse sind Berechnungskomplexität, algorithmische Komplexität, logische Tiefe und thermodynamische Tiefe.

Dannach wird auf die Eigenschaft von Komplexitätsstufen von verschiedenen Systemen eingegangen und weitere Ansätze gesammelt.

Schliesslich wird versucht, einige qualitative Unterschiede und Gemeinsamkeiten zwischen Komplexitätsmassen aufzuzeigen. Dabei werden die Begriffe Tiefe vs. Breite, Objektivität vs. Subjektivität, hierarchisch vs. kontinuierlich, sowie Schwierigkeit und Verständlichkeit vs. Komplexität untersucht. Bei der Untersuchung der den Komplexitätsmassen zugrunde liegenden Modellen wird ersichtlich, dass diese Modelle entscheidenden Einfluss auf das Komplexitätsmass haben. Insbesondere wird die Meinung vertreten, dass die Touringmaschine als Modell für Komplexität zu schwach ist.

Inhaltsverzeichnis

1. Einleitung: Begriffliches

In diesem Abschnitt geht es darum, dem Begriff Komplexität zuerst mit umgangssprachlichen Mitteln, dann aber etwas genauer vom Wesen her, auf die Spur zu kommen.

Ganz allgemein beschreibt die *Komplexität* eine Eigenschaft eines Systems. Sie beschreibt die Struktur, den inneren Aufbau.

Bei einer kleinen Umfrage in meinem Kollegenkreis, was Komplexität sei, kamen in den Antworten sehr häufig die Ausdrücke *Unverständlichkeit* und *Schwierigkei*t vor. Die Komplexität eines Systems ist hoch, wenn es schwer verständlich ist. Sie ist ausserdem hoch, wenn das Lösen einer Aufgabe schwierig ist. Ein weiteres Kriterium ist die *Vorhersagbarkeit* des Verhaltens eines Systems. Ein System gilt als komplex, wenn über sein Verhalten nichts oder nur wenig ausgesagt werden kann. Vorhersagbarkeit und Verständlichkeit sind mindestens korreliert.

Casti [Casti 1994] nennt folgende Eigenschaften komplexer Systeme: Sie haben keine zentrale Steuerungseinheit, sondern bestehen aus vielen, kommunizierenden Einheiten. Weiterhin gibt es viele Rückkoppelungen innerhalb des Systems. Das wichtigste Merkmal aber ist die Beobachtung, dass ein komplexes System irreduzible ist, d.h. das System als Ganzes ist 'mehr' als die Summe aller Teile, es treten Emergenzphänomene auf.

Frese [Frese 1987] identifiziert folgende Gründe für hohe Komplexität eines Systems: Es gibt eine grosse Anzahl von Zielen, die mit diesem System erreicht werden sollen, von Plänen, wie vorgegangen werden kann, und von Signalen, die verarbeitet werden müssen. Dazu existieren bei einem System mit hoher Komplexität viele Beziehungen innerhalb jeder der drei Gruppen und zwischen den Gruppen. Schliesslich hat ein solches System eine Vielzahl bedingter Beziehungen, z. B. eine Aktion kann eine von vielen möglichen Reaktionen hervorrufen oder sie ist nur möglich, wenn auch andere Aktionen durchgeführt werden.

Basis einer Diskussion von Komplexität ist die Aufteilung eines *Kontextes* in interagierende Systeme. Die *Betrachtung* und Analyse eines Systems ist somit die Interaktion oder auch die Wechselwirkung eines Systems mit einem anderen. Durch die Gleichsetzung von Betrachtung und Interaktion wird ersichtlich, dass der Vorgang der Betrachtung im Normalfall beide Systeme verändert.

Komplexität hängt sehr stark vom Betrachter ab, dabei spielt der Vorgang der Betrachtung selbst eine wichtige Rolle. Was wird aufgenommen, wie wird es aufgenommen usw. Wichtig ist ebenfalls der Kontext des betrachteten Systems, da die Betrachtung im allgemeinen vom globalen Kontext abhängt, sowie auch von der Repräsentation und der Wahrnehmung des Kontextes im Betrachtersystem. In diesem Sinne ist Komplexität ein sehr *subjektives* Mass, verschieden für jedes System.

Es ist deshalb wünschenswert das Mass der Komplexität zu *objektivieren*, indem mehrere Systeme die Betrachtung und damit die Betrachtungsdaten, sowie die Interpretation der Betrachtungsdaten, in gleicher Weise durchführen, und somit auf das gleiche Resultat kommen.

Also gibt es für verschiedene Gebiete unterschiedliche Masse für Komplexität. Physiker versuchen, möglichst fundamentale Masse für Komplexität zu finden. Mathematiker interessieren sich eher für approximative Komplexitätsmasse: wie verhält sich z. B. die Berechnungszeit im Grenzübergang nach Unendlich. Komplexität in der Biologie bezieht sich auf Diversifikation und evolutionäre Prozesse. Psychologen legen ihr Augenmerk auf den Beobachter Mensch, was heisst Komplexität bei einem Menschen, wann nimmt ein Mensch ein System als komplex war und wann nicht.

Um dem intuitiven Mass der Komplexität, welches mit *Schwierigkeit* und *Unverständlichkeit* zu tun hat, auf die Spur zu kommen, betrachten wir kurz die beiden Masse Schwierigkeit und Unverständlichkeit.

Schwierigkeit und Unverständlichkeit sind sehr *subjektive* Masse, d.h. sie sind abhängig von einer Versuchsperson. Weiterhin werden sie als dimensionale Grössen behandelt. Unverständlichkeit reicht dabei von trivial über verständlich und schwer verständlich zu unverständlich, Schwierigkeit von einfach über angemessen und schwierig zu unmöglich.

Unverständlichkeit bezeichnet das Mass der Unfähigkeit einer Versuchsperson einen Sachverhalt über ein System in ihr *mentales Modell* zu integrieren oder in Übereinstimmung zu bringen. Es gibt zwei grundsätzlich verschiedene Möglichkeiten, dass dies überhaupt nicht gelingt: Im ersten Fall ist das mentale Modell der Versuchsperson zu inkompatibel mit dem Sachverhalt. Es benötigt noch einen grossen Lernaufwand, bis dieser verstanden werden kann. Im zweiten Fall handelt es sich um eine prinzipielle Unfähigkeit, sowie es zum Beispiel der Touringmaschine unmöglich ist, das Halteproblem zu verstehen (d.h. zu berechnen).

Schwierigkeit ist aus mehreren Teilbegriffen zusammengesetzt. Zum grossen Teil wird Schwierigkeit durch die Unverständlichkeit definiert. In zweiter Linie kommen auch Koordinations- und körperliche Tätigkeiten zur Geltung. Schwierigkeit enthält implizit die Forderung nach einem Ziel, einer Aufgabe. Wie schwer fällt es einer Versuchsperson etwas mit einem System zu tun. Je grösser die Unverständlichkeit, desto grösser ist auch die Schwierigkeit die Aufgabe zu lösen

Somit lässt sich intuitiv Komplexität mit der Schwierigkeit einer Aufgabe oder der Unverständlichkeit eines Sachverhaltes identifizieren.

Frese [Frese 1994] differenziert zwischen *Kontrolle* und *Kompliziertheit* als anderer Ansatz für *Unverständlichkeit* und *Schwierigkeit*.

Eine Person hat *Kontrolle*, falls sie in Übereinstimmung mit einem höheren Ziel Einfluss auf die Bedingungen sowie auf ihre eigene Aktivität nehmen kann. Dabei unterscheidet er *potentielle* Kontrolle und *tatsächliche* Kontrolle. Kontrolle wird durch interne (mentales Modell, Fähigkeiten) und äussere Gegebenheiten (Entscheidungsmöglichkeiten) bestimmt. Diese Gegebenheiten werden stark durch die Funktionalität, die Transparenz und die Vorhersagbarkeit des Systems beeinflusst.

Kompliziert ist nun ein System dann, wenn es schwierig zu kontrollieren ist. D.h. es ist komplex und zusätzlich trifft mindestens eine der folgenden Bedingungen zu: kleine Funktionalität, grosse Intransparenz, grosse Unvorhersagbarkeit, zu wenige Entscheidungsmöglichkeiten, zu viele Entscheidungsmöglichkeiten für die Fähigkeiten oder das mentale Modell, oder die Komplexität ist nicht nötig oder angemessen. Kompliziertheit ist also die *übermässige Komplexität*, die kaum mehr zu kontrollieren ist.

Für die Gestaltung von Software fordert Frese folgende Richtlinien: Kontrolle sollte maximiert, Komplexität optimiert und Kompliziertheit minimiert werden.

2. Komplexitätsmasse und Ansätze

2.1. Axiomatische Komplexitätstheorie, Berechnungskomplexität

Die wichtigsten mathematischen Komplexitätsmasse werden in der *axiomatischen Komplexitätstheorie* behandelt. Basis für alle diese Komplexitätsbetrachtungen ist die Touringmaschine. Mit Hilfe der Touringmaschine können verschiedene Komplexitätsmasse eingeführt werden, die alle eine gemeinsame mathematische Theorie haben.

Ein *Problem* L ist definiert als eine Teilmenge der natürlichen Zahlen. Eine *Problemstellung* ist die Frage, ob eine bestimmte Zahl w in L enthalten ist. Es ist möglich, dass diese Frage nicht *entscheidbar* ist, d.h. die Touringmaschine beendet die Berechnung nicht.

Interessant ist nun, welche Komplexität ein Problem oder eine Problemstellung hat. Dabei unterscheidet man unter anderem folgende Masse: Die *Bandkomplexität*, d.h. wieviel Band der Touringmaschine wurde belegt, die *Zeitkomplexität*, d.h. wieviele Schritte hat die Touringmaschine benötigt um zu enden, und die *Umkehrkomplexität*, wie oft hat der Lese-Schreibkopf der Touringmaschine die Richtung gewechselt. Es ist leicht ersichtlich, dass die Zeitkomplexität immer grösser oder gleich der Bandkomplexität sein muss, da pro Zeiteinheit auf der Touringmaschine maximal eine Bandeinheit gebraucht werden kann. Das gleiche gilt auch für die Umkehrkomplexität. Diese drei Masse gehören alle derselben Klasse von Komplexitätsmassen an, und haben eine einheitliche Basis, die axiomatische Komplexitätstheorie.

Mit diesen Komplexitätsmassen lässt sich die Menge der Probleme in klar definierte Teilmengen zerlegen. Im Falle der Zeitkomplexität sind dies z.B.: *P*, mit polynominalem Aufwand berechenbar oder *NP*, auf der nicht deterministischen Touringmaschine mit polynominalem Aufwand berechenbar. *Polynominaler Aufwand* bedeutet, dass es ein Polynom p(t) gibt, welches für jede Eingabe w die obere Grenze für den Aufwand ist. Die Zeit T(L, w), die die Touringmaschine benötigt, um das Problem L bei der Eingabe w zu berechnen, ist für alle w durch p(|w|) begrenzt. Weiterhin gibt es auch die Klassen der mit exponentieller Zeit berechenbaren Problemen: 1-Exptime, 2-Exptime usw. (vgl. [Schöning 1985]).

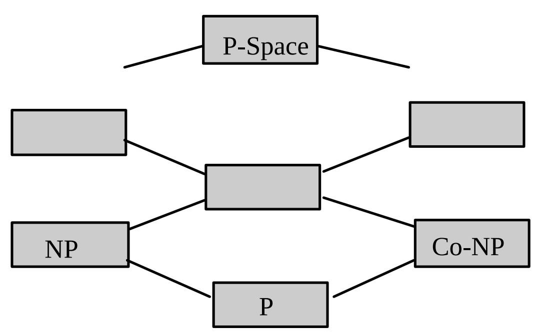

Abbildung 1: Polynominale Hierarchie

Die Komplexitätsklassen bilden eine Hierarchie, wie sie zum Beispiel in [Schöning 1988], [Schöning 1985] beschrieben wird und auf Stockmayer [Stockmayer 1977] zurück geht. In Abbildung 1 ist die Struktur und die allerwichtigsten Klassen der polynominalen Hierarchie dargestellt. Dabei bedeutet eine Verbindungslinie, dass die tiefer gelegene Menge eine Teilmenge der höher gelegenen ist. Die *polynominale Hierarchie* ist ein Teil der *exponentiellen Hierarchie*. Die exponentielle Hierarchie wird durch die mit *primitiv-rekursiven* Funktionen berechenbaren Probleme eingeschlossen. Um diese sind dann die mit *rekursiven*, dann die mit *partiell-rekursiven* berechenbare Probleme und schliesslich, als äußerstes Gefäss der Rekursionstheorie, die Menge aller Funktionen auf natürlichen Zahlen (Vgl. dazu [Engeler/Läuchli 1992]). Dabei bedeutet jeder Aufstieg in der Hierarchie einen Aufstieg in eine höhere Komplexitätsstufe. Die Echtheit der Inklusionen in der Komplexitätshierarchie ist in den meisten Fällen weder bewiesen noch widerlegt.

Aus der Komplexitätstheorie geht als wichtiger Begriff der Begriff *machbar* hervor. Ein Problem ist dann mit einem Computer in brauchbarer Zeit machbar, wenn es mit polynominalem Aufwand berechenbar ist, dies sind nur die Probleme der Klasse P. Die Resultate dieser mathematischen Theorie haben somit direkten Einfluss auf andere Wissenschaften. Denn sobald ein Problem nicht mehr aus P stammt, gilt es bei heutigem Wissen als nur mit exponentieller Zeit berechenbar (auch auf realen Computern), es ist nicht mehr machbar. Bei solchen Problem müssen somit ganz neue Strategien eingesetzt werden, um es zu lösen oder um eine möglichst gute Lösung zu finden.

Der Begriff machbar wird in der Praxis z. B. durch die *Schiebeinstruktion* abgeschwächt. Reale Computer können nämlich, im Gegensatz zur Touringmaschine, 2^n mit linearem Aufwand berechnen. Dies wird durch n-maliges Schieben der Bits in einem Register erreicht. Die Schiebeoperation ist dabei überhaupt eine der einfachsten Operation, da sie in der Grundform ohne Gatter auskommt. Die Touringmaschine benötigt hingegen exponentiellen Aufwand, da sie eine Variable in jedem Schritt um maximal 1 erhöhen kann und somit mindestens 2^n Schritte benötigt.

Ein weiterer Punkt, der den Begriff machbar in der Praxis relativiert, ist die Begrenztheit heutiger Computer. Speicher und Berechnungszeit sind klein. Die Touringmaschine kann hingegen eine beliebig grosse endliche Zahl von Variablen speichern, wobei jede Variable eine beliebige natürliche Zahl enthalten kann. Auch die Berechnungszeit kann beliebig gross werden, solange sie endlich bleibt.

2.2. Algorithmische Komplexität

2.2.1. Allgemeine Definition

Ähnlich der Komplexitätstheorie geht die *algorithmische Komplexität* von der Touringmaschine aus. Sie definiert die Komplexität eines Problems als die Länge der kürzesten Touringmaschine, die dieses Problem lösen kann. Daraus lässt sich insbesondere eine Definition für Zufallsfolgen gewinnen. Eine Folge ist dann *algorithmisch zufällig*, wenn das kürzeste Programm für diese Folge die Folge selbst enthält. Es lässt sich zeigen, dass der grösste Teil aller Folgen algorithmisch zufällig ist.

Bei diesen ganzen Definitionen schleicht sich aber das Problem der Unentscheidbarkeit ein. Der Satz von Rice [Engeler/Läuchli 1992] besagt nämlich, dass eine Menge von Funktionen (Programmen), die nicht leer ist und nicht alle Funktionen enthält, unentscheidbar ist. D.h. also, dass es keine Touringmaschine gibt, die besagt, ob eine bestimmte Funktion in dieser Menge enthalten ist oder nicht. Einer Touringmaschine ist es deshalb nicht möglich herauszufinden, ob ein Programm das kürzeste ist oder nicht.

Die algorithmische Komplexität deckt sich nicht unbedingt mit dem intuitiven Begriff von Komplexität. So ist eine von einem Affen getippte Sequenz zufälliger Buchstaben algorithmisch komplexer als zum Beispiel Goethes 'Faust'.

2.2.2. Definitionen und Theoreme

Definitionen und Theoreme aus [Chaitin 1974].

Def. 1: Ein *Computer* sei eine partiell rekursive Funktion C(p). Das Argument p (das *Programm*) sei ein binäres Wort. C(p) berechne ein binäres Wort w in Abhängigkeit des Programmes p. C(p) sei undefiniert, falls das Programm p nie terminiert.

Def. 2: Die *Komplexität* $I_C(w)$ eines binären Wortes sei die Länge des kürzesten Programmes p, welches auf dem Computer C die Ausgabe w erzeugt: $I_C(w)=\min_{C(p)=w}\text{len}(p)$.

Falls ein Programm keine Ausgabe erzeugt, zum Beispiel weil die Berechnung nie enden wird, so sei $I_C(p)$ unendlich gross.

Def. 3: Ein Computer U heisst *universell*, falls für alle Computer C und alle binären Wörter w gilt: $I_U(w) \leq I_C(w) + c$, wobei die Konstante c nur vom Computer C abhängt. Da ein solcher Computer U existiert, misst man die Komplexität eines Wortes w mit diesem Computer und schreibt I(w).

Satz 1: **a)** Es gibt eine Konstante c, so dass für alle w gilt: $I(w) \leq len(w) + c$. **b)** Es gibt weniger als 2^n binäre Wörter der Komplexität kleiner n.

These 1: Ein Wort w ist dann *zufällig*, wenn I(w) ungefähr gleich len(w). Es lässt sich zeigen, dass fast alle Wörter zufällig sind (Vgl. auch [Abu-Mostafa 1988])

2.2.3. Einführen einer Intelligenz

Stellen wir das von Chaitin aufgestellte System etwas um: Wir besitzen einen Computer wie er in Kapitel 2.2.2 definiert ist. Unser System enthält nun zusätzlich eine *Intelligenz* Q, welche zu einem binären Wort w als Eingabe ein binäres Wort p berechnet, also Q(w)=p, so dass $len(p) \leq len(w) + c$ und C(p)=w. Dabei entspricht len(w) + c der in Satz 1a) definierten oberen Grenze.

Dieses System lässt sich folgendermassen interpretieren: Eine intelligentes Wesen nimmt ein binäres Wort wahr. Ein Wort erscheint diesem Wesen *zufällig*, wenn es kein Programm konstruieren kann, welches kürzer ist als die Länge des wahrgenommen Wortes (Vgl. These 1). Für dieses Versagen des Wesen gibt es zwei mögliche Ursachen: Entweder ist das Wesen zu dumm gewesen, um dieses Programm zu konstruieren oder aber es existiert kein solches Programm.

Betrachten wir den Vorgang der Wahrnehmung etwas genauer: Als Input erhält das Wesen ein binäres Wort w. Auf dieses Wort wird eine Funktion f angewandt, welche wiederum ein binäres Wort w_q erzeugt. Die Funktion f kann dabei auf einen grossen Speicher zugreifen, welcher ein binäres Wort q enthält (das *mentale Modell* von Q).

Die Funktion f sei folgendermassen aufgebaut: $f(w_i) := \begin{cases} 1, w_i = q_i \\ 0, w_i \neq q_i \end{cases}$

Ist nun q ein Folge von Einsen, dann ist w_q=w, dies entspricht dem Fall, den Chaitin betrachtet. Ist q eine Folge von Nullen, dann wird w_q invertiert: w_q=INV(w). Interessant ist der Fall, bei welchem q eine zufällige Folge nach These 1 aus 2.2.2 ist. Falls w genau dieses Folge ist, wird w_q eine Folge von Einsen sein, d.h. das wahrgenommene Wort w_q hat minimale Komplexität $I(w_q)$. Ausserdem wird das Wort aus lauter Einsen in eine zufälliges Wort übersetzt, welches für die Intelligenz unverständlich wird. Die Eigenschaft aus These 1 beschreibt in diesem Sinne also nicht die Zufälligkeit, sondern die *Verständlichkeit*, d.h. in welchem Fall ein wahrgenommenes Wort unverständlich, weil nicht reduzierbar, ist. Dabei lässt sich die Komplexität des mentalen Modells ebenfalls aus der algorithmischen Komplexität berechnen: I(q). Somit ist die Komplexität des wahrgenommenen Wortes abhängig von der Komplexität des mentalen Modells und der Komplexität der Umgebung.

Bemerkung: Die Funktion f, die vorgeschlagen wurde, ist invertierbar. Die Anzahl Wörter w, die verstanden werden können, ist für jedes mentale Modell q gleich.

2.3. Logische Tiefe

Die *logische Tiefe* berechnet die Komplexität eines Problems. Gegeben ein Problem, dann ist dessen Komplexität die Zeit, welche das kürzeste Programm benötigt, um es zu lösen.

Die logische Tiefe ist nahe verwandt mit der Berechnungskomplexität. Die Berechnungskomplexität betrachtet das schnellste Programm, wogegen die logische Tiefe das kürzeste betrachtet.

2.3.1. Tiefe von endlichen Wörtern

Dieses Kapitel fasst die Ideen von Bennett aus [Bennett 1988] zusammen.

Bennett benötigt eine *universelle Touringmaschine* U(p,w), wie sie auch Chaitin bei der Definition der algorithmischen Komplexität (vgl. 2.2.2) eingeführt hat. Die Grösse p ist ein *Programm*, codiert in einem binären Wort, und w ist die Eingabe des Programmes p, ebenfalls codiert als binäres Wort. Falls der Input das leere Wort ist, so wird die universelle Touringmaschine mit U(p) notiert. T(p,w) misst die Zeit, die die Touringmaschine U benötigt, um das Programm p mit dem Eingabewort w zu berechnen. T(p) berechnet dasselbe beim leeren Wort als Argument. T(p,w) und U(p,w) sind undefiniert, falls U die Abarbeitung von p inkorrekt oder gar nicht beendet. Weiter ist U *genügend universell*, d.h. U kann jede andere Maschine mit linearem Wachstum der Ausführungszeit und konstanter Änderung der Programmgrösse simulieren.

Def. 1: Das *minimale Programm* x* zur Berechnung ist x sei min {p:U(p)=x}. Das minimale Programm (x|w)* für x relativ zu w ist min {p:U(p,w)=x}.

Bem. 1: Jedes Wort x hat auch ein *Printprogramm* der Länge |x|+O(log|x|). Die Grösse log|n| muss hinzugefügt werden, da einer Printanweisung innerhalb des Programmes auch die Länge des auszugebenden Wortes mitgegeben werden muss.

Bem. 2: (x|w)* kann sehr viel kürzer als x* sein, aber nur um einen konstanten Wert länger sein. Falls nämlich w stören würde, genügte es, w vom Band der Touringmaschine zu entfernen, und dann x* durchzuführen.

Def. 2: *s-Komprimierbarkeit*: Ein Wort x sei um s Bits komprimierbar, falls |x*|≤|x|-s.

Def. 3: $D_S(x)$ bezeichne die *Tiefe* eines endlichen Wortes: Gegeben Wörter x und w, sowie ein *Signifikanzlevel* s. Dann sei $D_S(x)$ das Minimum von {T(p): |p|-|p*|<s und U(p)=x}.

Bem. 4: Der *Signifikanzlevel* s bei der Definition der Tiefe wird von Bennett eingeführt, da ein Programm, das ein klein wenig grösser als ein anderes ist, unter Umständen sehr viel schneller als dieses ist. Also kleine Änderungen in x können zu grossen Änderungen in D(x) führen. Um diese Instabilität zu verkleinern, wird der Signifikanzlevel eingeführt.

Def. 4: x wird *t-tief* genannt, wenn $D_S(x)$ grösser t ist, und *t-seicht*, falls $D_S(x)$<t.

Def. 5: die *relative Tiefe* $D_S(x,w)$ eines Wortes x bzgl. w: $D_S(x,w)$ sei das Minimum von {T(p,w): |p|-|(p|w)*|<s und U(p,w)=x}.

Satz 1: Gesetz des langsamen Wachstums. Dieser Satz besagt, dass es nicht möglich ist, aus einem seichten Wort mittels eines schnellen Programmes ein tiefes zu machen. Die mathematisch genaue Definition findet sich in [Bennett 1988]. Der gleiche Satz gilt auch für die relative Grösse bzgl. eines weiteren Wort w.

Bem. 5: Die relative Tiefe ist nicht transitiv. Falls x seicht relativ zu w ist, und y seicht relativ zu x ist, dann folgt nicht, dass y seicht relativ zu w ist.

Die Tiefe, so wie sie bisher definiert ist, befriedigt in der Hinsicht nicht, dass sie von einem Signifikanzparameter s abhängt und nicht nur von dem gemessenen Wort x. Dieser Unschönheit kann Abhilfe verschafft werden, indem über alle Programme, die x erzeugen, gemittelt wird. Bennett definiert $D_{rmr}(x)$ als den reziproken Mittel der reziproken Tiefe:

$$D_{rmr}(x) = \frac{\sum \{2^{-|p|} : U(p) = x\}}{\sum \{\frac{2^{-|p|}}{T(p)} : U(p) = x\}}$$

2.4. Deterministische Entropie

Das schwache Komplexitätsmass der *deterministischen Entropie* ist aus der Mustererkennung motiviert. Dort tritt häufig der Fall auf, dass das zu erkennende Muster selten und die restlichen Muster häufig auftreten. Die Ideen sind [Abu-Mostafa 1988] entnommen.

Gegeben eine Funktion f: $\{0,1\}^N$ _ $\{0,1\}$. Dann sind die Mengen $f^{-1}(1)=\{x \mid f(x)=1\}$ und $f^{-1}(0)=\{x \mid f(x)=0\}$ die beiden Inversen dieser Funktion. Sei $h(f) = \min\{ |f^{-1}(0)|, |f^{-1}(1)| \}$, dann definiert Abu-Mostafa die *deterministische Entropie* mit $H(f)=\log_2(1+h(f))$ [Bits].

Interessant ist die Beziehung, die Abu-Mostafa zwischen der *algorithmischen Komplexität* $R(f)=\log_2(I_C)$, der *Schaltungstiefe* C(f) und der deterministischen Entropie H(f) zeigt. Bei Vernachlässigungen von Termen o(N) gilt nämlich die folgende Ungleichung: $R(f) \leq C(f) \leq H(f)$.

Für die Definition von $I_C(f)$ vergleiche 2.2.2, für C(f) vergleiche 2.5.2.

Anmerkung: Nicht alle Funktionen, die auf einer Touringmaschine berechenbar sind, können mit einer Bool‘schen Schaltung modelliert werden.

2.5. Schaltungstiefe

Die Schaltungstiefe ist ein Mass für die Komplexität einer Bool‘schen Funktion. Sie besagt, wieviele Stufen von Gattern eine Funktion bei ihrer kleinsten Realisierung benötigt (nur mittels Gatter mit zwei Eingängen und einem Ausgang).

Wie die algorithmischen Komplexität, benötigt auch die Schaltungstiefe die kleinste Repräsentation einer Funktion. Auch hier stellt sich somit das Problem der Unentscheidbarkeit.

2.5.1. nach Karchmer

Diese Definitionen und Sätze sind aus [Karchmer 1988].

Def. 1: *Bool‘sche Schaltung*: gerichteter azyklischer Graph, bestehend aus Inputknoten (x_i und $\neg x_i$), Gates (zwei Eingänge) und einem Endknoten (ein Ausgang). Ein Gate repräsentiert die Bool'schen Funktionen AND oder OR. Ein solche Schaltung kann jede Bool‘sche Funktion berechnen.

Def. 2: Die *Tiefe* d(C) einer Bool‘schen Schaltung C ist die grösste Distanz von Eingabeknoten zu Ausgabeknoten. Die *Grösse* s(C) ist die Anzahl von Kanten dieses Graphen.

Def. 3: Nun kann die *Tiefe d(f) einer Funktion* f berechnet werden als die kleinste Tiefe aller Graphen, die diese Funktion repräsentieren. Die *Grösse s(f) einer Funktion* f kann definiert werden, als die Grösse des kleinsten Graphen für diese Funktion.

Satz 1: Die Tiefe und die Grösse einer Funktion sind äquivalent (bis auf Konstanten und Skalierung)

Für weitere Betrachtung vergleiche auch [Schöning 1985], wo die Definition auf Sprachen ausgedehnt wird (unendliche Objekte).

2.5.2. nach Abu-Mostafa

Diese Definition ist aus [Abu-Mostafa 1988] entnommen.

Gegeben ein *universelles Gatter* mit n Eingängen und einem Ausgang, das eine gewünschte Funktion f simuliert. Eine *kombinatorische Schaltung* ist aus einer beliebigen endlichen Zahl von solchen Gattern zusammengesetzt, wobei keine Zyklen existieren dürfen. Die *Kosten* eines Gatters mit n Eingängen sei 2^n, bei mehreren Gattern seien die *gesamten Kosten* die Summe der Kosten der einzelnen Gatter. Eine Schaltung S besteht nun aus solchen Gattern mit Verbindungslinien und simuliert eine Funktion F.

Def. 1: Die *Schaltungskomplexität* einer Funktion F: C(F)=min {Kosten von S: S simuliert F}.

2.6. Komplexität der Kommunikation

Die Komplexität einer Kommunikation zwischen zwei Partnern ist definiert durch der Anzahl hin und her gesandter Bits. Die Partner müssen sich ausgehend von zwei unterschiedlichen Positionen auf eine der möglichen Lösungen einigen. Es werden die Bits gezählt, die im besten, d.h. minimalen, Fall gesendet werden müssen. Die generelle Beobachtung ist, dass je ungeordneter das Problem ist, desto mehr Bits müssen kommuniziert werden und desto komplexer ist die Kommunikation. Komplexität und Unordnung sind verwandt.

2.6.1. Definitionen und Theoreme

Die in diesem Kapitel dargestellten Ideen, Definitionen und Sätze sind [Karchmer 1988] entnommen.

Gegeben sind zwei *Partner* A und B, die mit fehlerlosen, binärem Kanal kommunizieren. Die Kommunikation hat ein deterministisches Protokoll und nur die Geschichte der Kommunikation bestimmt, welcher Partner an der Reihe ist mit Senden. Gegeben sind drei endliche Mengen X, Y, Z und eine Relation $R \subseteq X \times Y \times Z$.

Def. 1: $S(R) \subseteq X \times Y$ ist der *Support* von R falls für jedes Paar $(x,y) \in S$ ein Trippel $(x,y,z) \in R$ existiert.

Folgendes Spiel wird nun gespielt: $x \in X$ wird A und $y \in Y$ wird B zugeteilt, wobei $(x,y) \in S(R)$. Die Aufgabe der Partner ist die Festlegung eines gemeinsamen $z \in Z$ so dass $(x, y, z) \in R$ erfüllt.

Def. 2: D sei ein *Protokoll* für obiges Spiel, dann ist D(x,y) die Anzahl kommunizierter Bits, wenn die Spieler D befolgen. a(x,y) ist die *Geschichte* von D mit (x,y).

Def. 3: Die *Komplexität* der Kommunikation einer Relation C(R) ist das Minimum über alle Protokolle D vom Maximum über alle Paare (x,y) von S(R) der Anzahl kommunizierten Bits D(x,y). $C(R) = \underset{D}{Min}(\underset{(x,y)\in S(R)}{Max}(D(x,y)))$

Karchmer beweist, dass die Komplexität der Kommunikation einer Relation äquivalent der Schaltungstiefe (Vgl. 2.5.1) ist. D.h., für jede Funktion f: $\{0,1\}^n \rightarrow \{0,1\}$ gilt: $d(f)=C(R_f)$. R_f ist eine Relation auf den drei Mengen $f^{-1}(0)$, $f^{-1}(1)$ und {0, .., n}, und nicht mehr auf beliebigen Mengen X, Y und Z. Dabei sind $f^{-1}(1)=\{x \mid f(x)=1\}$ und $f^{-1}(0)=\{x \mid f(x)=0\}$ die Inversen zur Funktion f.

Berechenbarkeit kann mit verschiedenen Gerätetypen erreicht werden, z. B: Akzeptierende Geräte (Touringmaschine), generierende Geräte (Grammatiken) und Geräte zum Trennen von Wörtern und Nichtwörtern (Bool'sche Schaltung). Die Bool'sche Schaltung trennt die Menge aller Eingabewörter in die Mengen $f^{-1}(0)$ und $f^{-1}(1)$ auf. Karchmer identifiziert die folgende Basis von Komplexität: Die Funktion f wird um so komplexer, je unorganisierter die Mengen $f^{-1}(0)$ und $f^{-1}(1)$ sind. Je unorganisierter diese nämlich sind, desto grösser muss die Geschichte a(x,y) werden, damit sich die Partner einigen können.

2.7. Bedingte Komplexität

Das Konzept der bedingten Komplexität ist aus [Crutchfield, Young 1989] entnommen.

Die Komplexität eines dynamischen Systems kann durch Symmetrie beschrieben werden. Dabei gibt es die gewöhnliche Symmetrie, d.h. wiederholte Strukturen, aber auch statistische Regularität. Ein System ist komplex, falls es aus vielen Symmetrien zusammengesetzt ist.

Gegeben sei die *Modellbasis* $\{B_t, P_t\}$, B_t repräsentiert einen idealen Münzwurf, der alle t Sekunden durchgeführt wird (Bernouilli-Fluss), P_t ist ein vollständig vorhersagbarer Prozess, welcher sich alle t Schritte wiederholt. Hiermit kann die Bernouilli-Touringmaschine (BTM) definiert werden, die zusätzlich zur normalen Touringmaschine noch ein Zufallsregister enthält. Die Menge von Symmetrien S zerlegt ein System, d.h. die Daten D dieses, in Äquivalenzklassen.

Def. 1:Die *Komplexität* C(D|S) ist der Shannon'sche Informationsgehalt einer durch S entstandenen Äquivalenzklasse plus die Datenmenge, welche durch S nicht klassifiziert wurde.

Falls eine Numerierung der Symmetrien existieren würde, könnte die *absolute Komplexität* C(D) als das Infinum über alle Symmetrien der bedingten Komplexität definiert werden.

Die Bedingte Komplexität genügt einer wichtigen Bedingung, sie ist klein für total

geordnete und total ungeordnete Daten und gross irgendwo in der Mitte.

2.8. Thermodynamische Tiefe

Die *thermodynamische Tiefe* ist ähnlich der logischen Tiefe. Sie ist aber nicht mathematisch motiviert, sondern stammt aus der Physik. Lloyd und Pagels definieren die thermodynamische Tiefe (Vgl. [Norretranders 1994]) als die Menge von Information (nach Shannon) die beim tatsächlichen Entstehungs- und Entwicklungsprozess eines Objektes (oder Systems) aussondiert wurde (*Exformation*).

Dynamische Systeme reichen in ihrem Ordnungsgrad von völliger Ordnung bis zu völliger Unordnung. Lloyd und Pagels erkannten, dass Komplexität irgendwo in der Mitte am höchsten ist. Weiter ist zum Beispiel die Komplexität von zwei Hühnern nicht zweimal so gross wie die Komplexität von einem Huhn, denn zur Herstellung des ersten benötigt man einen grossen Verarbeitungsaufwand, wogegen das zweite Huhn ganz einfach kopiert werden kann. Lloyd und Pagels folgerten daraus, dass die Komplexität eine Funktion desjenigen Prozesses ist, der das System erzeugt hat.

2.8.1. Mathematisch formuliert

Gegeben seien experimentelle Daten über Zustände und die Übergänge zwischen den Zuständen, die zu dem zu messenden Endzustand führen können. Die Daten geben insbesondere Auskunft, mit welcher Wahrscheinlichkeit die Zustandsübergänge benutzt werden. Die folgenden Definitionen und Sätze sind alle aus [Lloyd, Pagels 1988].

Gegeben seien ein *Makrozustand* d und eine *Sequenz von Zuständen* $a_i b_j \ldots c_k$. $p(a_i b_j \ldots c_k d)$ bezeichne die experimentell bestimmte Wahrscheinlichkeit, dass diese Zustände durchlaufen worden sind, um am Zustand d anzukommen. Die Summe der Wahrscheinlichkeiten über alle solche Sequenzen, die mit d enden sei p(d), d.h. die Wahrscheinlichkeit, dass der Zustand d überhaupt erreicht wurde. Daraus folgt dass $p(a_i b_j \ldots c_k | d)$ durch $p(a_i b_j \ldots c_k d)/p(d)$ berechnet werden kann.

Def. 1: Die *mittlere Tiefe*: $S(d) = -k * \sum p(a_i b_j \ldots c_k | d) * \log(p(a_i b_j \ldots c_k | d))$

Def. 2: Die *Tiefe* D(d) eines tatsächlichen Überganges ist $-k * \ln p(a_i b_j \ldots c_k | d)$.

Def. 3: Falls die tatsächliche Entwicklung nicht bekannt ist, definieren wir die *absolute Tiefe* |D|(d) = min {D(d): über alle Übergänge zum Zustand d}.

Def. 4: Die *thermodynamische Tiefe* $D_T(d)$ ist definiert als die Differenz zwischen der *grobkörnigen Entropie* eines Zustandes d minus der *feinkörnigen* dieses Zustandes: $D_T(d)=S(d) - S_0(d)$.

Diese Definition werden von Lloyd und Pagels in die klassische Physik und die Quantenphysik übertragen. Dort wird auch eine exaktere Definition der thermodynamischen Tiefe $D_T(d)$ gegeben. Die *makroskopischen Zustände* eines Systems sind definiert durch die Punkte der Interaktion eines Systems mit einem Messgerät. Die Definitionen und Theoreme für diese Anwendungsgebiete stehen in [Lloyd, Pagels 1988] und sind Präzisierungen der in diesem Kapitel vorgestellten Formeln. Bei der Übertragung in die Thermodynamik zeigt sich, dass die Tiefe eines dynamischen Systems gleich der thermodynamischen Tiefe ist, falls die Konstante k in der Definition der Tiefe mit der Boltzmannkonstante k_B ersetzt wird.

Satz 1: Die Funktion f in der Definition der mittleren Komplexität ist die einzige, welche folgenden drei Punkten genügt:

(1) Sie muss eine kontinuierliche Funktion der Wahrscheinlichkeiten der Übergänge zu dem Zustand d sein.

(2) Falls alle n Übergänge die gleiche Wahrscheinlichkeit haben, ist das Mass monoton steigend mit grösser werdendem n. D.h. eine genauere Beschreibung eines Prozesses kann die Komplexität nicht verkleinern.

(3) Additivität: Die Komplexität des Überganges b - c - d muss gleich sein wie die Komplexität von b - c (nur für die Übergänge, die auch für nach d relevant sind) plus die Komplexität von c - d.

2.8.2. Diskussion der thermodynamischen Tiefe

Lloyd und Pagels diskutieren in ihrem Artikel [Lloyd, Pagels 1988] auch die Verwandtschaft der thermodynamischen Tiefe mit anderen gängigen Komplexitätsmassen.

Für die *Berechnungskomplexität* eines Problems gilt, dass sie gleich der Tiefe der Ausgabe ist, welche ein Computer liefert, der dieses Problem löst. Die Tiefe und die Berechnungskomplexität sind äquivalent.

Die *algorithmische Komplexität* eines Wortes ist proportional zur minimalen Information, die ein Computer benötigt, um dieses Wort zu berechnen. Die absolute Tiefe eines Systems in einem Zustand ist proportional zur minimalen Information, die benötigt wird, um zu diesem Zustand zu gelangen. Die beiden Masse sind ähnlich, aber nur in unnatürlichen Gegebenheiten gleich (näheres dazu in [Lloyd, Pagels 1988]).

Die *logische Tiefe* ist gleich der Berechnungskomplexität des besten Ablaufes, das ist das algorithmisch wahrscheinlichste Programm. Die thermodynamische Tiefe ist gleich der Berechnungskomplexität für den wahrscheinlichsten Ablauf. Thermodynamische Tiefe und logische Tiefe sind nur gleich, wenn die Wahrscheinlichkeit eines Programmes gleich der algorithmischen Wahrscheinlichkeit ist.

Die thermodynamische Tiefe für die Berechnung auf einem Computer kann mit der *Abfallinformation* (*Exformation*) identifiziert werden, die bei der Berechnung anfällt. Also der Information, die vergessen werden muss. Die Abfallinformation berechnet sich aus der Anzahl kopierten Bits minus die brauchbare Information (Länge vom Resultat minus Länge von der Eingabe).

These 1: Lloyd und Pagels schliessen ihre Diskussion mit der Beziehung zwischen dem Alter des Universums und der Zufälligkeit von Zahlen: Falls eine Zahl einen Zustand eines physikalischen Systems repräsentiert, hat sie eine thermodynamische Tiefe. Seit der Entstehung des Universums ist nur eine endliche thermodynamische Tiefe generiert worden, es existieren aber unendlich viele Zahlen. D.h. es gibt unendlich viele Zahlen, die zu einer bestimmten Zeit mit einem physikalischen System nicht repräsentiert werden können. Es gibt also Zahlen, die eine kurze Beschreibung haben, deren thermodynamische Tiefe aber zu gross ist, als dass diese Beschreibung zur jetzigen Zeit gefunden werden könnte. Sie sind im Moment nicht unterscheidbar von zufälligen Zahlen.

2.9. Informationstheorie

Als wichtiger Bestandteil vieler Betrachtungen über Komplexität ist das Shannon'sche *Informationsmass*, [Shannon, Weaver 1949]. Es misst Information aus der Auftretenswahrscheinlichkeit eines Ereignisses.

Wie verschiedene Autoren bereits gezeigt haben (z. B. [Norretranders 1994]), entspricht dieses Informationsmass nicht der landläufigen Bedeutung von Information. Es ist vielmehr ein Mass für Überraschung (Vgl. auch 3.6).

Trotzdem wird der Informationsgehalt häufig mit der Komplexität eines Objektes gleichgesetzt.

2.9.1. Informationsbasierte Komplexität

Traub stellt in [Traub 1988] die wichtigsten Ideen der *informationsbasierten Komplexität* dar.

Traub geht von folgenden, grundlegenden Aussagen über *Information* aus: (1) die zu Verfügung stehende Information ist *unvollständig*, (2) sie ist *fehlerbehaftet* und (3) Information *kostet*. Information ist hier nicht im Sinne Shannons zu verstehen. Diese misst nämlich den Informationsgehalt einer Botschaft und nicht die Information, die in einer Botschaft enthalten ist. Information im weitesten Sinn ist, was wir über das zu lösendes Problem wissen.

Die Berechnungskomplexität misst die minimalen Kosten, die zur Lösung eines Problems führen. Dabei gibt es zwei Grundlagen: Die Information über das Problem ist komplett, korrekt und gratis oder nicht. Ersteres nennt Traub kombinatorische Komplexität.

Die informationsbasierte Komplexität ergibt sich aus den Kosten der Berechnungskomplexität in dem Fall, in welchem die Informationsbeschaffung zusätzliche Kosten verursacht. Sie berechnet sich also durch die Kosten der Informationsbeschaffung plus die Kosten der Berechnung. Das Berechnen der Komplexität kann somit in zwei Phasen zerlegt werden: die *Informationsphase* und die *kombinatorische Phase*.

Die zentrale Fragestellung ist, wie gross die Berechnungskomplexität sei, um einen bestimmten Level von Sicherheit zu erreichen.

Für tiefergehende Betrachtungen und einige angewandte Beispiele kann auf [Traub, Wasilikowski, Wozniaskowski 1988] zurück gegriffen werden. In diesem Buch wird die Theorie der Informationsbasierten Komplexität mathematisch genau definiert.

2.9.2. Ansätze aus der Biologie

Shimizu definiert in [Shimizu 1989] die Komplexität eines Biosystems durch die unterscheidbaren Relationen unter den Elementen dieses Biosysteme.

Eine andere, ebenfalls strukturelle Komplexität, wird durch die Anzahl von Arten definiert, die in einem bestimmten Biosystem leben, die Diversität (Vgl. [Johnson 1988]). Johnson definiert die Diversität mit der gleichen Formel, wie Maxwell und Shannon ihre Entropie definiert haben: $S = -k\sum p_i \log(p_i)$. Dabei ist p_i die Wahrscheinlichkeit, eines idealisierten physikalischen Systems ist, im Zustand i (von n möglichen) zu sein.

Wiley [Wiley 1988] beschreibt eine Definition der Komplexität eines Systems (nach [Layzer 1977]) mittels der beobachteten Entropie: H_{obs}. Damit kann die makroskopische Information definiert werden: $I=H_{max} - H_{obs}$. Wobei H_{max} die Entropie im Gleichgewicht bezeichne.

In der gleichen Richtung wie die thermodynamische Tiefe ist das von Wicken [Wicken 1989] benutzte Mass für Komplexität: Er definiert die Komplexität einer DNA-Sequenz als die benötigte Information (im Sinne Shannons), die für die Erstellung dieser Sequenz nötig ist. Die genetische Komplexität ist proportional zur genetischen Information, die während des Selektionsprozesses ausprobiert und verworfen wurde ([Lloyd Pagels 1988] nach [Kuhn, Waser 1983]). Diese Definition der Komplexität ist abgedeckt durch die thermodynamische Tiefe.

2.9.3. Messen der Gehirnaktivität

Die Komplexität des Denkvorganges lässt sich (vgl. [Norretranders 1994]) durch Messen der Gehirnaktivität bestimmen, d.h. durch Messung der Hirndurchblutung bis in die kleinsten Verästelungen. Dabei konnten verschiedene Zentren des Gehirnes für verschiedene Aufgaben lokalisiert werden. Die erhöhte Hirndurchblutung wird für den Wegtransport von Abfallstoffen benötigt, nicht für die Aktivität selbst.

Je nach Aufgabe des Gehirnes werden andere Teile beim Denken und Wahrnehmen aktiviert. Somit könnte die Komplexität eines Problems für einen Menschen zu einem bestimmten Zeitpunkt als die gemessene Gehirnaktivität definiert werden.

Interessant ist, dass praktisch alle Gehirnteile aktiv werden, wenn man versucht zu verstehen, was ein rückwärts laufendes Tonband zu sagen hat.

2.10. Komplexität eines Systems

Frese [Frese 1987] definiert die Komplexität eines Systems. Dabei geht er von zwei interagierenden Systemen aus. Er betrachtet die folgenden Elementgruppen dieser Interaktion: (a) die Ziele und Pläne des Betrachtersystems, sowie die Signale des betrachteten, (b) die Beziehungen zwischen Zielen, Plänen und Signalen und (c) die bedingten Beziehungen. Dabei differenziert Frese zwischen den Aufgaben, die mit Elementen der Gruppe (a) gemacht werden müssen: Elemente müssen bearbeitet, in einen Zeitrahmen gesetzt oder nach dem Inhalt verbunden werden.

Frese differenziert in drei Dimensionen: Ziele, Pläne und Signale; bearbeiten, in Zeitrahmen setzten und nach Inhalt verbinden; sowie Elemente, Beziehungen und bedingte Beziehungen. Dabei sind die Elemente der dritten stark abhängig von der ersten und der zweiten Dimension

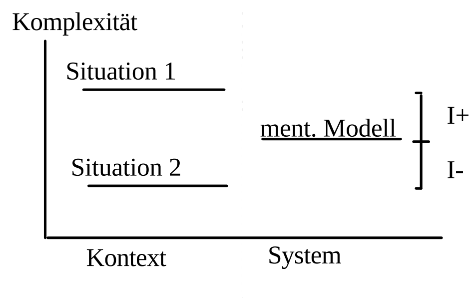

Abbildung 2: Inkongruität, aus [Rauterberg 1995]

Wichtig bei der Definition der Komplexität ist der Zwang (der Gegebenheiten) etwas zu tun: Es zählen die Entscheide, die gefällt werden müssen.

Die Komplexität eines Systems ist nun die Anzahl gezählter Elemente dieser drei Gruppen. Sie charakterisiert die Interaktion einer Person mit der Umgebung. Komplexität ist nach Frese somit nur die Anzahl Entscheidungen, die eine Person fällen muss, plus der Anzahl Beziehungen innerhalb dieser Entscheide.

2.11. Inkongruität

Die *Inkongruität* ist ein Mass aus der Softwareergonomie. In diesem Gebiet der Informatik und der Arbeitspsychologie geht es darum, Software auf Benutzerfreundlichkeit zu prüfen. Eine grundlegende Frage für diese Untersuchungen ist die Komplexität: Wie komplex ist ein Programm?

Rauterberg [Rauterberg 1995] betrachtet ein System und dessen Kontext. Einer Situation in diesem Kontext kann eine Komplexität zugeordnet werden. Gleichfalls wird die Komplexität des für diese Situation benötigten Teiles des mentalen Modells des Systems definiert. Der Begriff *Inkongruität* (Nichtübereinstimmung, Missverhältnis, Unangemessenheit) beschreibt nun die Differenz zwischen diesen beiden Komplexitäten, die Komplexität des Kontextes minus die des mentalen Modells (Vgl. Abbildung 2). Rauterberg unterscheidet dabei positive und negative Inkongruität. Mit Hilfe der Inkongruität lassen sich die Begriffe 'trivial' und 'unverständlich' definieren. Ein System ist trivial, falls die Inkongruität stark negativ ist und unverständlich, falls sie stark positiv ist.

Zwischen Information (nach Shannon) und der Inkongruität besteht ein qualitativ gleicher Zustand wie zum Beispiel bei der bedingten Komplexität (Vgl. 2.7). Information und Inkongruität sind in einer umgekehrten U-Kurve miteinander verbunden.

Rauterberg stellt in seinem Artikel [Rauterberg 1995] ein Modell für die Komplexitätsbetrachtung vor. Er betrachtet dabei die folgenden Komplexitäten: Die Komplexität des Inputs EC, die des Outputs AC, die des mentalen Modells MC und die der internen Quelle der Stimulation (ISS) BC, Vergleiche Abbildung 3.

Def. 1: Die *Komplexität des Kontextes* CC ist nun EC+BC.

Def. 2: Also ist die *Inkongruität* I=CC-MC.

Vorschläge, wie die Komplexität MC gemessen werden kann, finden sich in [Rauterberg 1992].

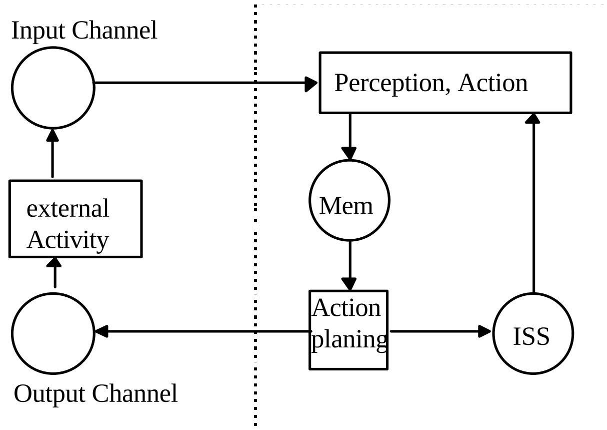

Abbildung 3: Wahrnehmung und Komplexität, aus [Rauterberg 1995]

3. Weitere Ansätze und Ideen

3.1. Komplexitätshierarchie von Mengen

Casti [Casti 1994] definiert eine *Mengenhierarchie*: Gegeben eine Menge R von Elementen. Die Elemente dieser Menge stehen auf dem Level N. Dann ist eine Menge S von Teilmengen aus R auf dem Level N+1. Dies lässt sich rekursiv weiterführen (vgl. Abbildung 4)

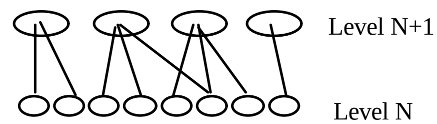

Abbildung 4: Mengenhierarchie

Atkin [Atkin 1981] überträgt diese Hierarchie auf Gefühle. Ein Betrachter beobachtet das Geschehen auf Level N. Atkin behauptet nun, dass durch einen Sprung des Betrachter von Level N auf Level N+1 (befreiend) Lachen und durch einen von Level N auf Level N-1 (einengend) Traurigkeit hervorgerufen wird.

Casti [Casti 1994] löst mit diesem hierarchischen Schema Paradoxe eines bestimmten Typs auf. Er behandelt das Barbierparadox: In einem Dorf rasiert der Barbier alle Männer, die sich nicht selber rasieren. Rasiert sich der Barbier nun oder rasiert er sich nicht? Es ist leicht zu sehen, dass sich der Barbier rasiert, genau dann wenn er sich nicht rasiert und umgekehrt, was eben ein solches Paradox darstellt. Casti löst dieses Paradox auf, indem er auf Level N alle männlichen Personen in eine Menge X nimmt. Die Person des Barbiers ist natürlich in dieser Menge enthalten. Das Konzept Barbier ist allerdings auf Level N+1 in einer Menge Y. Y ist die Teilmenge von X, welche alle Männer enthält, die sich nicht selber rasieren. Damit ist ersichtlich, dass das Konzept Barbier die Person Barbier nicht rasiert. Dies ist insofern auch natürlich, da sich der Barbier kaum selber bezahlen wird, und somit die Person Barbier nicht die Dienste des abstrakten Konzeptes Barbier in Anspruch nimmt.

Nach Casti entsteht ein Paradox dadurch, dass ein Element von einem höheren Level N+k in einen tieferen Level N gepresst wird.

Elemente auf dem gleichen Level sind auf der gleichen Komplexitätsstufe. D.h. für obiges Beispiel, dass der Begriff ‚Barbier' komplexer ist als der Begriff ‚Mann', da der ‚Barbier' eine Menge von ‚Männern' ist.

3.2. Syntax vs. Semantik: Komplexitätshierachie der Formalismen

Zentraler Bestandteil dieses Kapitel ist der Gödel'sche Beweis. Gödel hat gezeigt [Gödel 1931], dass, gegeben eine endliche Menge von Axiomen der Zahlentheorie, es Axiome gibt, die nicht aus jenen herleitbar sind und dass durch Hinzunahme von gewissen vorher nicht herleitbarer Axiomen das Axiomensystem widersprüchlich wird.

Gödel benutzte zum Beweis dieses Satzes ein damals ganz neues Verfahren: Er codierte Aussagen über die Zahlentheorie (logische Ausdrücke) in die Zahlentheorie hinein.

Gegeben sei ein *Formalismus* und eine Menge von Aussagen über diesen Formalismus, dann gilt, dass es Aussagen gibt, die nicht aus den vorhandenen Aussagen ableitbar sind und dass das System der Aussagen widersprüchlich wird, wenn genügend Aussagen eingebunden wurden.

Dies bedeutet, dass es mehr Aussagen über einen Formalismus gibt, als in diesen Formalismus abgebildet werden können. Also ist der Formalismus weniger mächtig, als die Aussagen über diesen Formalismus.

Rosen [Rosen 1989] interpretiert Gödels Beweis folgendermassen: Mathematik als *syntaktisches* Gebilde kann nicht von externen Referenzen (*Semantik*) befreit werden, jeder solche Versuch muss scheitern. Die Formalisierung der Zahlentheorie ist zu arm, zu eng, um die ganze Bedeutung der Zahlentheorie zu fassen und auszudrücken. Es benötigte mindestens unendlich viele Formalisierungen der Zahlentheorie, um alle deren Eigenschaften syntaktisch vorzunehmen. Die Semantik eines Formalismus ist stärker als die Syntax.

Es gibt somit Stufen der Komplexität von formalen Systemen. Zu jedem Formalismus gibt es einen weiteren Formalismus, der nicht in ersterem abgebildet werden kann und komplexer ist.

3.3. Makro- und Mikrozustände: Komplexitätshierarchie von Superzeichen

Die Definition für *Makro- und Mikrozustände* kommt aus der Physik, genauer gesagt aus der Thermodynamik. Für die Bewegung der Moleküle in einem Gas gelten die Maxwell'schen Gleichungen. Sie beschreiben Mikrozustände der Moleküle. Direkt davon abhängig und sehr viel einfacher ist die Temperatur dieses Gases. Zur Berechnung der Temperatur wird die mittlere Geschwindigkeit der einzelnen Moleküle benutzt. Die Temperatur ist ein Makrozustand, sie entspricht einer grossen Anzahl von Mikrozuständen (Verteilung der Geschwindigkeitsvektoren der einzelnen Moleküle). Viele völlig verschiedene Verteilungen können die gleiche Temperatur haben. Eine Verteilung kann sich sogar ändern, ohne dass sich die Temperatur verändert (z. B. beim Zusammenstoss zweier Moleküle).

Zufälligkeit ist auch ein Makrozustand oder eine Makrobeschreibung: Der Wurf eines Würfels wird ganz einfach mit der Zufallsvariable 'Anzahl oben liegende Augen' beschrieben. Tatsächlich könnte man, hätte man sämtlich Daten des Würfels gerade nach dem Abwurf und eine gewisse Anzahl weiterer Daten, das Verhalten des Würfels berechnen und voraussagen, welche Augenzahl oben zu liegen kommt. Die Aussage mit Wahrscheinlichkeit 1/6 ist die 3 oben, entspricht also einer ungeheuren Anzahl möglicher Flugbahnen des Würfels, bei welchen am Schluss eine 3 oben liegt.

Die Unterscheidung zwischen Makro- und Mikrozuständen impliziert verschiedene Standorte bzw. Sichtweisen eines Beobachters. Dabei kann ein Makrozustand selber wieder Mikrozustand sein.

Ähnliches geschieht beim Lesen eines Buchstabens. Es gibt viele Möglichkeiten den Buchstaben 'A' zu schreiben. Farbe, Grösse, Kontur, Hintergrund, Schriftdicke und mehr können fast beliebig variiert werden, und trotzdem ergibt sich ein Makrozustand 'A'. Der Makrozustand 'A' ist eine Repräsentation der Tatsache, dass ein Zeichen mit der semantischen Bedeutung 'A' gelesen wurde.

Der Mensch liest nun nicht einfach Buchstabe für Buchstabe sondern ganze Wörter. Dabei erkennt er ein Wort als Superzeichen aller darin vorkommenden Buchstaben. Das Wort selber sagt aber nichts aus. Das Konzept, welches der Leser mit diesem Wort verbindet enthält die Bedeutung des Wortes. Dieses Konzept ist ebenfalls ein *Superzeichen*, nämlich für alle Wahrnehmungen, die in diesem Konzept resultieren. Dabei hat ein Superzeichen höhere Komplexität als die einzelnen Teile. Superzeichen definieren somit eine Komplexitätshierarchie.

3.4. Theorie über Komplexe und Simplexe

Casti [Casti 1994] erwähnt eine Theorie der Komplexe. Gegeben seien zwei Mengen X und Y. Ausserdem sei eine Relation $R(x,y) \subseteq X \times Y$ gegeben. Diese Relation kann geometrisch wie folgt dargestellt werden: Die Elemente von Y sind Punkte in einem p-dimensionalen Raum. Zu jedem y_i in Y betrachten wir alle verwandten y_{i1} bis y_{in} (y_i ist mit sich selbst verwandt). Verwandt heisst, dass genau ein $x \in X$ existiert, so dass $R(x,y_{ik})$ für alle k gilt. Jedes Paar (y_{ik}, y_{il}), l<>k, definiert eine Kante zwischen diesen beiden Knoten, jedes Trippel eine Fläche usw. Somit stellt ein x ein n-1-dimensionales Objekt dar. Dieses Objekt ist ein *Simplex*. Für jedes x ⌊ X lässt sich ein solcher Simplex darstellen. Ein Simplex mit aus n+1 Knoten (Eckpunkten) hat die *Dimension* n und wird *n-Simplex* genannt.

Es ist klar, dass zwei Simplexe x und x' gemeinsame Punkte, Kanten usw. haben können, nämlich genau dann, wenn für gewisse Knoten y von x und x' die Relation R(x,y) und R(x‘,y) erfüllt ist. Diese Simplexe bilden somit ein p-dimensionales Objekt, welches *Komplex* genannt wird. Die Dimension p des Komplexes ist das Maximum über alle Simplexe der Dimension n eines Simplexes.

In einem Komplex sind die Simplexe miteinander verbunden, wenn auch nur indirekt. Die *Konnektivität* zweier Simplexe x, x' wird als die kleinstdimensionale Verbindungsstelle in der Kette von x nach x' definiert. Falls die Konnektivität von x nach x' q ist, wird von *q-Konnektivität* gesprochen. Ist zum Beispiel q=1, dann ist die kleinste Verbindungsstelle eine Gerade.

Oft interessiert, wie gut ein Simplex in die Struktur integriert ist. Ein Simplex x der Dimension n kann mehr oder weniger in einen Komplex integriert sein, d.h. er hat mehr oder weniger starke Verbindungen mit anderen Simplexen. Falls m die Anzahl Knoten der höchstdimensionale Verbindung mit einem anderen Simplex ist, so wird die *Exzentrizität* als n-m definiert. Diese Differenz ist signifikanter für kleine m. Aus diesem Grund wird noch durch (m+1) dividiert. Die Exzentrizität ist definiert: E=(n-m)/(m+1)

Diese Theorie ist verwandt mit der *Mengenhierarchie* aus 0. So wie die Elemente der Mengen Teilmengen einer Menge auf tieferem Level sind, so ist ein 2-Simplex aus 1-Simplexen zusammengesetzt und diese wiederum aus 0-Simplexen.

3.5. Zelluläre Automaten als Modell für Komplexität

Die in diesem Kapitel dargestellten Ideen sind aus [Wolfram 1984] entnommen.

Def. 1: Ein *zellulärer Automat* ist ein n-dimensionaler Raum, wobei jedem Punkt in diesem Raum zum Zeitpunkt t der Wert null oder eins zugeordnet wird. Bei jedem Zeitschritt werden die Werte aller Punkte mittels einer Funktion $f_{\underline{p}}(\cdot)$ bestimmt, wobei f von den Punkten im Umkreis $|\underline{r}|$ vom Punkt $\underline{p}$ abhängt.

Interessant ist, dass mit diesen einfachen Regeln im Verlaufe der Zeit Effekte auftreten, die man nicht sofort vermuten würde. So ist ein Vorgang in der Zeit im allgemeinen Irreversibel, d.h. praktisch, dass mehrere Zustände zu einem gleichen Folgezustand führen können. Ausserdem kann das Phänomen der Selbstorganisation beobachtet werden.

Für einen Automaten f(·) gibt es verschiedene Anfangskonfigurationen. Wolfram unterscheidet anhand des Evolutionsverhaltens vier Klassen von Anfangsmustern.

Klasse 1: Sie enthält die Muster, die sich zum leeren Muster hin entwickeln.

Klasse 2: Diese Klasse enthält die, welche zu einer konstanten, endlichen Grösse wachsen und dort dann periodische (die Periode kann eins sein) Mustersequenzen erzeugen.

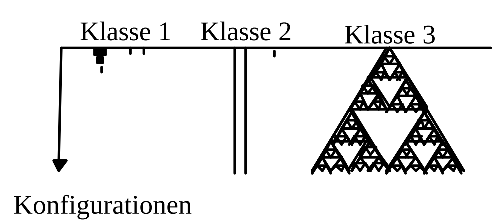

Abbildung 5: Anfangskonfigurationen, Skizze nach [Wolfram 1994]

Klasse 3: Muster der Klasse 3 zeigen fraktales Verhalten: Sie wachsen mit fester Geschwindigkeit und sind häufig ähnlich zu sich selbst.

Klasse 4: Diese Klasse schliesslich enthält Muster, die 'unvorhersagbares' Verhalten zeigen, sie wachsen und schrumpfen unregelmässig, sie zeigen aber häufig selbstähnliche Teilmuster.

In Abbildung 5 sind die Muster über die Zeit dargestellt (die Klassen 1 bis 3 sind mit einem Beispiel vertreten, Muster der Klasse 4 fehlen). Pro Zeiteinheit berechnet der Automat genau eine horizontale Linie der Muster.

Mathematisch interessant ist, dass der zelluläre Automat eine Touringmaschine in polynominaler Zeit simulieren kann, und umgekehrt, d.h. der zelluläre Automat ist *berechnungsuniversell*. Dazu benötigt es allerdings Anfangsmuster der Klasse 4.

3.6. Kommunikation und Informationsgehalt

Die Kommunikation zwischen zwei Maschinen (Computern) lässt sich sehr gut mit der *Informationstheorie*, begründet durch Shannon, beschreiben. Computer bewegen Bits von einem Ort zu einem anderen. Der Inhalt dieser Bits (Semantik) ist den Computern völlig unbekannt. Die Kapazität eines Kanals kann also durch die Information nach Shannon definiert werden.

Kommunikation zwischen zwei Menschen funktioniert aber nicht auf derselben Ebene wie bei Maschinen. Nach Norretranders [Norretranders 1994] wird auf unterster Ebene *Information* mittels Worten übermittelt. Die Worte sind aber nur das *syntaktische* Gerüst für das, was eigentlich gesagt werden sollte, der *Formalismus*. Auf einer höheren Ebene wird *Exformation* übertragen. Dies ist der *semantische* Inhalt der Worte, die übertragen werden.

Die Kommunikation auf der höheren Ebene läuft folgendermassen ab: Der Sender hat ein bestimmtes mentales Modell, das wir uns hier, der Einfachheit halber, als Baum vorstellen. Durch die Wahrnehmung und Verarbeitung von Information wird der Baum wachsen. Bei der Kommunikation mit einem anderen Menschen wird nun so ein Baum, oder ein Teilbaum, in eine Aussage codiert (Norretranders nennt es *Inzitation*), dem anderen mitgeteilt und dieser versucht nun aus diesem syntaktischen Gerüst die gemeinte Bedeutung wieder aufzubauen (*Exzitation*) (vgl. Abbildung 6). Dies kann er aber nur, wenn sein mentales Modell genügend grosse Komplexität aufweist. Bei dem Vorgang der Exzitation steht dem Empfänger nicht nur der gesprochene Satz sondern auch sehr viele weitere Eingabedaten zu Verfügung, wie z. B. Tonfall, Mimik und Gestik. Ohne aber ein genügend mächtiges Modell des Kontextes kann ein Empfänger die gemachte Aussage nicht verstehen.

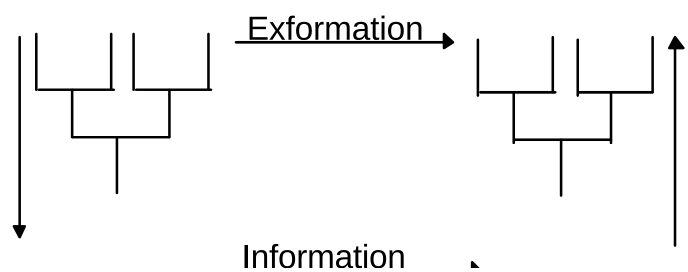

Abbildung 6: Information, Exformation, nach [Norretranders 1994]

Nach Norretranders [Norretranders 1994] steckt die Komplexität einer Aussage direkt im mentalen Modell des Senders. Wird eine Aussage aufgrund eines grossen Baumes gemacht, d.h. musste viel Information verarbeitet werden, um diesen Baum aufzubauen, so hat diese Aussage eine grosse Tiefe und somit eine hohe Komplexität. Es ist direkt ersichtlich, dass dieselbe Aussage (von den gesprochenen Worten aus gesehen) je nach Sprecher eine völlig andere Bedeutung und Tiefe hat.

Die Exformation beinhaltet die Tatsache, dass Information verarbeitet wurde. Sie ist die verarbeitete Information.

Es ist naheliegend, das durch jedes verarbeitete Ereignis das mentale Modell komplexer wird. Abstraktionslevel und Komplexität sind in diesem Modell zumindest verwandt.

Die Unterscheidung zwischen Information und Exformation hilft uns zum Beispiel bei der algorithmischen Komplexität (Vgl 2.2) weiter: Es wurde das Beispiel vom zufällig getippten Text eines Affens und Goethes Faust erwähnt. Der Unterschied zwischen den beiden Sequenzen ist doch offensichtlich der, dass der vom Affen getippte Text keine Exformation enthält, während diese bei Goethes Faust sehr gross ist. Die Information hingegen ist bei Faust eher klein, doch bei der Zufallssequenz des Affen sehr hoch.

4. Vergleich und Diskussion der Komplexitätsmasse

4.1. Information und Komplexität

Komplexität als Mass für die Struktur eines Systems hat eine wichtige Verbindung mit der Shannon'schen Information dieser Struktur. Die Shannon'sche Information ist maximal, wenn das System völlig ungeordnet ist, also keine sichtbare Struktur aufweist und minimal, wenn die Ordnung des Systems vollkommen ist (d.h. trivial). Ist das System völlig geordnet, dann ist die Komplexität des Systems klein. Die Frage ist, wie gross die Komplexität ist, wenn das System völlig ungeordnet ist. Die *algorithmische Komplexität* gibt einem solchen System maximale Komplexität, es ist völlig zufällig. Die *bedingte Komplexität* oder auch die *logische Tiefe*, geben diesem System eine sehr kleine Komplexität. Bei diesen beiden Komplexitätsmassen (als Beispiel) ist die Komplexität dort maximal, wo die Struktur am aufwendigsten zu berechnen ist. Bei der bedingten Komplexität ist die Struktur aus vielen verschiedenen Symmetrien aufgebaut, bei der logischen Tiefe braucht die Berechnung riesigen Aufwand. Die *thermodynamische Tiefe* und die *Inkongruität* messen ähnlich: sie sind irgendwo zwischen totaler Ordnung und Chaos am grössten.

Die Frage, wo die Komplexität am höchsten sein sollte, ist nicht so eindeutig beantwortbar, wie es vielleicht scheint. Einerseits scheint der Ansatz der logischen Tiefe bestechend. Er entspricht vielen Beobachtungen aus dem täglichen Leben (für Beispiele vgl. [Rauterberg 1995]). Andererseits stellt sich die Frage, ob nicht eine *algorithmisch zufällige* Folge wesentlich komplexer ist als irgendeine andere, da sie auf einer Touringmaschine nur geprinted, aber nicht ‚berechnet' werden kann. Dies bedeutet nämlich nur, dass das Modell der Touringmaschine zu schwach ist, um eine kürzere Beschreibung als das Printprogramm zu finden. Daraus folgt, dass das *Modell der Touringmaschine* kleinere Komplexität als eine solch zufällige Folge hat. Diese Beobachtung wird durch Chaitin bestärkt: Er definiert die Komplexität eine Folge, für welche es kein Programm gibt, als unendlich (vgl. Def. 2 in 2.2.2) und akzeptiert damit implizit die Schwäche der Touringmaschine.

4.2. Breite vs. Tiefe

In diesem Kapitel wird ein *Breite-Tiefe-Modell* vorgestellt. Es hat zwei Dimensionen. Die erste ist die räumliche Dimension, die zweite ist die der Veränderung. Betrachten wir ein beliebiges System. Im allgemeinen hat dieses System zu einem Zeitpunkt t eine räumliche Struktur. Zusätzlich unterliegt diese Struktur einer Veränderung, wobei es sich von einem Zeitpunkt t_o zu einem Zeitpunkt t_n immer weiter entwickelt. Es können hier zwei Arten von Komplexität definiert werden (Vgl. Abbildung 7). Einerseits kann die Komplexität die Struktur zu einem Zeitpunkt t beschreiben (*Breite*), andererseits aber auch die Entwicklung der Struktur (*Tiefe*).

Benutzen wir den von Wolfram beschriebenen *zellulären Automaten* (vgl. 3.5), dann sind die Breite und die Tiefe in einigen Fällen gleich, in anderen aber völlig verschieden. Für fraktale Muster (Klasse 3) sind sie ähnlich, aber für Muster der Klasse 4 ist keine Korrelation vorhanden.

Durch die einzelnen Ebenen der Tiefe wird eine *Hierarchie* definiert. Von einer Ebene zur nächsten ist eine informationsverarbeitende Abbildung vorhanden.

Als Illustration werden die Grundoperationen betrachtet. Auf der ersten Ebene ist eine endliche Menge von Ziffern, die alle addiert werden sollen. Durch die Definition der Multiplikation wird auf eine höhere Ebene aufgestiegen. Auf dieser Ebene existieren weniger Zahlen, da einige Zahlen zu einem Produkt zusammen gefaßt werden konnten, dafür aber die zwei Operatoren, Addition und Multiplikation. Die Anzahl der Elemente nimmt ab, wogegen die Anzahl verschiedener Elemente zugenommen hat, und tiefere Operatoren erscheinen. Es wurde mehr *Semantik* und *Exformation* eingeführt, um weniger *Syntax* und *Information* zu haben.

4.3. Komplexitätsmasse und das Breite-Tiefe-Modell

Wenden wir das Breite-Tiefe-Modell auf einige der vorgestellten Komplexitätsmasse an:

Beginnen wir mit der *Berechnungskomplexität*. Das zu betrachtende System ist die Touringmaschine, genauer gesagt das Band der Touringmaschine. Die Touringmaschine wird mit einer Anfangsbelegung bestückt und ändert diese solange ab, bis ein Problem gelöst ist. Die Komplexität ist demzufolge ein Mass für die Entwicklung, ein sehr einfaches übrigens, denn es wird die Anzahl Zustandsübergänge gezählt. Die Berechnungskomplexität ist eine Tiefe.

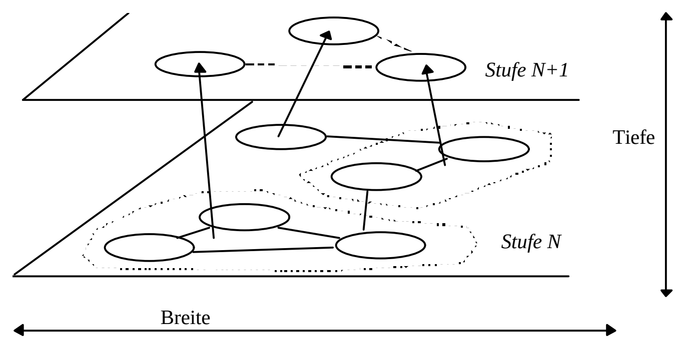

Abbildung 7: Breite und Tiefe

Die *algorithmische Komplexität* von Chaitin und Kolmogorov benutzt eine Touringmaschine, welche durch ein Programm p charakterisiert wird. Das System ist ein binäres Wort welches auf eine bestimmte Art ein anderes binäres Wort beschreibt. Dabei gibt es durchaus Zustände, sie werden aber nicht betrachtet. Die eigentliche Entwicklung beginnt nämlich mit einem beliebigen Wort w, welches nun Schritt um Schritt so komprimiert werden soll, dass am Schluss die kürzest mögliche Form herauskommt (vgl. mit *Intelligenz* in 2.2.3). Bei dieser Entwicklung wird der Initialzustand, ein syntaktisch inkorrektes Wort w, grosszügig als Programm interpretiert. Die algorithmischer Komplexität operiert in diesem Sinne lediglich auf der räumlichen Dimension und vernachlässigt die Dimension der Veränderung vollständig.

Bennetts *logische Tiefe* mischt die Berechnungskomplexität und die algorithmischen Komplexität. Sie zählt ebenfalls die Anzahl benötigter Zustandsübergänge um aus einem Anfangszustand ein Endzustand zu berechnen. Dabei bedient sich Bennett des (nicht berechenbaren) kürzesten Programmes. Die logische Tiefe ist ebenfalls ein Mass in der Dimension der Veränderung. Im Gegensatz zur Berechnungskomplexität, welche vorallem nur qualitativ eingesetzt wird, ist die logische Tiefe ein quantitatives Mass.

Die *thermodynamische Tiefe* von Lloyd und Pagels operiert ebenfalls auf der Dimension der Veränderung. Es ist dies das erste Mass, welches dies explizit tut. Es ist das kompletteste dieser vier Masse, da es sowohl mathematisch als auch physikalisch definiert und auf Plausibilität überprüft worden ist.

Die *Shannon'sche Information* betrachtet nur eine Ebene und berechnet ein Mass über die vorkommenden strukturellen Elemente. Dabei behandelt das Shannon'sche Informationsmass aber alle Elemente, wie wenn sie auf der untersten Stufe der Hierarchie stünden, ohne das *semantische* Gewicht eines Elementes zu berücksichtigen (die Multiplikation ist semantisch schwerer, da auf höherer Ebene, als die Addition). Der Informationsgehalt presst also Objekte von Level N auf den Level 1.

Er betrachtet lediglich die *syntaktischen* Elemente. Die Masse der Tiefe gewichten eine Multiplikation stärker als eine Addition. Das Gleiche gilt, zumindest für dieses Beispiel, auch für die algorithmische Komplexität, bei welcher eine Multiplikation ein längeres Programm benötigt, als eine Addition.

4.4. Objektive und Subjektive Komplexitätsmasse

Bei der Definition von Komplexität gibt es zwei Arten von *Subjektivität*. Einerseits ist die Definition des Masses selber von Person zu Person unterschiedlich, d.h. jede Person definiert und versteht die Komplexität anders. Andererseits kann die Komplexität explizit abhängig von einer Person definiert werden. Im ersten Fall ist ein System und ein Versuchsleiter da, der misst wie komplex das System ist. Im zweiten Fall besteht die Versuchsanordnung aus einem System und einer Versuchsperson (VP), welche mit dem System interagiert. Dabei misst ein Versuchsleiter, wie komplex das System für die Versuchsperson ist. Die ersteren Masse beziehen sich also auf ein System, die zweiten auf ein System relativ zu einem anderen. Erstere Art nennen wir von nun an *objektiv* oder *absolut* (da kein Zwischensubjekt vorhanden ist) und zweite Art nennen wir *subjektiv* oder *relativ*.

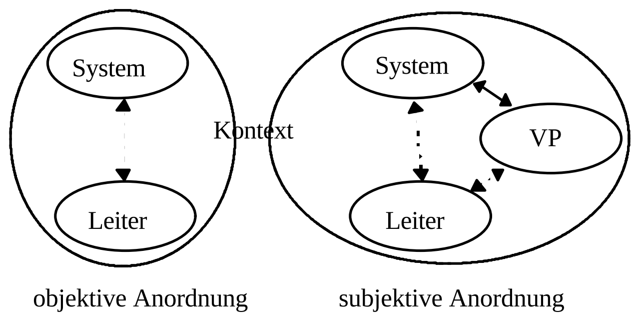

Abbildung 8: objektive und subjektive Masse

In Abbildung 8 ist dieser Sachverhalt dargestellt. Nicht unwesentlich ist, dass alle Systeme teil eines Kontextes sind, auch wenn der Versuchsleiter dies eigentlich nicht will. Ausserdem ist die Bedeutung der Pfeile wichtig: Die Doppelpfeile sagen aus, dass eine Wechselwirkung zwischen den Systemen geschieht. Die gestrichelten Pfeile sind die nicht erwünschten Seiteneffekte, die durchgezogenen Pfeile sind die beobachteten Interaktionen.

Die meisten Komplexitätsmasse sind objektive Masse. Mathematik, Physik und Biologie interessieren sich nur für solche Masse. Erst die Psychologie versucht subjektive Masse zu definieren. *Inkongruität* (vgl. 2.14) ist ein Beispiel für diesen Ansatz. Aus der Definition von Subjektivität ist ersichtlich, dass subjektive Masse aus objektiven zusammengesetzt werden können. Dies wurde z. B. auch für bei der Inkongruität (Vgl. [Rauterberg 1995]) gemacht. Ähnliches wurde hier für die *algorithmische Komplexität* (vgl. 2.2.3) durchgeführt.

4.5. Unverständlichkeit und Schwierigkeit vs. Komplexität

In der Einleitung wurde gesagt, dass die Komplexität damit zu tun habe, wie *unverständlich* und *schwierig* ein System sei. Durch die verschiedenen dargebrachten Argumente, besonders aber durch das Mass der *Inkongruität*, ist klar ersichtlich, dass es umgekehrt ist, nämlich das Unverständlichkeit und Schwierigkeit Funktionen der Komplexität sind. Die Definition von Unverständlichkeit benötigt zwei Systeme. Jedem System (oder einem Teil) kann nun eine Komplexität zugeordnet werden. Die Unverständlichkeit ist eine Funktion dieser Komplexitäten, zum Beispiel der Differenz (oder Inkongruität).

Nach der These von Lloyd und Pagels (vgl. These 1 in 2.8.2) gibt es aber drei verschiedene Arten von völliger Unverständlichkeit: Wie schon in der Einleitung erwähnt, kann es sich um die prinzipielle Unfähigkeit überhaupt handeln (wie Touringmaschine und Halteproblem) oder das betrachtete System ist zu ‚dumm' und müsste lernen. Es kann nun aber auch sein, dass etwas zwar zum aktuellen Zeitpunkt prinzipiell unmöglich ist, weil die Zeit zu kurz war, um den benötigten Komplexitätslevel zu erreichen, aber in absehbarer Zeit möglich sein wird.

Schwierigkeit kann direkt mit der *Kompliziertheit* nach Frese (vgl. Einleitung) definiert werden.

Unverständlichkeit und Schwierigkeit können somit durch die *subjektive Komplexität* ausgedrückt werden.

4.6. Komplexität und Hierarchien von Komplexitäten

Im Zusammenhang mit Komplexität zeigen sich zwei fast widersprüchliche Betrachtungsweisen. Einerseits gibt es die *kontinuierliche* Betrachtung, wie sie im Kapitel 2 dargestellt wurde, andererseits scheint es *Stufen* komplexer Systeme zu geben. Darauf deuten zum Beispiel die in den Kapiteln 0, 3.2 und 3.4 erwähnten Hierarchien hin. Besonders die Komplexitätshierarchie der *Formalismen* in 3.2 ist ein mathematisch genaues Konstrukt.

Es stellt sich die Frage, ob es sich bei den Komplexitätsmassen nicht gleich verhält, wie mit der Shannon'schen Information. Diese operiert im Breite-Tiefe-Modell nur auf der untersten Ebene und presst alles, was weiter oben ist, in diese hinunter. Es könnte durchaus sein, dass die verschiedenen Masse in Kapitel 2 sich insofern ähnlich verhalten, als dass sie nur auf einer Ebene sind und für jede solche Ebene neu angewandt werden müssen. Die *thermodynamische Tiefe* wird zum Beispiel für die Quantenphysik und die klassische Physik durch Lloyd und Pagels jeweils neu definiert.

Die Interpretation solcher *Komplexitätstufen* ist durchaus interessant. Zum Beispiel deutet die Anwesenheit solcher Stufen auf die in der Einleitung erwähnten *Emergenzphänomene* hin. Dabei stellt sich die Frage, welche Komplexität benötigt ein System, um ein bestimmtes Phänomen zu zeigen, wie z. B. das Bewusstsein beim Menschen. Ausserdem ist die Frage interessant, ob ein System von einer Stufe auf die nächste aufsteigen kann.

Reduktionisten sind der Ansicht, dass sich sämtliche Gebiete der Wissenschaft auf physikalische Kategorien reduzieren lassen, sich also durch die physikalischen Gesetzte allein erklären lassen. Diese Ansicht entspricht einem stufenlosen Komplexitätsmass. Im Gegensatz dazu steht die Ansicht, dass sich bei genügend hoher Komplexität eines Systems Phänomene ergeben, die durch die bis anhin gültigen Gesetze nicht mehr beschrieben werden können. In so einem Fall tritt eine neue Komplexitätstufe auf. Dies ist die viel diskutierte Frage von Syntax vs. Semantik.

4.7. Modellierung von Komplexität

Zum Messen der Komplexität benötigen wir ein *Modell*. Diese Modell *formalisiert* Zusammenhänge aus der ‚Realität' (vgl. Abbildung 9). Das Modell kann zum Beispiel die Touringmaschine sein. Aus der Diskussion in 4.1 kam als Teilresultat heraus, dass jedes Modell eine bestimmte Komplexität hat. D.h. das gewählte Mo-

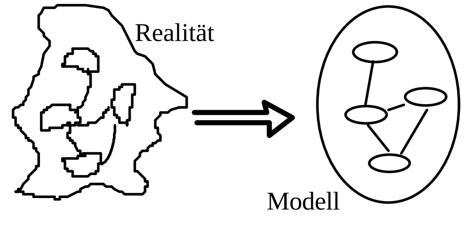

Abbildung 9: Modellbildung

dell ist ein Teil der *Subjektivität*, die in den *objektiven* Massen versteckt vorhanden ist. Das Modell und die Abbildung eines Sachverhaltes in dieses Modell sind wichtige Faktoren für die gemessene Komplexität. Um genau diesen Umstand dreht sich die Diskussion, die durch Gödel angefacht wurde (Vgl. Kapitel 3.2). Die Argumente von Gödel und Rosen zeigen nämlich, dass es zu jedem Modell ein Modell gibt, das noch komplexer ist. Es gibt also ein Modell, das komplexer als die Touringmaschine ist, und dem somit weniger Zahlen zufällig erscheinen als der Touringmaschine. Als Beispiel diene ein Modell, welches weniger komplex als die Touringmaschine ist: Der Kellerautomat. Der Kellerautomat ist sehr ähnlich der Touringmaschine definiert. Er hat aber kein Band sondern k Stapel (Ein Stapel ist ein Speicher mit den Operationen ‚Put' und ‚Get', wobei ‚Put' ein Zeichen schreibt und ‚Get' das zuletzt geschriebene Zeichen holt). Falls k=2, dann ist der Kellerautomat gleich mächtig wie die Touringmaschine. Für k=1 gibt es Funktionen, die auf einer Touringmaschine berechenbar sind, nicht aber auf einem Kellerautomaten. D.h. es sollte Wörter geben, die auf einem Kellerautomaten *algorithmisch zufällig* sind, nicht aber auf einer Touringmaschine.

Somit stellt sich die Frage aus Kapitel 4.1 neu: Wo soll das Maximum der Komplexität sein. Je stärker das Modell für die Komplexität ist, desto weniger algorithmisch zufällige Zahlen gibt es. D.h. die *logisch tiefen* Zahlen sind tendentiell auch die algorithmisch komplexesten. Es gibt übrigens keinen Grund anzunehmen, dass die neu nicht zufälligen Zahlen im neuen Modell eine grosse Komplexität haben.

In dieser Diskussion wurde die Situationen betrachtet, in der das gewählte Modell explizit die Touringmaschine ist. Wie verhält es sich aber nun bei der *thermodynamischen Tiefe*, die nicht auf jener definiert wurde? Die thermodynamische Tiefe ist nicht für ein bestimmtes Modell sondern für beliebige Modelle definiert. Sie behandelt Zustände, die in einem solchen angenommen werden können. D.h. die thermodynamische Tiefe muss für jedes gewünschtes Modell neu angepasst werden. Ein solches kann zum Beispiel das Modell der Quantenphysik oder der klassischen Physik sein, aber auch ein mathematisches Modell, eben wie z. B. das der Touringmaschine.

Um ein angemessenes Mass für Komplexität zu finden, muss deshalb eine genügend starke Modellierung gefunden werden.

5. Zusammenfassung

Im Kapitel 2 wird eine Reihe von Komplexitätsmassen vorgestellt. Diese Aufzählung ist sicher nicht vollständig und es exisitieren zweifellos Masse, die grössere Bedeutung haben, als einige der hier aufgeführten.

Es fällt auf, dass Komplexitätsmasse, welche von grosser theoretischer Bedeutung sind wie zum Beispiel die Berechnungskomplexität, die algorithmische Komplexität oder die thermodynamische Tiefe zwar generell, aber in der Anwendung (zum Beispiel die Komplexität eines Huhnes) nur schwer zu gebrauchen. Andere Masse wie zum Beispiel einige Ansätze aus der Biologie oder die *Inkongruität* sind zwar auf dem speziellen Gebiet gut anwendbare Masse, können aber nicht als grundsätzliches Mass für Komplexität betrachtet werden.

Die wichtigsten Masse aus theoretischer Sicht sind: Die *Zeitkomplexität* (Berechnungskomplexität) als Mass für die *Machbarkeit*, die algorithmische Komplexität als Mass für *Zufälligkeit*, die *logische Tiefe* als spezielles Tiefenmass auf der Touringmaschine, sowie die *thermodynamische Tiefe* als allgemeines Tiefenmass.

Kapitel 3 enthält einige Argumente und Konzepte, welche mit Komplexität zu tun haben. Als erstes wurde das Konzept von *Komplexitätsebenen* eingeführt. Dies anhand den Beispielen der *Mengenhierarchie* (3.1), von *Syntax* und *Semantik* (3.2), sowie *Mikro- und Makrozuständen* (3.3).

Dann wurde die Theorie der *Simplexe* und der *Komplexe* eingeführt, womit Zusammenhänge in einem Kontext einfach modelliert werden können und die ebenfalls eine Hierarchie darstellen (3.4). Die Dimensionalität eines Simplexes ist ein Mass für die Komplexität dieses Simplexes. Weiterhin wurde der *zelluläre Automat* als ein einfaches Modell angegeben, bei dem einige Phänomene komplexer Systeme betrachtet werden können (3.5).

Die Sammlung wird abgeschlossen durch einige psychologischen Betrachtungen. Es wurde der Begriff der *Exformation* als Alternative zur Shannon‘schen *Information* eingeführt (3.6).

In Kapitel 4 wurden die Masse, Argumente und Konzepte geordnet und in Zusammenhang gebracht. Dabei interessierte zuerst der Zusammenhang zwischen *Information* und Komplexität (4.1). Dabei ist klar, dass kleiner Informationgehalt auch kleine Komplexität bedeutet. bei grossem Informationsgehalt ist es nicht eindeutig, ob eine grosse oder eine kleine Komplexität zugeordnet werden muss. Es wurde ersichtlich, dass das dem Komplexitätsmass zugrunde liegende *Modell* eine zentrale Rolle spielt.

Als nächstes wurden die Begriffe *Tiefe* und *Breite* eingeführt (4.2) und auf die theoretisch wichtigsten Masse angewandt. Einzig die *algorithmische Komplexität* ist ein Breitenmass. Die andern drei (*Berechnungskomplexität*, *logische Tiefe* und *Thermodynamische Tiefe*) sind Tiefenmasse. Ausserdem wurde die Idee vertreten, dass die Shannon‘sche *Information* ein Breitenmass ist, und dabei die Elemente auf der untersten Ebene betrachtet (4.3). In 4.4 wurden *objektive* und *subjektive* Masse mittels zwei bzw. drei Systemen definiert und gezeigt, dass eine praktische Definition (z. B. die *Inkongruität*) eines subjektiven Masses zwei objektive Messungen enthält.

Die Begriffe *Unverständlichkeit* und *Schwierigkeit* konnten ins rechte Licht gerückt werden. Beide können nämlich auf subjektive Komplexität zurückgeführt werden (4.5).

Insbesondere die Resultate von Gödel [Gödel 1931] und Rosen [Rosen 1989] deuten darauf hin, dass es *Komplexitätsstufen*. gibt. Die in Kapitel 2 aufgezählten Komplexitätsmasse zeigen keine solche Stufen. Es wurde die Annahme vertreten, dass die Masse nur auf einer Komplexitätsstufe definiert worden sind, und auf anderen nicht von vornherein gelten (4.6) .

Zum Abschluss wurde das Modellieren von Komplexitätsmassen betrachtet. Es wurde argumentiert, dass die Wahl des *Modells* entscheidenden Einfluss auf die Brauchbarkeit des Komplexitätsmasses hat und dass z. B. die Touringmaschine nicht mächtig genug ist. Es wurde auch gezeigt, dass die *thermodynamische Tiefe* im ersten Ansatz auf keinem Modell definiert wurde. Sie muss deshalb für die benötigten Modelle neu definiert werden. Dies wurde für die Quantenphysik und die klassische Physik von Lloyd und Pagels schon getan (4.7).

6. Anhang

6.1. Literaturverzeichnis

[Abu-Mostafa 1988] Abu-Mostafa, Y. S. (1988): Complexity of Random Problems. In: Abu-Mostafa, Complexity in Information Theory. New York: Springer, p115-131.

[Atkin 1981] Atkin, R. (1981): Multidimensional Man. London: Penguin Books.

[Bennett 1988] Bennett, C. (1988): Logical Depth and Physical Complexity. In: Rolf Herken (Editor), The Universal Touring Machine. Berlin: Oxford University Press, p 227-259.

[Casti 1994] Casti, J. L. (1994): Complexification, Explaining a Paradoxial Wordl Through the Science of Surprise. USA: HarperCollins Publisher.

[Chaitin 1974] Chaitin, G. J. (1974): Information theoretic computational complexity. In: Information, Randomness & Incompleteness. Series in Computer Science Vol 8; World Scientific, p 29-38.

[Crutchfield, Young 1989] Crutchfield; Young (1989): Computation at the Onset of Chaos. In: Wojciech H. Zurech, Complexity, Entropy and the physics of Information. Santa Fe: Addison-Wesley, p 223-269.

[Engeler, Läuchli 1992] Engeler, E. & Läuchli P. (1992): Berechnungstheorie für Informatiker. Zürich: B.G.Teubner.

[Frese 1987] Frese, M. (1987): A theory of control and complexity. In: Frese, Ulich und Dzida (Editors), Psychological Issues of Human Computer Interaction in the Work Place. North-Holland: Elsevier Science Publishers B.V., p 313-337.

[Gödel 1931] Gödel, K. (1931): Über formal unentscheidbare Sätze der Prinzipia Mathematica und verwandter Systeme. In: Kurt Gödel, collected Works Vol 1. New York: Oxford University Press.

[Gell-Mann 1994] Gell-Mann, M. (1994): The Quark and the Jaguar, Adventures in the Simple and the Complex. New York: W.H. Freeman and Company.

[Johnson 1988] Johnson, L. (1988), The Thermodynamic Origin of Ecosystems, A Tale of Broken Symmetry. In: Bruce, Weber, Depew und Smith (Editors): Entropy, Information and Evolution. Cambridge MA: MIT Press, p 75-105.

[Karchmer 1988] Karchmer, M. (1988): Communication Complexity, a new approach to circuit depth. Cambridge MA: MIT Press.

[Kuhn, Waser 1983] Kuhn; Waser (1983): Biophysics. Hoppe, Lohmann, Markte und Ziegler (Editors). New York: Springer Verlag.

[Layzer 1977] Layzer, D. (1977): Information in cosmology, physics and biology. In: International Journal of Quantum Chemistry 12 (suppl. 1), p 185-195.

[Llyod, Pagels 1988] Lloyd, S. & Pagels (1988): Complexity as Thermodynamic Depth. In: Annals of Physic vol 188, p 186-213.

[Norretranders 1994] Norretranders, T. (1994): Spüre die Welt, die Wissenschaft des Bewusstseins. Rowolth.

[Rauterberg 1992] Rauterberg, M. (1992): A Method of a Quantitative Measurement of Cognitive Complexity. In: Van der Veer, Tauber, Bagnaram und Antalovits (Editors): Human Computer Interaction, Tasks and Organization. Roma: CUD, p 295-307.

[Rauterberg 1995] Rauterberg, M. (1995): About a framework for information and information processing of learning systems. In: International Conference on Information System Concepts, Marbrug/Lahn, p 4.1-4.15.

[Rosen 1989] Rosen, R. (1989): Hard Science vs soft science. In: Shimizu, Biological complexity and information. Fuji-Suomo (Japan): World Scientific, p 228-240.

[Schöning 1988] Schöning, U. (1988): Complexity Theorie and Interaction. In: Rolf Herken (Editor): The Universal Touring Machine. Berlin: Oxford University Press, p 561-580.

[Schöning 1985] Schöning, U. (1985): Complexity and Structure. Goos, Hartmanis (Editors): Lecture Notes in Computer Science. Berlin: Springer Verlag.

[Shannon, Weaver 1949] Shannon, C. & Weaver (1949): The mathematical Theorie of Communication. Urbana: University of Illinois Press.

[Shimizu 1989] Shimizu, H. (1989) A Dynamical Approach to Semantic Communication - The Significance of Biological Complexity in the Creation of Semantic Information. In: Shimizu: Biological complexity and Information. Fuji-Suomo: World Scientific; p 145ff.

[Stockmayer 1977] Stockmayer (1977): The polynominal-time hierarchie. In: Theorie of Complexity 3 (1977), p 1 - 22.

[Traub, Wasilikowski, Wozniaskowski 1988] Traub, J. F.; Wasilikowski, G. W. & Wozniakowski H. (1988): Information-based Complexity. New York: Academic Press.

[Traub 1988] Traub, J. F. (1988): Introduction to Information-Based Complexity. In: Abu-Mostafa: Complexity in Information Theory. New York: Springer, p 62-76.

[Wicken 1989] Wicken, J. S. (1989): Can Information be Quantified by Shannon Formalism. In: Shimizu: Biological complexity and information. Fuji-Suomo (Japan): World Scientific, p 5-25.

[Wiley 1988] Wiley, E. O. (1988): Entropy and Evolution. In: Entropy, Information and Evolution. Bruce, Weber, Depew, Smith (Editors); Cambridge MA: MIT Press, p 173-188.

[Wolfram 1984] Wolfram, S. (1984): Cellular automata as models of complexity. Nature 311; p 19-24.