SSOAR

Open Access Repository
www.ssoar.info

Wie ist es Max Weber zu sein?

Mumdzhiev, Milko

Preprint / Preprint
Arbeitspapier / working paper

Empfohlene Zitierung / Suggested Citation:
Mumdzhiev, M. (2011). *Wie ist es Max Weber zu sein?* (Nürnberger Beiträge zur Sozial- und Wirtschaftsforschung, 06/11). Nürnberg. https://nbn-resolving.org/urn:nbn:de:0168-ssoar-256793

Nutzungsbedingungen:
Dieser Text wird unter einer CC BY-NC-ND Lizenz (Namensnennung-Nicht-kommerziell-Keine Bearbeitung) zur Verfügung gestellt. Nähere Auskünfte zu den CC-Lizenzen finden Sie hier:
https://creativecommons.org/licenses/by-nc-nd/4.0/deed.de

Terms of use:
This document is made available under a CC BY-NC-ND Licence (Attribution-Non Comercial-NoDerivatives). For more Information see:
https://creativecommons.org/licenses/by-nc-nd/4.0

gesis
Leibniz-Institut
für Sozialwissenschaften

Mitglied der
Leibniz-Gemeinschaft

Nürnberger Beiträge zur Sozial- und Wirtschaftsforschung 06/11

Wie ist es Max Weber zu sein?

Milko Mumdzhiev

Zusammenfassung:
Die Präsenz eines Soziologen von der Größe Max Webers soll operationalisiert und simuliert bzw. approximiert werden.
Entsprechende soziologische Architekturen werden an der Folie der Informatik (Soft- und Hardware, Theorie und Praxis) vorgezeichnet, und in (Gedanken-) Experimenten zur Leistungs- oder Effizienzsteigerung, allgemein zur Anhebung einer gewöhnlichen soziologischen Theorie- und Praxisführung auf Größtenniveau, der Realisation und Verifizierung übergeben.

Copyright © 2011 Milko Mumdzhiev
Das Arbeitspapier darf nicht ohne die ausdrückliche Genehmigung des Autors reproduziert werden.

Gliederung

1. Einführung S.3
2. Die geisteswissenschaftliche Sichtweise S.12
2.1 “Fingierte” Hypothesen S.12
2.2 Theorienleistungen, die Soziologie und der Max Weber Automat S.17
2.2.1 Theorienleistungen S.17
2.2.2 Methode der Konstruktion von Theorien S.17
2.2.3 Modellierung sozialer Prozesse S.18
2.2.4 Empfehlungen für die Modellierung sozialer Prozesse S.19
2.2.5 Schlussfolgerungen für die Soziologie S.19
3. Die technische Sichtweise S.29
3.1 Grundlagen S.29
3.1.1 Klassifikation von Algorithmen S.29
3.1.2 Logische Programmierung S.32
3.1.3 SAT-Solver S.33
3.1.4 SMT-Solver S.37
3.1.5 Simulation von Max Weber S.39
3.2 Konzepte für Computerarchitekturen S.40
3.3 Einprozessorsysteme und Speicher S.43
3.4 Pipelining, Ausführungssteuerung, Datenübertragung und Leitungen, Busse, Schaltungen S.45
3.5 Erweiterung des von-Neumann-Berechnungsmodells S.48
3.6 Multiprozessorsysteme S.50
4. Diskurse und Inhalte S.55
4.1 Kommunikation und Konversation S.55
4.2 Die ideale Sprechsituation S.59
4.3 Regelkanon für den allgemeinen praktischen Diskurs S.61
4.4 Diskussion, Argumentation, Ausblick S.63
5. Verwendete Literatur S.66
6. Anhang S.68

Müll rein, Müll raus, weiß der Informatiker.
Müll rein, besteht der Eimer.
Müll raus, sagt der Deutsche.

1. Einführung

Kant leitete die Beantwortung der Frage: Was ist Aufklärung? wie folgt ein:
Aufklärung ist der Ausgang des Menschen aus seiner selbstverschuldeten Unmündigkeit. Unmündigkeit ist das Unvermögen, sich seines eigenen Verstandes ohne Leitung eines anderen zu bedienen. Selbstverschuldet ist diese Unmündigkeit, wenn die Ursache derselben nicht am Mangel des Verstandes, sondern der Entschließung und des Mutes liegt, sich seiner ohne Leitung eines anderen zu bedienen. Sapere aude! Habe Mut, dich deines eigenen Verstandes zu bedienen! ist also der Wahlspruch der Aufklärung (Kant in Renner, 1991: S.91 f.).
Habe den breitesten Verstand und nutze ihn in inkommensurablem Maße, soll für die vorliegende Arbeit der Ausspruch lauten. Dies soll so bleiben, selbst wenn du anderen Aufklärung verschaffst. Kläre jeden auf, wie er zu einem Max Weber werden kann.
Hat man es nur mit Aufgeklärten zu tun, sich selbst Leitenden, wird (gesellschaftlicher) Strukturzwang (durch einen anderen) nicht mehr definierbar, und so muss es den jeweils Aufgeklärteren ergehen. Diejenigen, die den Rahmen überlisten, Trittbrettfahrer und dergleichen, schaden dem Marketing der Aufklärung. Die Strukturierung der Aufgeklärten in mehr oder weniger mutigen (Denker-, Kämpfer-, Krieger-, Vernichtereinheiten etc.), sowie anhand ihrer unterschiedlichen geistigen Voraussetzungen in realen Sprech-, Denk- und Handlungssituationen, ist selbstverständlich.

Spezialisiert man die Frage der Aufklärung auf die Soziologie, sollen in dieser Arbeit mögliche einschlägige Strukturen und Architekturen derart variiert, geprüft und abgelehnt werden können, sodass eine effizientere und effektivere Forschungsprozedur plus Praxis als aktuell gegeben realisierbar erscheint. Dass es Optima hierbei geben soll, muss natürlich angenommen werden.

Im Sinne der Informatik wird versucht, die (ersten) Großen der Soziologie, Hobbes, Spencer, Compte, Durkheim, Marx, Weber etc., hier der Einfachheit halber durch Max Weber repräsentiert, in mehr oder weniger äquivalenten Simulationen, welche auf Kombinationen verhältnismäßig einfacher soziologischer Prozeduren und Bauteile (Studenten, Datenbanken, Arbeitsgruppen und dergleichen) beruhen, zu reproduzieren und zu vervielfältigen.
Hardware und Software sind logisch äquivalent, so die Informatik. Hardware sei versteinerte Software; jeder von der Hardware ausgeführte Befehl lässt sich von der Software simulieren. Jede durch Software ausgeführte Operation kann auch direkt in Hardware integriert werden, vorzugsweise nachdem sie ausreichend bekannt ist (Tanenbaum, 2001: S. 27). Diese grundlegende Konstellation soll für die Soziologie nutzbar gemacht werden können.

Fabuliert man weiter, und nimmt z.B. Marx als umgestülpten Hegelianer, denke man sich die großen Soziologen als Umkehrer der politischen Theologie und Lieferanten von funktionalen Äquivalenten zu Gott, Staat, Leviathan und andere (vgl. Wagner, 1993).
Oberflächlich gesprochen: ein kostengünstiger Simulant von Max Weber als Architekt oder Begründer der neuen Soziologie sei ein umgestülpter Durkheim, Weber etc., der die Bedeutung der Sozialwissenschaft von privaten Sprachspielen (z.B. das Wissenschaftsspiel) ins göttliche Ernsthafte, in ein dominantes Überlegensein, verzerrt, wie immer das auch gehen soll. Der umgekehrte Nochsoziologe ist kritischer als die kritischste Theorie und erweitert oder erhebt die funktionalen Äquivalente von Gott (der Christenheit) zu ihren Ursprüngen, indem sie zerlegt oder aufgehoben oder zurückgeführt werden. Wenn der erste Soziologe das Äquivalent von Gott liefert, der zweite höchstens das Äquivalent der ersten Soziologie etc., kann das Prinzip der Ausdifferenzierung nicht die Lösung sein.
Diese angedachte Umkehrung der Soziologie ist natürlich mit progressiven und nicht restaurativen oder nihilistischen Mitteln zu führen, ansonsten würde man ja nicht „verlustfrei komprimieren“, und die Umwertung der Werte in eine Abwertung verwandeln. Dieser allgemeine „Pseudocode“ soll operationalisiert in die Imitation eines künstlichen Max Webers einleiten können.

Diejenigen, die sich des Verstandes von Max Weber auf einem niedrigeren Niveau bedienen, seien die nicht aufgeklärten Aufklärer, die heute aktiven Soziologen.

Metaphorisch könnte dieser Vergleich auch wie folgt ausgedrückt werden:

- Schöpfer / Werk
- Aktiv / passiv
- Prinzipal / Agent
- Einzelner / Gruppe
- System / Umwelt
- Initiator / Imitatoren / executives
- Gründungsvater / Nachfahren, Praktikanten
- Manager / Angestellte
- Vater / Kind
- Autor / Publikum
- Verfassung / Gesetze
- Entdeckungs- und Verwertungszusammenhang / Begründungszusammenhang
- Rahmen / Zwischenrahmen
- System / Element
- Schlüssel zur Verschlüsselung / Rauschen
- Legislative / Exekutive
- Gehirn / Organe bzw. Körper
- Regel / Beispiel
- Inhalt / Form
- Was / Wie
- Einmalig / stetig
- Sprache / Automat
- Haus / Bewohner
- Außerordentlich / repetitiv
- Betriebssystem / Prozesse
- CPU / Subsysteme
- Multiprozessor / Singleprozessor
- Programmierer / Schaltkreise oder Bausatz
- Server / Client

Diese Auflistung geht in der Analogie bewusst etwas weiter, als vielleicht in der praktischen Soziologie vorgefunden; da diese z.B. durch Zwischenstufen oder realen ebenbürtigen Denkern und Soziologen eine komplexer verknotete Struktur etc. aufweist.
Geht man präziser in die Sprache der Informatik über, was in dieser Arbeit ausführlich statt haben wird, soll eine Approximation von Max Weber Leistungen oder komplexen MW-Automaten (was immer das hier bedeuten mag) durch einfachere Automaten, Strukturen, schnellere Busse, Chips etc. auf jeder Ebene der soziologischen Theorie und Praxis angegangen werden.
Eine Art höhere, aber auch ganzheitliche, soziologische Argumentationsart (flexible Intelligenz, Entscheidbarkeit, Problemlösungskompetenz, Beweisstrategie, die gegenüber „geradlinigen“ Abhandlungen, Theorienaneignungen, soziologischen Erzählungen und Romanen, Erklärungen, Verstehen etc. gestellt sind) könnte z.B. durch vervielfältigte Verschlüsselungsprinzipien, zusätzlichen Komplexitätsaufbau und anderen „Forschungs-logiken der Informatik“ (unangreifbar, oder nie falsifizierbar, nie verstanden, nicht faktorisierbar oder reduzierbar) zu konstruieren sein; dabei sollte es nicht möglich sein, zu zeigen, dass diese nicht auf Max Webers bzw. höchstem Niveau liegt; der Aufgeklärteste von allen ist der dialektische Aufklärer, einer, dem mehr Chancen und Möglichkeiten, Kanäle, Operationen etc. im Diskurs und überall zur Verfügung stehen. Er dient als Server, während andere als ebenbürtige Clients um ihn versammelt, untereinander eine Verbindung suchen und benötigen.
Soziologische Prozessoren, allgemein auch Architekturen weiter zu entwickeln und zu vergleichen, personelle und strukturelle „Überlegenheiten“ durch multivariate Leistungsmessungen operationalisierbar zu machen (über Publikationslisten, Projektmittel, Mitautorenschaften, Noten im Maximum hinweg), ist der Sinn dieser Arbeit.

Man könnte umgangssprachlich diverse Varianten von personellen (nebst strukturellen) „Überlegenheiten“ formulieren, die mehr oder weniger zur Entlastung oder Überlastung der Situation, zur Ausschließung aus dem Arbeitskreis und Diskurs führen können:

- Ich weiß, dass ich nichts weiß.
- Ich weiß ,dass du nichts weißt.
- Ich bin ein Narr, der den Weisen komplett überfragt.

- Ich bin der Weise, der jeden Weisen (außer sich selbst) überlastet.
- Ich bin die Gruppe von Weisen, die andere Weisen überfragt.
- Ich bin die Gruppe von Narren, die andere Weisen überfragt (zum Schweigen bringt etc.).
- Ich bin der Weiseste unter den Weisen (der echte Max Weber).

Praktisch wären neben Störprozessen z.B. Ideen und Lösungen, Anstöße, Änderungen, Verbesserungen, Wandlungen von theoretischen und praktischen Verläufen vorzugeben, in der Absicht, dass eine automatisierte (stupide) Produktion derselben den soziologischen Forschungsprozess bestenfalls auf simplen Entscheidungsfragen (richtig oder falsch) reduzieren, kostengünstiger anbieten und somit beschleunigen kann. Probleme, die auf zu viele Denker, Autoren, Literatur, Daten etc. zurückzuführen sind, durch Theorienlöser oder –prüfer im Sinne von SAT-Solvern hinweg zu simulieren, und Überforderungssymptome (verteilter Systeme) durch Wiedereinführung von „Neumannsystemen“ zu lindern, soll im gleichen Sinne als denkbarer Ansatzpunkt geprüft werden. Dabei wird davon ausgegangen, dass in diesem Kontext der Spruch, Hardware sei gleich Software, genauer: Hardware und Software sind logisch äquivalent, ebenfalls realisierbar ist (Tannenbaum, 2001: S.27).

Eine grundlegende Analogie ist die „absichtliche“ Verkomplizierung oder Verschlüsselung des Arguments, in der Form, dass z.B. die Zeit für eine Reduktion auf voneinander unabhängigen, verstandenen Elementen unverhältnismäßig steigt (etwa durch wachsende Literaturdatenbanken, Ausdifferenzierung von vielen Forschungsergebnissen), damit die Berechnung, Lösung, Falsifikation, oder der Widerspruch und Gegenbeweis auf gleicher Ebene annähernd unmöglich gemacht wird, oder zumindest der Vergleich zwischen dem Argument und einem anderen (siehe z.B. den Fragebogen im Anhang dieses Arbeitspapieres). Man kann sich selbst die Grazie von Adornosätzen denken, welche allein durch „Schnörkel“ ihren Sinn beinah unangreifbar emporheben. Das Rezept der Adornonachfolger könnte lauten, die Sätze imitativ beizubehalten, und sie vielfach minimal zu variieren, ohne ihr „Muster“ unwiderruflich zu zerstören.

In der Programmierung oder Berechnung könnte man analog etwa (rekursive) Anweisungen wie etwa: bedingt dass x gilt und xy und xyz etc…, (xy hat gesagt, dass y richtig ist, und…);

oder erste Schleife,

darin zweite Schleife,

darin dritte Schleife…, ansetzen, die das Terminieren oder Decodieren und Verstehen zumindest verzögern oder gar erschweren.

Der Dümmste scheitert im Diskussionsspiel, nicht jedoch der weniger Dumme. Damit kann der Proponent unschlagbar sein, aber auch des Diskursfeldes enthoben, da die anderen Teilnehmer oder Gegner offensichtlich nicht die gleiche Chance haben, sein Argument zu knacken. In Fragen, die mehrere Fachgebiete oder hohe Interdisziplinarität übergreifen, kann z.B. der Nur-Soziologe nicht die Ressourcen zum Gegenbeweis finden oder recherchieren.
Man darf oder muss in diesem Sinne nicht 100 ähnliche Artikel lesen (im Sinne von Stringsuchen), die keine Lösungen bieten, sondern nur die Lösung schreiben (oder entscheiden), das ist einleuchtend. Hierzu soll die Einführung von bestimmten Strukturen oder Architekturen Problemlösungen beschleunigen.

In der theoretischen Informatik ist folgende Unterscheidung von Sprachklassen üblich (Socher, 2008: S.115):

Sprachklasse	Name	Akzeptierender Automat	Erzeugende Grammatik
Typ 0	Rekursiv aufzählbar	Turing Maschinen	Beliebige Grammatik
Typ 1	Kontext sensitiv	Linear beschränkte TM	Kontext-sensitive Grammatik
Typ 2	Kontextfrei	Nd. Kellerautomaten	Kontextfreie Grammatik
DFK	Deterministisch kontextfrei	Det. Kellerautomaten	LR(k)-Grammatiken
Typ 3	regulär	Endl. Automaten	Reguläre Grammatik

Sogenannte Entscheidbarkeitsresultate für dieselben Sprachklassen lauten (Socher, 2008: S.116):

Sprachklasse	Wortproblem	Leerheits-problem	Endlichkeits-problem	Äquivalenz-problem
Typ 0	nein	nein	nein	nein
Typ 1	ja	nein	nein	nein
Typ 2	ja	ja	ja	nein
DFK	ja	ja	ja	?
Typ 3	ja	ja	ja	ja

Zusammengefasst: je einfacher oder regulärer die Problemsprache bzw. die Problemdefinition, desto einfacher die Lösung bzw. deren Sprache und Ausdruck.

Dazu werden die sogenannten Abgeschlossenheitseigenschaften der Sprachklassen, mit dem einzigen Hinweis auf Unterschiede, ohne weitere Erklärung, gezeigt (Socher, 2008: S. 115):

Sprachklasse	Durchschnitt	Vereinigung	Komplement	Konkate-nation	Kleene-Stern
Typ 0	ja	ja	nein	ja	ja
Typ 1	ja	ja	ja	ja	ja
Typ 2	nein	ja	nein	ja	ja
DFK	nein	nein	ja	nein	nein
Typ 3	ja	ja	ja	ja	ja

Anhand eines „fehlerhaft“ formulierten bzw. in einer „Hochsprache“ beschriebenen Beispiels wird das sogenannte Halteproblem, welches nicht entscheidbar sein soll, illustriert:
In einem Dorf gibt es einen Barbier, der die Männer des Dorfes rasiert. In seinem Laden hängt ein Schild: Ich rasiere genau diejenigen Männer des Dorfes, die sich nicht selbst rasieren.

Rasiert der Barbier sich selbst (Socher, 2008: S.145)?
Ist der Barbier nun ein Roboter, oder kein Mann und etwa ein Kind (in manchen Sprachen ist diese Fragestellung als Scherzfrage bekannt, warum ein Chirurg seinen Sohn nicht operieren will, wobei ersterer (Beruf ohne weibliche Form) nicht sein Vater, aber seine Mutter ist), oder man ergänzt das obige Schild, in dem es heißt: …, die sich nicht selbst die Beine rasieren… etc., scheint es kein internes Problem mehr zu geben. Wer genau liest, wird im ersten Satz lesen, dass der Barbier auf jeden Fall Männer rasiert. Dann wird künstlich ein Widerspruch eingebaut, dass nur diejenigen rasiert werden, die sich nicht selbst rasieren. Man kann auch fragen, wenn sich alle selbst rasieren, was tut der Barbier beruflich, etc.

Die (geschickte) Formulierung der Sprachklasse ist es, die ich wähle, welche das Halteproblem intern entscheidbar oder nicht entscheidbar zeigt, oder in Worten und in regulären Ausdrücken: so wie ich ein Problem (bewusst) formuliere, so kann ich es lösen. Die Formulierung oder die Sprache der Automaten (die Grammatik, die Alphabete, die Terminalsymbole etc.) ist das A und O der Berechenbarkeit, der Informatik, der Wissenschaftlichkeit, der Problemfindung und Lösung; das dürfte keinen überraschen, der sich selbst rasieren kann.

Anstatt also diverse Wissenschaftstheorien (der mittleren Reichweiten, Konstruktivismus, Kulturabhängigkeit der Beobachtung etc.) samt der notwendigen Kritik daran, an sich, differenziert, oder in Kombination mit expliziten Thesengefügen „objektiv“ zu betrachten, oder nur die einzelnen Soziologien, Modelle und deren Konsistenzen gegeneinander optimieren zu wollen, sollen die einzelnen Vertreter bzw. (lebende) Produzenten persönlich gekontert (falsifiziert, auf Vorliegen geistiger Voraussetzungen geprüft, in ihrer Leistung gemessen) werden, um aus einem großen endlosen Optimierungsproblem ein „leichteres“ oder viele kleine zu machen (kein Laufzeitproblem, Berechenbarkeit liegt vor, Teilprobleme lösen etc.).
Eine Vorgehensweise dazu könnte lauten: jedes vorgefundene Argument (des Automaten xy) an sich zu nehmen, Komplexität oder Präzisionsanforderungen (automatisch, im Sinne von +1,+2,+3 etc.) solange einzuführen, die es für seinen beschränkten Verwender bzw. den Diskursraum undenkbar (falsch oder unzutreffend) machen; in der zweiten Variante, Komplexität abzubauen, zu vereinfachen, zu reduzieren als Lösung, oder reduzieren, um

dann zu lösen, then goto (siehe dazu die Algorithmendesigns und Konzepte weiter unten). Eine schwer lesbare Formel kann durch einen Tippfehler (und durch viele andere) unbrauchbar gemacht oder erschwert werden; diesen zu finden, ist sehr umständlich, diesen zu setzen, weniger.

Was in der Informatik als P bzw. NP-Klasse der Probleme bezeichnet wird, heißt umgangssprachlich:
Ich selbst (Milko Mumdzhiev) errate nur die (vielfachen) Lösungen, kann sie, da ich überfordert bin, aus mir selbst heraus nicht anders prüfen, als zu reduzieren, und dann zu hinterfragen oder entscheiden. Die spezielle Antwort, Gegendarstellung, These oder Theorie, das bestimmte Argument, die Teilaussage kann ich kontern, darauf kann diese extern gekontert werden etc. (etwa ob man für Paneldaten xtreg oder regress nehmen sollte).
Eine Verteilung der Systeme, der Aussagen als Diskursraum zwecks Datenfluss ist eine Alternative, jedoch innerhalb von Arbeitscluster. Man könnte pessimistisch paraphrasieren: ich halbiere und halbiere und halbiere… in der Summe, also 1/2+1/4+1/8…., wobei die Konvergenzen von vornherein feststehen.
Dafür soll aber der vorgefundene öffentliche Diskursraum und private Sprachspieler in neuen Variationen konzipiert, und die Ressourcen anders oder besser verteilt werden. Welche Möglichkeiten zur Verfügung stehen könnten, soll im Laufe dieser Arbeit vorgeschlagen werden.

Im folgenden sind mehr oder weniger automatisierte (und oft banale) Schlussfolgerungen zum Kontext (als Auswahl aus einer Vielzahl anderen, bereinigt) vorgestellt, die leicht und schnell erstellt (dabei findet eine sogenannte Parallelisierung oder Bastardisierung der einzelnen Elemente statt, die immer mehr zunehmen kann; außerdem gibt es mehrere Lesarten der Schlüsse, die von Leser zu Leser variieren), durch noch leichtere Fehlerkorrekturen abgelehnt oder zumindest verdächtig gemacht werden können. Dies soll der Leser für sich nachvollziehen, und die Überlassung von Ansätzen durch die Floskel bedenken. Aus meiner (M.M.) Perspektive sind die Lösungen also geraten oder „fingiert“; um sie (nach einer ersten Bereinigung) zu prüfen, müsste ich über mich selbst hinauswachsen (NP).

2. Die geisteswissenschaftliche Sichtweise

2.1 „Fingierte“ Hypothesen

Es gibt kein Perpetuum Mobile (also kein (energieverbrauchendes) System, das ohne Energiezufuhr weiter funktioniert, welches keine Energie verbraucht; Ausfuhr gleich Einfuhr; das Objekt besitzt eine endliche Energie, die es ausmacht, und zerfällt bei Energieverlust oder -freiwerden; eine Tautologie; Input- / Outputgrammatik)
Die Logik ist ein solches (Regel-)System.
Die Logik funktioniert an und für sich nicht (z.B. die digitale Logik).

Logisches, nicht zufälliges Denken ist auf logische Schlüsse etc. gegründet bzw. kombiniert, so die gesamte Wissenschaftstheorie (trivial; diese Schlüsse werden über entsprechende Schaltungen in Hardware oder digitale Logik übersetzt; Software gleich Hardware). Prinzipien der Thermodynamik sind über logische Schaltungen denkbar und simulierbar, denn wenn ich lückenlos verstehe, habe ich alle Schaltungen dazu, und verstehe und reproduziere gedanklich bzw. meine Gedanken und Schaltkreise imitieren die Prinzipien, und umgekehrt.
Diese physikalisch ausgedrückten Prinzipien machen im übertragenen Sinn ersichtlich, warum Blasen (Geld, Wissen etc. als verteilte Systeme, auch in Form von Kettenreaktionen, Datenfluss) nicht mehr durch rationale Energie (Vernunft, real existierende Schaltungen) gehalten werden können; denn die Blase oder der Problemkreis, das Phänomen wächst schneller als durch die vorgegebene Zahl der Schaltungen, der Akteure, der Hedgefonds etc. gerechnet oder realisiert werden kann. Auch der Einsatz von Hebeln in dieselbe Richtung verändert das Problem nicht. Die Umkehrung der Hebel aller (Geldmengenreduktion, Destabilisierung des Geldwertes oder Nutzens, Fluktuationen, Bandbreitensenkung, Trennung der Systeme etc.) lässt die Blase platzen; letzteres als Konstante würde nicht nur die Wirtschaft in der Praxis, auch in der Theorie, als Wirtschaftssystem undenkbar machen, man müsste praktisch in die Geistigkeit eines vorkapitalistischen Systems, eine Rückkehr in die Zeit, eine Umkehrung der Soziologie oder Ausdifferenzierung real machen können. Statt also das Kind mit dem Bade auszuschütten, lässt man es darin ertrinken.

Fortlaufende Beobachtung kann in einem geschlossenen Raum die Objekte finden, durch Wissen, Aufsätze etc. sehen und zählen; wenn die Zahl der Zusammenhänge (geistiger Natur) oder Wenn-Dann-Thesen (als Wissen) unverhältnismäßig durch das Thesensammeln steigt, ist das ein nicht lösbares Wissensproblem, ein Laufzeitproblem.
ALLE denkbaren (aufzählbaren) Zusammenhänge, die einzeln gesammelt werden können, steigen durch das Sammeln oder Erkennen von Zusammenhängen selbst überproportional in der Zahl und Prüfungslaufzeit an, und können deswegen niemals die Wissenschaft beenden, sondern sie nur unbedeutender machen. Da die reale Leistung von Wissenschaft als Erkenntniskollektor natürlich begrenzt ist, muss die Komplexitätsklasse von Beginn an manipuliert oder geschätzt werden (Spezialisierungen finden statt, um aus großen Thesenbereichen kleinere zu machen, Clusterbildungen, Untercluster, Ausdifferenzierungen etc.). Korrelationsmatrizen bzw. Tensoren bedeuten dies.

These: es gibt unendlich viele Primzahlen.
These: es gibt unendlich viele negative Primzahlen (z.B. teilbar durch 1 und –1, durch sich selbst, ohne positiven oder negativen Rest)
These: es gibt unendlich viele Thesen.
These: es gibt keine Thesen.
These: es gibt keinen Beweis.
These: es gibt unendlich viele Beweise.
These: es gibt unendlich viele Gegenbeweise.
These: es gibt unendlich viele richtige Beweise, die nicht gezeigt werden können, etc.
Diese darzustellen, ist natürlich die Aufgabe, die reale Wissenschaft für reale Objekte nicht realisieren kann.

Je mehr Objekte bzw. Produkte, desto überproportional mehr Zusammenhänge.
Die Anzahl der Elementarteilchen bzw. der Bestandteile des realen Universums ist nach oben hin beschränkt. Es gibt aber mehr Einzelwirkungen zwischen den Elementarteilchen, als Elementarteilchen. Wirkungen können nicht durch Elementarteilchen dargestellt werden, da es diese in der Mehrzahl nicht geben kann. Ich habe die (Einzel-)Realität, die Tatsache, und

ich habe den Zusammenhang im Diskurs. Diese Relation muss irreal oder ideal sein, denn sie geht materiell über die Realität hinaus.

Eine (physikalische) Formel ist auch dann richtig, wenn man Zahlenwerte einsetzt, die im ganzen materiellen Universum nicht existieren (als Elementarteilchen, Zeit etc.)

Ein Koordinatensystem habe 2 Achsen, und ein Zusammenhang soll gezeigt werden.
Würde man die Achsen nicht nur andeuten, sondern ganz zeichnen oder beschriften wollen, wird jeder Speicher (neben der Laufzeit) irgendwann überlastet. Man weiß ja, worum es geht, heißt es dem entgegen; außerdem können x, y nicht unendlich werden, Realitäten sind endlich. Inhalt und Darstellung passen zusammen. Ich wähle also nur einen Abschnitt, um ihn auf den Monitor zu zeigen, weil der Zusammenhang von allen mit allem z.B. die Matrizenrechnung überproportional überfordert. Es gibt per Definition mehr denkbare Zusammenhänge (diese können natürlich 0-wertig oder noch nicht messbar sein) zwischen den Objekten als die Anzahl der Objekte selbst. Die Struktur ist das Problem. Ich erfinde die Ceteris-Paribus-Bedingung. Ich bestimme die Präzision bzw. die Endlichkeit oder Möglichkeit des Verstehens als geistige Produktion.
Ceteris-paribus-Bedingungen (sonst alles gleich) macht es berechenbar, aber alles kann so nicht programmiert werden, sondern nur x,y,z, = 0,1,oder = a (in der Soziologie: z.B. alle Zwecke, alle Systeme, alle Akteure etc.).

Analog kann die Argumentation für einen Schätzer im Paneldatenkontext gedacht werden. Die Lösung oder Behandlung einfacher Probleme, die ohne Zusammenhang stehen, ist die einzig denkbare oder aussprechbare. Diejenigen Forscher, die Panelmodelle schätzen, sind keine Groß-Theoretiker, oder haben kein allgemeines Verständnis, jedoch zahlreiche Methodenkenntnisse der Branche.

Film bzw. Photografie plus Kontinuitätsargument (Schurz, 2006: S.60):
Sind Theorien empirisch mehr oder weniger adäquat, dann stellt ein Film eine erstklassige Theorie samt Prüfung (des alles darin gesehenen, und nicht mehr oder weniger, der Pixelstruktur (unausgesprochene Muster, Cluster, Struktur, Abhängigkeiten, Korrelationen,

Zusammenhänge), der Frames, der Pixel selbst etc.) dar. Die Beobachtung (ohne durch Objekte sehen zu können; mit Röntgen; mit Vergrößerung etc.) ist immer die gleiche. Es handelt sich dabei aber nicht nur um eine Theorie der Pixel (und weiteren angeschlossenen „Multiplikatortheorien“).
Die Gesellschaft ist die Soziologie. Wie ist es, für den Seite über Seite, schrittweise denkenden Soziologen, Gesellschaft zu sein (NP)?
Ob etwas korreliert oder nicht, zeigt der Versuch, die Beobachtung, das Hinstellen von x zu y. Wie wird Korrelation gemessen? Nicht zwischen x und einem Apfel, zwischen x und y.
Ich benötige also mehr Teleskopie, neue Maschinen, neue Schaltungen, die das, was ich prüfen will, darstellen (sind, realisieren, meine Beobachtung kontinuierlich auf einen gemeinsamen Nenner bringen etc.).

Ist der Schluss P / K gültig, dann füge K deinem Wissenssystem hinzu; leider wird dadurch das System im Hinblick auf neue Schlüsse überproportional komplexer (die Schlüsse müssen strukturiert werden). Es wird entwertet, und zweckhafter, ausdifferenzierter. Der Wald wird zu Bäumen. Das unteilbare Atom zu Elementarteilchen zu machen ist bereits ein ANALYSE-Algorithmus, den die Empirie nicht bestätigen muss (1+1+1+1+1…: 100-1-1-1…).

Es gibt keine Maschine, die das existierende Universum simulieren kann, die (auch ein Teilchen) mehr als das Universum, aber nicht Gott ist.

Unabhängigkeitsargument von Toulmin (Schurz, 2006: S.63):
Was ich sehe, bleibt gleich, ob ich zwei Hintergrundtheorien dazu denke, oder nicht (Sonne wandert über den Himmel). Die Wahrnehmungseindrücke bleiben gleich, bzw. nach wie vor sehen wir die Sonne tagtäglich in einem Bogen über den Himmel wandern, ob man Heliozentriker ist, oder nicht. Himmel meint hier Erde, der vorgelegte Ausdruck von Schurz (2006), der das U-Argument benutzt, ist die geozentrische Perspektive, sein Seherlebnis ist also doch geozentrisch (formuliert und realisiert). Das hat Schurz (2006) nicht ausgeführt, im Jargon, weil er nicht die Rechenzeit zur Verfügung gestellt hat, bis zum Ende zu kommen. Denn egal wie präzise er geht, seiner Präzision fehlt der Grad n+1. Man kann sagen, beide

Massen haben eine Beziehung zueinander etc., das ist aber ein anderes Argument, ein (soziologisches) Seherlebnis.
Statt hier weiter zu grübeln, das soll Schurz auf eigene Rechenzeit tun, warte ich als Opponent ab, wiederhole gegebenenfalls die argumentative Vorgehensweise (in einem Schritt). Während Schurz (2006) an seine Kapazitäten (z.B. im Buch) kommt, tue ich dies nie.

Das stetige weiße Rauschen in Strahlungsmessungen (oder bei Abwesenheit von Strahlung) deutet auf den Ursprung des Universums in einem Big Bang hin. Da Messungen einen Teil in Wahrheit, einen Teil in Fehlerterme ausgeben, stellt das Rauschen nur den Fehlerterm bei jeder Messung dar, und nicht etwa den Hinweis auf den Big Bang.

Soll der Begründungszusammehang „wertlos“ sein, kann Verwendung oder Interpretation daran nichts ändern. Egal was ich begründe, ich kann es beliebig (auch im Zirkel) verwenden (per Definition). Die und die Schachstrategie ist der anderen im Spiel überlegen, nicht wirklich beim Baden oder Haareschneiden.

Unter welchen Bedingungen wachsen die Literaturverzeichnisse diverser Forschungs- oder Qualifikationsarbeiten und wie? Welche Zusammenhänge haben die einzelnen Quellen? Wenn ich konstante Spezialisierungen zulasse, dann haben Literaturverzeichnisse etc. in etwa die gleiche Länge.

Was hier angedeutet wurde, ist eine stupide Vervielfältigung von allgemeineren Hypothesen außerhalb einer bestimmten Forschungsarbeit (die Theorie- und Empirieabschnitte zu einem Thema enthält), welche, weil sie stupide ist, in alle Fachbereiche steigen kann, und auf übergreifender Ebene (nach angenommener kostengünstiger Prüfung) versuchen soll, bestimmte Forschungsalgorithmen, die auf „fehlerhafte“ Annahmen beruhen können, und dadurch in die Endlosigkeit oder falsche Richtung gehen, im Sinne von Geniestreichs abzukürzen, sofern diese nicht an und für sich von Bedeutung sind. Bleibt man auf der Ebene der Wissenschaftstheorie, oder geht man hierarchisch die Ebenen der Fachbereiche und Unterfachbereiche hinunter, eine Auffächerung im Sinne von Baumstrukturen abzugehen impliziert ein Auf und Ab über die Knoten. Man kann immer wieder bei den europäischen

Vorsokratikern anfangen etc., das ist aber selten der bestellte Sinn der Sache, und ein (mehrfach) unendliches Unterfangen an sich; etwas anderes ist der zweckorientierte, alltägliche Beruf. Ob die täglichen Repetitionen, die jährlichen, die generativen etc., nicht die selben eigentlichen darstellen, bewährt sich in kurzfristigen realen Sprechsituationen.

2.2 Theorienleistungen, die Soziologie und der Max Weber Automat

Was Theorien im allgemeinen und im Kontext der Gesellschaftswissenschaften zu leisten haben, wird kurz vorgelegt.

2.2.1 Theorienleistungen (*und kursive Hinweise oder Vergleiche von mir, M.M.)*:

1. Systemcharakter: man versteht die Theorie, wenn man das Zusammenwirken der Hypothesen betrachtet (durch konjunktive Vereinigung mehrerer Hypothesen wird empirischer Gehalt erzeugt). *Man könnte z.B. die Clusterlösung betrachten.*
2. Empirische Kreativität: zwei oder mehr theoretische Hypothesen erzeugen neuen empirischen Gehalt, der über die Summe bzw. Vereinigung der empirischen Gehalte dieser einzelnen Thesen hinausgeht. *Die Gruppe ist mehr als die Summe der Bestandteile, das Individuum bekommt (etwa in SPSS) die Variable Gruppenmitglied.*
3. Globalität und Vereinheitlichungsleistung; Globalität: eine gute Theorie erklärt qualitativ verschiedene empirische Phänomene, Vereinheitlichungsleistung: einheitliche Erklärung qualitativ verschiedener Phänomene, die damit viele empirische Hypothesen auf wenige theoretische Prinzipien zurückführt. *Beispiel: Binärbäume 1, 0, 10, 11, 01, 00, 101 etc.* (Schurz, 2006 S:188 f.).

2.2.2 Methode der Konstruktion von Theorien (nach Maturana, in Esser, 1993):

Als Wissenschaftler machen wir wissenschaftliche Aussagen. Diese Aussagen werden durch das Verfahren validiert, das wir gebrauchen, um sie zu erzeugen: durch die wissenschaftliche Methode. Diese Methode kann durch die folgenden Operationen dargestellt werden:

- Beobachtung eines Phänomens, das als zu erklärendes Phänomen angesehen wird;
- Entwicklung einer erklärenden Hypothese in Form eines deterministischen Systems, das ein Phänomen erzeugen kann, welches mit dem beobachteten Phänomen isomorph ist;

- Generierung eines Zustandes oder Prozesses des Systems, der entsprechend der vorgelegten Hypothese als vorhergesagtes Phänomen beobachtet werden soll;
- Beobachtung des so vorhergesagten Phänomens (Esser, 1993: S.54 f.).

2.2.3 Modellierung sozialer Prozesse:

1. Es sollte für die Formulierung der Brückenhypothesen, der Handlungsgesetze und der Aggregationsregeln nicht allzu viel an Information über „einzelne“ Individuen erforderlich sein, um bestimmte strukturelle Vorgänge zu erklären. Also: es muss – zunächst – ausreichen, Modelle von typischen Situationen für typische Erwartungen und Motivationen und für typische Muster von Handlungsalternativen zu erstellen. Eine Modellierung, die weniger Parameter erfordert, ist daher einer – ansonsten etwa gleich erklärungskräftigen – komplexeren im Prinzip vorzuziehen.
2. Die Formulierung von Brückenhypothesen (und damit: die Modellierung der Logik der Situation) muss auf eine möglichst unkomplizierte Weise möglich sein.
3. Erkenntnisgewinne in den Theorien des Handelns und des Verhaltens (in den damit speziell befassten Disziplinen) müssen (im Prinzip) Eingang in die verwendeten Theorien zur Logik der Selektion finden können.
4. Es muss (im Prinzip) möglich sein, den Grad der Vereinfachung in der Handlungstheorie nach Bedarf zu verringern und stärker realistische Annahmen in die Modellierung aufzunehmen.
5. Die verwendete Handlungstheorie selbst soll – sowohl als psychologische Theorie wie im Zusammenhang mit ihrer soziologischen Verwendung – gut bestätigt sein und systematisch berücksichtigen, dass Menschen findig, kreativ und initiativ sind.
6. Die Parameter der Handlungstheorie sollen zunächst möglichst präzise benannt sein, und es muss eine explizite funktionale Beziehung zwischen den Parametern und der abhängigen Variable – der zu erklärenden Handlung – angegeben werden. Ansonsten lassen sich weder präzise und einfache Brückenhypothesen formulieren, noch eine „Logik“ der Selektion von Handlungen angeben (Esser, 1993: S.138).

2.2.4 Empfehlungen für die Modellierung sozialer Prozesse (Lindenberg und Wippler, in Esser, 1993):

1. Beginne mit möglichst einfachen und unkomplizierten Annahmen – über die Brückenhypothesen, über die Handlungstheorie und über die Transformationsregeln.
2. Vertiefe die Annahmen nur schrittweise und nur dann, wenn alle anderen Möglichkeiten – auch die einer weiteren Vereinfachung!- ausgeschöpft und fehlgeschlagen sind.
3. Orientiere dich bei der abnehmenden Abstraktion erneut am Prinzip der Einfachheit und daran, ob es für die neuen Annahmen bereits fertige Modelle in anderen Wissenschaften gibt (wie Modelle der Diffusion, des Wachstums, zyklischer Schwankungen, des Gleichgewichts von Märkten oder „Spielen“, Anpassungsmodelle, Verhandlungsmodelle, Marktmodelle u.a.).
4. Versuche immer wieder zu den einfachen Annahmen des Ausgangsmodells zurückzukehren und die Anomalien in dem ursprünglichen Rahmen aufzufangen. Wissenschaft bedeutet vor allem die Integration des Neuen in bereits Bekanntes und nicht die Aneinanderreihung von unverbundenen „Effekten“ und „Phänomenen“ mit begrifflichen Umschreibungen derselben.
5. Hände weg von der Komplizierung der Handlungstheorie, bevor nicht alle anderen Möglichkeiten ausgeschöpft sind.
6. Modelliere so einfach wie möglich und so realistisch wie nötig (Esser, 1993: S.139 f.).

2.2.5 Schlussfolgerungen für die Soziologie

Die Einfachheit der Formulierung wird als Grundgesetz gefordert, ebenso die der Sprache.

Wie weiter oben berichtet, stellt die Einfachheit der Sprache bereits den Problemkreis fest; die Vorstrukturierung der Themen, der Formeln, der Definition von Quatsch und Albernheiten im Vergleich zur ernsten Regel, ist der Lieferant der Lösungen, oder der umgestülpten Problemsprache. Ein (weitergehendes) Beispiel zu obigem Punkt 3 ist die Beschreibung diverser Phänomene bzw. des Theorienverbandes der klassischen Mechanik (Schurz, 2006: S.180).

Aktio gleich Reaktio, Schwerkraft, Reibungskraft, Federkraft, Impuls und Energieerhaltung, Hebelwirkung, Antrieb, Auftrieb, Widerstand, Reflexion, Brechung, Masse: abgesehen von der Formalisierung (doch selbst diese mitgenommen) können die Begriffe (bzw. das Denken

mit denselben) problemlos in der Soziologie, Ökonomie, Literaturwissenschaft etc. verwendet werden: der Zwang, die Motivation, das Wachstum, meine nicht intendierten Handlungsfolgen, die doppelte Kontingenz, das Gleichgewicht von Angebot und Nachfrage nach xy, der Leverageeffekt, Medien, Schwellenwerte etc.
Die Objekte sind natürlich andere, das wurde z.B. bei Weber durch die Definition des sozialen Handelns für die Soziologie, oder in der Physik durch die Betrachtung der als unbelebt definierten Objekte festgehalten.

Können Hypothesenautomaten und ihre „Helfer“ den Verstand von MW ersetzen oder approximieren, und Kriterien der Wissenschaftstheorie erfüllen? Und wie gut bzw. effizient durch unterschiedliche Architekturen? Wie viele Hilfskräfte müssen als Input- / Outputfaktoren herhalten? Das zu viele Ergebnisse-Problem als das verteilter Systeme soll aus einem Diskursmodell in einen gesteuerten Denkapparat verwandelt werden. So wie die ersten Computer (die noch einen Menschen in sich verstecken mussten) sich der vorgefundenen Rechensituation bzw. dem Bedarf anboten, und den Markt erst eröffneten, um ihn später auf anderer Ebene auszudifferenzieren, so soll einem verteilten Forschungsprozess eine erstrangige Neumann- bzw. Webermaschine zur Seite gestellt oder aufgedrückt werden, um diesen selbst zu automatisieren.
Diverse (architektonische) Vorüberlegungen, die (von der Informatik) bereits geboten sind, sollen in die Soziologie bzw. deren Wissenschaftstheorie (re-) investiert werden.

A.
Feste Schaltungen erfüllen die ceteris paribus Bedingung (wenn gestern so, dann heute auch so).
Ansatz: Alle bisher beobachteten x sind x, also ist das nächste x auch ein x. (Beginne mit der Tautologie, und gehe um 1 (des Typs) nach rechts auf dem Band) (Trick im Trick, es gebe ein nächstes x).
Besser: Alle bisher beobachteten xn sind xn, also ist (das nächste, ein anderes) xn (+1) auch ein xn (+1). Die x sind gleich, die Indizes (z.B. Eigenschaften) variieren.

Gesetze „gelten“ dann, wenn Allgemeinheit, also Merkmalgleichheit oder Ähnlichkeit vorliegt (Regel für alle). Perfekte unabhängige (aufgeklärte) Individuen kennen keine Gesetze (bzw. Korrelationen, Typisierungen, höchstens Indizes).

B.

Man sehe das offensichtlich unsinnige Beispiel: wenn Hitler die A-Bombe gehabt hätte, dann Mondmasse gleich 0. Schrittweise (denkbare) Reformulierung macht es jedoch interpretierbar:

Wenn Hitler (und Gegner) einen totalen Nuklearkrieg entzündet und niemand überlebt hätte, wäre die Mondmasse nicht messbar (von niemanden, da niemand überlebt hat). Sozialisationsregel: stelle keine schwachsinnigen Behauptungen auf, wenn du nicht gescheit genug bist, sie klug oder richtig aufzustellen bzw. zu lösen.

C.

Ich habe hundert bivariate Wenn-Dann-Datenpunkte, und muss regressieren. Fortlaufende Reduktion der Datenpunkte bis hin zu einer Linie soll die KQ-Schätzung erleichtern.

Sind diese Datenpunkte in alle Richtungen gesetzt (im Sinne der Forschungsrichtungen), bekomme ich eine Wolkenstruktur und entferne mich vom linearen Zusammenhang. Innerhalb der Wolke kann ich sehr wohl Linearitäten fixieren. Ich führe eine Mustererkennung der Linie durch, und lege sie fest. Ich clustere zur Sicherheit im Vorfeld meine Ergebnisse, Linien, Theorien, Thesensammlungen, Probleme, Problemklassen etc. Die Gesamtheit aller Daten (bzw. Chaos) kann keine Rest-Probleme enthalten, und keine Lösungen. Ich reduziere dieses Chaos durch Sinn oder Musterdefinition, durch Fixierung, Repetition etc.

Wenn ich alle Daten habe, habe ich (und brauche) keine Zielfunktion mehr (Restriktionen, Optimierungsproblem). Die Materialisierung der Thesen ist deren Lösung.

D.

Es macht Sinn, zu sagen:

Beweise den Holocaust.

Es macht allgemein (noch) keinen Sinn, zu formulieren:

Wenn A, dann geht die heutige Arbeitslosigkeit zurück (Eine Art Bastard Hypothese, Theorie; fehlende Zwischenschritte für das Zusammenhängen, die selbst das Zusammenhängen sind etc., etwa A ist eine Konjunkturmaßnahme auf Kredit, nicht A = die Maus beißt keinen Faden ab).
Ich clustere die Objekte von vornherein so, dass sie innerhalb der Gruppen identisch sind (sonst 1 Objekt = 1 Gruppe), dass ich mindestens 2 Objekte oder Sachverhalte habe, die (fast) identisch sind, also eine Korrelation oder Ähnlichkeit im Verhalten, Schwingen etc., einen Zusammenhang messbar machen, und damit meine Hypothese von selbst bestätigen oder aufzustellen ermöglichen. Gegenteiliges:
Beweise den Widerspruch widerspruchslos. Leite 1 aus 0 ab.
Beweise den Beweis oder zeige die Richtigkeit, die Schaltung.

E.
Thesen oder Zusammenhänge (Korrelationen) im sinnvoll vorgegebenen Datenrahmen bestätigen ist durch if then Anweisungen darstellbar, und eine Ausgabe oder Ergebnis der Prüfung (z.B. signifikante Korrelationen, Unterschiede usw.) durch einen Automaten (SPSS) berechenbar.
Bsp. zu Ausländerfeindlichkeit:
Übertragung in SPSS Syntax:
If (v1(Wohnort) = 1(Ost) and v2 = 1(arbeitslos) and v3(Bildung) =4 (Hauptschule)) and … and… and…
Durch diese (fortgesetzten) Konditionale wird man üblicherweise eine sehr geringe Zellhäufigkeit (selbst bei einer Totalerhebung) bekommen, Zusammenhänge mit Variablen wie Linksrechtseinstufung etc. auf individuelles Verhalten (n = 1) reduzieren, und damit natürlich außerhalb der aggregierten Ebene und außerhalb der Diskussion einer Regel oder These (Regel gilt per Definition für mehrere Personen) bleiben. Soziologie behandelt ja Gebilde; Individuen, die gleiche Motivationen (Nutzenfunktionen) haben, werden als Gruppe (mit gleichen Elementen) behandelt.
Praktisch ist also ein Mittelwertvergleich über den Faktor Ost / West machbar (berechenbar), diskutierbar (z.B. mit zweitem Konditional Linkswähler, Abitur etc.) und damit denkbar,

erweiterbar, kritisierbar. Das macht bereits der Soziologie-Diplomand mit SPSS, sollte aber automatisiert und schneller laufen.

F.

Zusammenhänge aus der Arbeitssoziologie, Bildungssoziologie, Familiensoziologie sind für das Individuum als Trajektorie ohne Fehlerterm, auf aggregierter Ebene trivialerweise mit großer Streuung zu fixieren; das dürfte niemanden überraschen.

Welche soziologische Regel gilt, wenn ich alle denkbaren Daten zu allen Personen, Motivationen etc. vorliegen habe? Welche Struktur ist notwendig, wenn ich alles haben kann? Diejenige (subjektive) Denkregel, die (objektive) Komplexität reduziert, Ebenen findet; auf alle Daten an sich, nicht auf minderwertige Daten, verzichtet.

G.

Warum gibt es keinen breiten Sozialismus in der USA? Warum gibt es keine Revolution in Deutschland? Warum gibt es keine Götter als Ministerpräsidenten? Warum gibt es keine Götter, die nach Deutschland einwandern? Warum gibt es die und die englische Sozialstruktur nicht in Frankreich? Warum ist die Unterschicht nicht Oberschicht? Warum gibt es keine Affen als Kinderbetreuer? Warum gibt es (heute) keine fliegenden Pferde? Warum gibt es keine Wunderlampe? Warum kann man nicht zum Mond fliegen (1833) ? Warum schmeckt die Suppe, wenn sie die und die Rezeptoren, verbunden mit dem und dem Stoffwechsel, aktiviert?

Warum handelt xy in der und der Situation wie er gehandelt hat?

Warum gibt es den Zusammenhang xy? Weil es yz gibt etc.

Warum gibt es keinen Zusammenhang xa in xyz?

H.

Man unterscheidet private und akademische Forschung bzw. Praxis. Welche Aggregationen sind mit welchen Mitteln denkbar, welche Diffusionen (Praktika, Consulting bzw. Seminartagungen, hoch- und minderwertige Qualifikationen) sind wie auszubauen? Wie können Hochbegabte für die Soziologie gewonnen werden? Welche Inhalte und Gesprächspartner sind deswegen zu finden? Welche soziologische Hardware soll ebensolche

Software ersetzen, oder auf höheren Level anheben? Freies Angebot und Zugang von großen (vorgefertigten) Themen erleichtern womöglich (hochwertige) Einzelforschung. Wie ist eine Totalbetreuung von Soziologie realisierbar? Durch keinen Überwacher oder seinen Simulanten? Welche Führung in der Soziologie, nicht in den Literaturangaben, macht Sinn? Wie kann eine Statusmeldung spontan den ganzen Apparat aktualisieren? Welche Fehler sind denkbar, wenn ich den und den Aufsatz nicht gelesen habe? Über die logischen Schaltungen keine. Wächst Soziologie durch den beschleunigten Ausschuss von Nichtsoziologie?

Auf die praktische Soziologie und ihre Themenfindung bezogen kann man etwa die Kategorien aufstellen:

- Die xy-Gesellschaft als Populärsoziologie
- Soziologie des Widerspruchs, des Optimums, des Beweises gemessen etwa mit Philosophie des Widerspruchs, Optimums etc.
- Soziologie der Restauration, des Wahnsinns, der Zahl, des Anfangs, des rechten Terrors etc.
- Arbeitssoziologie, Musiksoziologie, Mathesoziologie, Biosoziologie, Farbensoziologie etc.
- Qualitativ, quantitativ
- Theorien(-aneignungen), Empirien
- Überlebens-, Panel-, Diskriminanzanalysen etc.
- Ein multivariates Datum innerhalb eines Clusters mit einem solchen in einem fremden Cluster verbinden, ersetzen, simulieren etc.
- Signifikante Zusammenhänge in Stichproben suchen oder bestätigen, und sie clustern
- Eigenschaften von Schätzern prüfen, simulieren
- Alle denkbaren Messungen bzw. Testungen anhand von Variablenskalen führen; die Sichtung, bedingt, dass alles gerechnet wurde, muss Algorithmen einführen, um Komplexitätsfragen zu meistern. Wann wird eine Sichtung unnötig?
- Die Forschungsfrage ist so zu formulieren, dass sie sehr unwahrscheinlich (in einer Datenbank) auftaucht oder nur höchste geistige Voraussetzungen sie behandeln können

- Wie kann man (komplexe) Fragen beantworten, ohne sie zu reduzieren? Eine Diskussion, Zeitschriften, Artikelreihen einführen, die nicht Komplexität, sondern die Regel, die Permutation fordern (Konsens, Konsistenz, verteilte oder verbundene Systeme etc.)

Um Forschungsressourcen zu steigern, sind zusätzliche finanzielle Mittel, neue Stellen, Kostensenkungen für bestimmte Phasen des Forschungsprozesses oder gar deren Outsourcing etc. angebracht.
Das Forschungspraktikum bei MDS zur allgemeinen Ressourcensteigerung in der Soziologie wurde zu dem Zweck (nach einfacher gedanklicher Überlegung) ins Leben gerufen (Mumdzhiev, 2011). Die Fortsetzung der Idee auf Diplomanden, Doktoranden etc. als Ausführungseinheiten bzw. Entscheider von Theorienlisten, Normeninterpreten und Empiriker in einem soziologischen Pipelining inklusive der „Aufrüstung dieser Einheiten“ ist augenscheinlich.

Speziell soll dieses Forschungspraktikum jedem bewerbenden Studierenden die Möglichkeit bieten, in theoretischer, normativer und empirischer Forschung Kenntnisse und praktische Erfahrungen zu sammeln.
Theoretische Forschung hat nach Patzelt (1986) zwei Aufgaben:

1. Die Prüfung und Verbesserung der logischen Konsistenz von Aussagengefügen;
2. Die Optimierung der Leistung von Theorien, verfügbares Wissen überschaubar und nützliche Perspektiven beziehbar zu machen (Patzelt, 1986: S.319).

Normative Forschung hat die Aufgaben:

1. Wertmaßstäbe und von ihnen getragene Gefüge normativer Aussagen, also brauchbare normative Theorien zu erarbeiten;
2. Vorgelegte normative Theorien zu überprüfen;
3. Inhalte, Strukturen und Prozesse der außen bestehenden Wirklichkeit mittels brauchbarer normativer Theorien zu bewerten;
4. Auf der Grundlage solcher Werturteile konkrete Handlungsanweisungen zu formulieren (Patzelt, 1986: S.321).

Empirische Forschung im weiteren Sinne hat die Aufgabe, mit den Sachverhalten der außen bestehenden Wirklichkeit übereinstimmendes Tatsachen-, Zusammenhangs- und

Erklärungswissen zu erarbeiten. Neben Finanz- und Personalplanung, Einsatz der Mitarbeiter, Beschaffung der benötigten Materialien und Geräte etc. sollen sieben Phasen einer empirischen Studie unterschieden werden, wobei fettgedruckt die Einsatzbereiche der Telepraktikanten markiert sind:

1. **Vorstudienphase (Sichtung der Literatur, Bibliografieren, Einlesen und Plausibilitätsprüfungen)**
2. Explorationsphase
3. Konzeptualisierungsphase
4. Pretestphase
5. Datenerhebungsphase
6. **Datenanalysephase (mit SPSS, ALLBUS Quer- und Längsschnittdaten; archivierte Daten, Codes bzw. Daten der Privatwirtschaft; abarbeiten der Kapitel aus Statistiklehrbüchern)**
7. Dateninterpretationsphase (vgl. dazu Patzelt, 1986: S.325 ff.)

Einige allgemeine Fragestellungen bzw. Behauptungen für den nachfolgenden technischen Teil dieses Arbeitspapieres werden zusammengefasst:

Wie kann man den Stand der soziologischen Wissenschaft messen? Den Konsistenzgrad der Theorienaneignungen? Und davon direkt, verlustfrei, profitieren? Über Hard- und Softwareanalysen?

Wie kann es sein, dass es keinen Diskurs gibt? Dass grundlegende Überlegungen (etwa zu Werten, Systemen) nicht im geringsten verfolgt werden? Es ist die Eigenart verteilter Systeme.

Kein Gespenst geht um, die Minderwertigkeit der aktuellen Soziologie.

Es mag leichter sein, anwesende Pappkameraden abzuschießen, und durch die kumulative Übernahme der einschlägigen Ressourcen (Praktikanten-, Diplomanden-, Doktoranden-, Institutstätigkeiten sowie strukturelle Verbesserungen in der Produktionslogik der Sozialforschung) eine Simulation von Max Weber im Sinne einäugiger Könige zu präsentieren. Die Vorgabe „großer Themen“ und jederzeitige Überlegenheit (sofern realisiert) der besser organisierten Denkungsart eines beliebigen standardnormalen Soziologen (und Helfer, im Angesichte minderwertiger Opponenten) anhand einfachst anzuwendender

Komplexitätseigenschaften (Algorithmen), die Ausdehnung der Sprache auf Interdisziplinaritäten; die Entgegnung der Sätze von Adorno, Luhmann etc. anhand der selben Sätze, die eine Art Blitzsoziologie aus vorgegebenen „intelligenten Thesen“ implementieren u.ä. wird als soziologische (und personelle) Aufklärung zu betreiben sein.
Der ambivalente Zustand, dass die vorliegende desorganisierte Soziologie als verteilte Systeme oder deren Figuration Facebook, Fragen der Migration, Integration, Sozialstruktur, Handlungstheorie (wegen Ressourcenmangel) nicht ansatzweise auszudenken vermag, kann kurzfristig über Effizienzsteigerungen einer zentralisierten Produktion im Sinne der Entwicklung von „besseren“ Computerarchitekturen gemieden werden; die sinnvoll erscheinende Gleichschaltung der Soz-prozessoren von einem Datenfluss- oder Beamtendiskursmodell zu einem integrierten Pipelining der Richtungen ergibt sich aus der Lektüre einfachster Informatiktaschenbücher.
Die Simulation von Komplexität kann durch die Realisation von Server- / Clientmodellen nicht weniger sinnlos sein, als die endlose Absprache von Regeln oder Problemen, welche kaum voller Bedeutung den Wissenswandel von Mythos über Philosophie und Positivismus, oder im Kontext, von Gesellschaft zu Soziologie und wiederum zu Gesellschaft und Schweigen, oder diskursfreier „informierter“ Herrschaft und dem Sein zu überführen vermag. Der integrierende Schritt zurück ist derjenige, der im Lichte der Ausdifferenzierung nicht machbar erscheint, und welchen keine elende Geschichte als versäumte Entwicklung oder gar Ende zu kennen glaubt.
Wie sind allgemeine Restaurationen möglich? Wie kippt eine Hochschule, wenn das Intranet abgestellt wird? Wie ist Aufbau ohne Crash möglich? Wie ist absolute Interdisziplinarität möglich, und für Leute wie mich, Milko Mumdzhiev, imitierbar, approximierbar zu machen? Ist sie lehrbar, und warum?
Ist Himmler überwunden, der behauptet, anständig zu sein, heißt seinen Job weitermachen, während irgendwelche Menschen sterben?
Ich habe mich für Soziologie entschieden, und nicht für das Retten von zumindest einem Menschenleben. Das war rational, und sicher bewußt gegen Afrika. Also bin ich Rassist, obwohl ich keinen Fragebogen in diesem Sinne beantworten würde.
Solche linken Floskeln werden, aus der Nähe ihrer Komplexität betrachtet, sehr hart zu begegnen sein. Dass alle „echten“ Probleme wie repetitive Literatur im

Begründungszusammenhang florieren, ist keine Überraschung. Und weil der Problemlösungsmechanismus im Einfachen gezeichnet ist, im Komplexen jedoch ohne Algorithmen auskommen muss, scheint eine Besinnung auf die Vertreter der Thesen, als Turnier, als Vergleich von Hard- oder Software, und nicht mehr, realisierbar zu sein. In diesem Sinne ist ein individualistisch orientierter Geheimdienst effizienter als Gesis-Soziologie, oder anders gesagt, das individualistisch gesehene google ist der Tod vom Diskurs, der nie menschlich als Wert war und sein kann, jedoch als Zweck geleitet, gedrängt, überlastet, abgestellt, hochgefahren wird etc.

Die Reihen, die beliebiges Wachstum messen, scheinen zu konvergieren, wie jene, die entsprechende Schrumpfung vorweisen. In diesem Sinne sind viele kleine Zusammenhänge, so wie Zwecke, per Definition, wertlos. Die gescheiteren Luhmann, Weber etc., für den einfachen Soziologen imitierbar zu machen, und so effizient wie möglich, ist bei Einsicht der eigenen Unterlegenheit, ein Aufklärungsprogramm, das entweder der kalten Schulter von Titanen bedarf, oder des Hypothesenfingierens.

3. Die technische Sichtweise

3.1 Grundlagen

3.1.1 Klassifikation von Algorithmen

Ein Algorithmus ist eine Berechnungsvorschrift, welche aus einer bestimmten Menge von Eingaben, Ausgaben und einer endlichen Anzahl von eindeutigen Berechnungsschritten besteht, und dabei in einer endlichen Anzahl von Schritten terminiert (in der Regel).

Probleme, die durch Algorithmen gelöst werden können, werden auch entscheidbar genannt.

Algorithmen können nach Qualität der Lösung und Berechnungsaufwand unterteilt werden. Um diese Merkmale näher zu quantifizieren, definiert man wie folgt:

Ein Algorithmus heißt dann exakt, wenn er für alle Instanzen eines Problems eine exakte Lösung findet. Für alle entscheidbaren Probleme lassen sich auch exakte Algorithmen finden. Dabei ist der Berechnungsaufwand von exakten Algorithmen oft zu hoch, um in vertretbarer Zeit auf einem Computer gelöst zu werden. Deswegen verwendet man sog. Approximationsalgorithmen, welche nicht das Finden einer exakten Lösung garantieren, sondern in den meisten Fällen exakte Lösungen gut approximieren, also annähern. Approximationsalgorithmen heißen auch Heuristiken, da sie solche Strategien verwenden, die auf Vermutungen, plausiblen Annahmen, Erfahrungen etc. beruhen. Hier wird ein Ansatzpunkt der MW-Simulation gesehen.

Der Berechnungsaufwand (Komplexität) eines Algorithmus wird in Zeit- und Speicherbedarf gemessen:

1. im schlimmsten Fall (engl. worst case complexity) und
2. im Mittel (engl. average case complexity).

Für die Beurteilung des zeitlichen Berechnungsaufwandes eines Algorithmus, unabhängig vom Rechnertyp, definiert man die Anzahl der elementaren Operationen des Algorithmus als Maß des Rechenaufwands. Diese Zahl hängt aber von der Problemgröße (die als Eingabeparameter des Algorithmus verstanden werden kann) ab; daher stellt man als Maß der Zeitkomplexität das Wachstum an elementaren Operationen als Funktion der Problemgröße in Form der sog. O-Notation dar. Wenn die Problemgröße mit n bezeichnet wird (z. B. sortiere eine Liste von n ganzen Zahlen in aufsteigender Reihenfolge), dann besitzt

die Zeitkomplexität die Ordnung von f (n), wenn es eine Konstante c gibt, so dass c*f (n) eine obere Schranke der Anzahl elementarer Operationen zur Lösung des Problems darstellt. Die Zeitkomplexität des Algorithmus wird dann mit O (f (n)) notiert. Oft sagt man, dass Algorithmen, bei denen f (n) ein Polynom in n ist (= polynomielle Algorithmen, z. B. $O(n^3+ ½*n)$), effizient sind. Im Gegensatz dazu sind exponentielle Algorithmen (z. B. $O(2^n)$ oder $O(n^{n/2})$) ineffizient. Bei der Bewertung der Konstanten c muss man aufpassen, da diese, wenn n klein ist, die Rechenzeit dominieren können.

So kann man die Effizienz eines exakten Algorithmus bewerten, wenn man seine Komplexität mit der der inhärenten Komplexität vergleicht. Diese stellt die untere Schranke für die Anzahl benötigter Operationen dar. Die inhärente Komplexität des Problems, unter n ganzen Zahlen die maximale Zahl zu bestimmen, ist $\Omega(n)$, da auf jeden Fall $n-1$ Vergleiche dazu notwendig sind. Ein Algorithmus heißt dann optimal, wenn seine Komplexität mit der inhärenten Komplexität des Problems identisch ist. Also ist ein Suchalgorithmus für die größte unter n Zahlen mit der Komplexität O(n) optimal. Algorithmenoptimalität muss jedoch von der Optimalität einer Lösung unterschieden werden.

Bestimmte Probleme lassen sich mit polynomiellen Algorithmen lösen. Man bezeichnet diese Klasse von Problemen im allgemeinen mit dem Symbol P.

Andere Probleme lassen sich mit polynomiellen Algorithmen auf sogenannten nichtdeterministischen Maschinen lösen. Es handelt sich dabei um hypothetische Computer, die die Möglichkeit besitzen, Lösungen zu erraten und diese dann in polynomieller Zeit zu verifizieren. Diese Problemklasse heißt NP. Offensichtlich gilt $P \subseteq NP$. Ob jedoch P = NP gilt, ist immer noch ein ungelöstes Problem der theoretischen Informatik.

Wenn es irgendein Problem unter solchen einer bestimmten Problemklasse gibt, das in polynomieller Zeit gelöst werden kann, dann sind auch alle Probleme dieser Klasse in polynomieller Zeit lösbar. Die Klasse dieser Probleme wird als NP-schwer bezeichnet, und deren Teilklasse, die in NP enthalten ist, heißt NP-vollständig (so SAT, 3-SAT für nicht mehr als 3 Literale je Klausel, siehe weiter unten).

Oft gibt es für Probleme in NP Algorithmen mit exponentieller (oder höherer) Komplexität. Für manche Problemgrößen mögen diese Algorithmen in ihrer Laufzeit tolerierbar sein, für andere Problemgrößen ist man hingegen auf Heuristiken mit polynomieller Laufzeit angewiesen, um in begrenzter Zeit eine Lösung zu erhalten (Haubelt & Teich, 2010: S.541 f.).

Zwei grundlegenden Definitionen werden nochmal hervorgehoben:

Zeitkomplexität

Unter der Zeitkomplexität eines Problems versteht man die Anzahl der Rechenschritte, die ein optimaler Algorithmus zur Lösung dieses Problems benötigt, in Abhängigkeit von der Länge der Eingabe. Man spricht hier auch von der asymptotischen Laufzeit und meint damit, in Anlehnung an eine Asymptote, das Zeitverhalten des Algorithmus für eine potenziell unendlich große Eingabemenge. Es interessiert also nicht der Zeitaufwand eines konkreten Programms auf einem bestimmten Computer, sondern viel mehr, wie der Zeitbedarf wächst, wenn mehr Daten zu verarbeiten sind, also z. B. ob sich der Aufwand für die doppelte Datenmenge verdoppelt oder quadriert. Die Laufzeit wird daher in Abhängigkeit von der Länge n der Eingabe angegeben, und für immer größer werdenden solchen asymptotisch unter Verwendung der Landau-Notation abgeschätzt.

Platzkomplexität

Unter der Platzkomplexität eines Problems versteht man den (minimalen) Bedarf an Speicherplatz eines Algorithmus zur Lösung dieses Problems, in Abhängigkeit von der Länge der Eingabe. Es interessiert also nicht der Speicherbedarf eines konkreten Programms auf einem bestimmten Computer, sondern vielmehr, wie der Speicheraufwand wächst, wenn mehr Daten zu verarbeiten sind. Also z. B. ob sich der Aufwand für die doppelte Datenmenge verdoppelt oder quadriert.

Probleme gemeinsamer Komplexität fasst man in einer Komplexitätsklasse (einer Komplexitätskategorie) zusammen. Man bedient sich dafür sogenannter Komplexitätsmaße. Diese zeigen den Aufwand an, mit dem Turing Maschinen Funktionen berechnen oder Sprachen erkennen. Aufwand bedeutet bspw. Laufzeit oder Platzbedarf. Oft ist eine exakte Kostenangabe nicht möglich. Deshalb verwendet man Komplexitätsschranken, die uns die Größenordnung der Kosten anzeigen (Althaus, 2009: S. 34 f.).

Es existieren diverse (sich z.T. selbsterklärende) Algorithmendesigns und Konzepte, die nachfolgend (ohne weiteres Erläutern oder Übersetzen zwecks Einordnung) gelistet werden:

Divide and Conquer, Dynamic programming, Backtracking, Local search, Greedy algorithms (Hromkovic, 2004: S.134); Pseudo polynomial time algorithms, Parametrized complexity, Branch and bound, Lowering the worst case exponential complexity, Local search, Relaxation

to linear programming (Hromkovic, 2004: S.150). Eine Verwendung von diversen Alternativen im Kontext von Komplexität wird sich nicht vermeiden lassen.

3.1.2 Logische Programmierung

Speziell soll im weiteren die sogenannte logische Programmierung berichtet werden.

Es handelt sich dabei um einen Berechnungsformalismus, der die folgenden Prinzipien vereinigt:

- Er benutzt Logik als Sprache zur Formulierung von Sachverhalten.
- Er benutzt Inferenz-Prozeduren, um dieses Wissen zu befragen, zu benutzen und zu vermehren.

In der logischen Programmierung benutzt man deklarativ repräsentiertes Wissen in Form einer Wissensbasis, unabhängig vom auszuführenden Programm. Eine Frage wird in Form einer Behauptung gestellt, und dann die Wissensbasis durchsucht, bis ein Objekt gefunden ist, das die Aussage erfüllt. Dabei werden Formeln der Prädikatenlogik als Anweisungen der Programmiersprache benutzt.

Ein Logik-Programm ist also eine Menge von Axiomen oder Regeln, die Relationen zwischen Objekten definiert. Die Berechnung des logischen Programms ist eine deduktive Folgerung des Programms. Ein Programm definiert eine Menge von Folgerungen, welche gleichzeitig sein Inhalt ist. Die Kunst der logischen Programmierung ist es, kurze und elegante Programme zu konstruieren, die die geforderte Bedeutung haben.

Den zentralen Bestandteil bildet eine applikationsunabhängige Inferenz-Maschine. Diese virtuelle Maschine ist eine Herleitungsprozedur, die nach einer vorgegebenen Suchstrategie die Wissensbasis durchsucht, um eine Schlussfolgerung auf eine Anfrage mit Hilfe der Deduktion zu ziehen, und die vorgibt, nach welcher Methode Folgerungen gezogen werden sollen, damit die gestellte Frage als richtig oder falsch beantwortet werden kann.

Inferenzregeln sind die Resolution oder der Modus Ponens, als Suchstrategien werden Standard-Suchverfahren in einem geeigneten Graphen genutzt.

Auf den Benutzer kommen zwei Aufgaben zu. Als erstes stellt er dem System Anfragen, um den Wahrheitswert gewisser Aussagen oder Relationen zu finden. Allerdings muss ein Programmierer vorher Wissen zu dem Problem in der Wissensbasis abgelegt haben (Fuhr, 2003: S.168).

Praktisch soll darin eine weitere Möglichkeit zur Beschleunigung der Sozialforschung gesehen werden. Die Generierung einer nach strengen Regeln eingespeisten und kodierten Thesendatenbank für das “logische Programm” (es ist üblich, dass in Forschungsarbeiten Hypothesenbatterien samt Prüfung als Strings gelistet werden; diese zu extrahieren, und kombinieren, ist mit angeschlossener Rechnung oder Syntax etwa an einem externen Datensatz, keine komplexe Aufgabenstellung) ist ein Vorschlag dazu. Sinnvoll wäre neben einer ausdifferenzierten Theoriendatenbank auch eine Empiriedatenbank, ein Datensatz, der in SPSS-Form die gesamte empirische Soziologie (und deren Ergebnisse) enthält, und auszuwerten erlaubt.

Bestimmte Recherchen samt den gedanklichen Vorüberlegungen (in der Explorations- und Konzeptualisierungsphase, während Plausibilitätsprüfungen etc.), die vor oder während einer Studie Pflicht sind, sollen nicht auf Volltexte, sondern auf argumentative Schlagworte reduziert und beschleunigt werden können. Im besten Falle ist die jeweilige Extraktion der Abstacts, Hypothesen, oder der Konklusionen und des empfohlenen neuen Forschungsbedarfs automatisiert durchführbar (z.T. sind herkömmliche Bibliografiestrukturen nur auszuweiten); etwaige notwendige Kategorisierungen oder Komprimierungen (von Büchern, Theorienaneignungen etc.) würden hintangesetzt werden müssen, also von vornherein auf dieser Ebene wohl mehr Zeit beanspruchen, als im günstigsten Fall beisteuern können.

Hochwertige Soziologen im theoretischen Arbeitsbereich sollten als nicht komprimierbar, jedoch imitierbar vorgestellt werden (im Sinne der MW-Simulation). Diese selbst können genannte Datenbank z.T. erübrigen, vorwegnehmen, geringschätzen, sie auf höherer Architekturebene zusammenfassen, als “banale” Feldarbeit betrachten etc.

Konkret werden im nächsten Abschnitt dazu die sogenannten SAT-Löser ausgeführt.

3.1.3 SAT-Solver

Das Boolesche Erfüllbarkeitsproblem, oder kurz SAT-Problem (engl. boolean satisfiability), ist der Prototyp aller sogenannten NP-vollständigen Probleme. Es besitzt vielfältige Anwendungen im Bereich der Synthese und Verifikation von Computersystemen; innerhalb der Soziologie davon verstärkt zu profitieren, wird hier als beachtenswert angesehen.

Das Boolesche Erfüllbarkeitsproblem definiert man wie folgt:

Definition (Boolesches Erfüllbarkeitsproblem): Gegeben sei eine Boolesche Funktion f. Entscheide, ob f erfüllbar ist.

Anders ausgedrückt: man soll zeigen, dass mindestens eine Variablenbelegung β existiert, sodass f (β) = T. Um das Problem zu lösen, werden sog. SAT Solver herangezogen.

Die meisten SAT-Solver basieren auf der Repräsentation der Booleschen Funktion als aussagenlogische Formel φ, dabei ist diese in konjunktiver Normalform (KNF) gegeben. In KNF besteht φ. aus der Konjunktion von sog. Klauseln. Jede Klausel besteht aus der Disjunktion von Literalen, wobei ein Literal eine Variable bzw. deren Negation ist. Um eine aussagenlogische Formel in KNF erfüllen zu können, muss jede Klausel und somit mindestens ein Literal in jeder Klausel den Wert T annehmen.

Ein Literal heißt dann unspezifiziert, wenn ihm kein Wert zugewiesen wurde. Wenn ihm der Wert T zugewiesen wurde, so heißt es erfüllt, im anderen Falle (Wert F) heißt es verletzt. Diese Bezeichnungen werden auf Klauseln übertragen. Eine Klausel heißt unspezifiziert, solange sie keinen eindeutigen Wert angenommen hat. Sie ist erfüllt, wenn sie den Wert T annimmt, und heißt verletzt, wenn sie den Wert F annimmt. Klauseln, bei denen lediglich ein Literal unspezifiziert ist, werden als Einerklauseln (engl. unit clause) bezeichnet. Diese implizieren direkt die Belegung des Literals.

Beispiel:

Gegeben ist die Boolesche Funktion $f := (\neg x1 \vee x2 \vee x3) \wedge (\neg x2 \vee x4) \wedge (\neg x3 \vee x5) \wedge (\neg x5 \vee x4)$. Es gilt die Belegung x1 := T, x2 := T, x5 := T und x3 := F. Die Variable x4 ist noch nicht belegt. Somit sind die Literale $\neg x1$, $\neg x2$, x3 und $\neg x5$ verletzt. Die Literale x2, $\neg x3$ und x5 sind erfüllt, während das Literal x4 unspezifiziert ist. Es gilt:

$f = (\neg x1 \vee x2 \vee x3)$	$\wedge$	$(\neg x2 \vee x4)$	$\wedge$	$(\neg x3 \vee x5)$	$\wedge$	$(\neg x5 \vee x4)$.
erfüllt		unspezifiziert		erfüllt		unspezifiziert

Die zweite und vierte Klausel sind Einerklauseln, da lediglich ein Literal, nämlich x4 unspezifiziert ist. Die Funktion f kann durch die Belegung x4:= T erfüllt werden.

Die meisten SAT-Solver basieren auf dem DPLL-Algorithmus, der nach seinen Erfindern Davis, Putnam, Longman und Loveland benannt ist.

Der grundlegende Algorithmus SAT SOLVE ist im Folgenden beschrieben.

```
SAT SOLVE(Φ)
{
IF (DEDUKTION(Φ,β) = F)
        RETURN F;
WHILE (VERZWEIGUNG(Φ,β) != F)
        WHILE (DEDUKTION(Φ,β) = F)
                blevel := DIAGNOSE(Φ,β);
                IF (blevel = 0)
                        RETURN F;
                ELSE
                        BACKTRACK(blevel);
RETURN T;
}
```

Der Algorithmus erhält als Eingabe eine bestimmte Klauselmenge Φ und besteht im großen und ganzen aus drei Funktionen: DEDUKTION, VERZWEIGUNG und DIAGNOSE.

Zu Beginn sind alle Variablen nicht belegt, und somit alle Literale unspezifiziert. Im ersten Schritt findet eine Vorverarbeitung durch die Funktion DEDUKTION statt. Diese reduziert Φ durch sukzessives Auffinden von Einerklauseln und Belegung der zugehörigen unspezifizierten Variablen, so dass die jeweilige Einerklausel erfüllt ist, und von der Menge der Klauseln entfernt werden kann.

Durch diese Variablenbelegung zur Erfüllung einer bestimmten Einerklausel kann es zu einem Konflikt kommen, indem eine andere Klausel verletzt wird. In diesem Fall wäre die Boolesche Funktion nicht erfüllbar und DEDUKTION liefert F zurück.

Andererseits können durch die entsprechende Variablenbelegung neue Einerklauseln entstehen. Die sukzessive Reduktion durch Behandlung der Einerklauseln wird im Englischen unit propagation oder Boolean Constraint Propagation (BCP) bezeichnet. Neben BCP werden in der Funktion DEDUKTION oftmals auch sog. reine Literale (pure literals) eliminiert. Dabei wird eine Variable, die lediglich als positives (negatives) Literal in den Klauseln auftritt, mit T (F) belegt. Die Variablenbelegungen und deren Reihenfolge werden in der Belegung β gespeichert, um später eventuelle Fehlentscheidungen revidieren zu können.

Der Hauptteil des SAT SOLVE-Algorithmus besteht aus einer WHILE-Schleife, die mit der Funktion VERZWEIGUNG aufgerufen wird. Eine freie Variable wird gewählt und mit einem Wert, dem sog. Verzweigungsliteral, belegt. Die Auswahl dieses Verzweigungsliterals ist entscheidend für die Geschwindigkeit des SAT-Solvers, da manche Entscheidungen schneller eine Lösung finden können als andere. Hierin soll die Findung von neuen Paradigmen, Theoriengründungen, Geniestreichen etc. gesehen, und für die MW-Simulation ausgenutzt werden.

Solange die Funktion VERZWEIGUNG freie Variablen zum Verzweigen findet, liefert diese T zurück. Gibt es allerdings keine freien Variablen mehr, ist eine Belegung der Variablen gefunden, welche die Boolesche Funktion erfüllt.

Nach der Verzweigung wird die Funktion DEDUKTION aufgerufen, dabei werden alle Einerklauseln durch BCP eliminiert, bis keine Einerklauseln mehr gefunden werden, oder ein Konflikt durch eine verletzte Klausel entsteht. Im konfliktfreien Fall wird eine neue Verzweigung gemacht, während im Konfliktfall die Funktion DIAGNOSE den Grund für den vorgefundenen Konflikt analysiert, das Ergebnis lernt und schließlich eine Zurückverfolgung (backtracking) durchgeführt wird. Die Zurückverfolgung in der Funktion BACKTRACK kann zuvor getroffene Entscheidungen, die in β gespeichert sind, bis zu einem bestimmten Punkt (blevel) rückgängig machen. Wird jedoch erkannt, dass bis zur Entscheidung blevel = 0 zurückgegangen werden muss, so gilt die Funktion als nicht erfüllbar.

Fortschrittliche SAT-Solver führen dabei ein sog. Rücksprungverfahren (backjumping) oder eine nichtchronologische Zurückverfolgung durch.

Die Effizienz von solchen SAT-Solvern misst man in der Anzahl der benötigten Zurückverfolgungsschritte, bis eine konsistente Belegung gefunden wurde oder die Formel als unerfüllbar konstatiert wurde. Wenn eine schlechte Verzweigungsstrategie gewählt wurde, müssen häufiger Zurückverfolgungsschritte durchgeführt werden (Haubelt & Teich, 2010: S.542 ff.).

Es stellt sich allgemein die Frage, wie ex ante Verifikation von Soziologien (wenn etwa eine empirische Phase noch bevorsteht) genau zu nehmen ist. Eine Variation / Permutation des Gegebenen scheint realistischer, eine von vornherein breiter als Alternativen angelegte Strategie sicherer zum Erfolg zu führen (z.B.: simuliere viele kleine Soziologen als Initiatoren,

und weise viele Helfer in die Ausführung ein: viele Lehrstühle, Institute, Arbeitskreise, Datensätze etc.).

3.1.4 SMT-Solver

SAT-Solver dienen u. a. zur Verifikation von Hardware und Software auf höheren Abstraktionsebenen. Dabei werden Eigenschaften der betrachteten Systeme in Aussagenlogik übersetzt, um diese (u.a.) mittels SAT-Solvern beweisen zu können. Dies kann zu sehr großen aussagenlogischen Formeln führen, sofern überhaupt eine ganzheitliche Übersetzung möglich ist, da nicht alle Eigenschaften in Aussagenlogik dargestellt werden können. Obwohl sich die verfügbaren SAT-Solver enorm verbessert haben, macht es Sinn, die Eigenschaften der Systeme in Logiken zu beschreiben, die expressiver als Aussagenlogik sind.

Eine Möglichkeit hierzu ist die Prädikatenlogik erster Ordnung mit Äquivalenz. Um die Erfüllbarkeit solcher prädikatenlogischer Formeln zu zeigen, können Theorembeweiser eingesetzt werden. Leider ist der Einsatz allgemeiner Theorembeweiser für viele praktische Probleme oft inadäquat.

Bei diversen Problemen wird versucht, die Frage zu beantworten, ob eine prädikatenlogische Formel φ. bezüglich einer entscheidbaren Hintergrundtheorie T erfüllbar ist. Aus diesem Grund spricht man auch von dem sogenannten Erfüllbarkeitsproblem modulo Theorien (engl. Satisfiability Modulo Theories, SMT). Dabei wäre die Kombination mehrerer Hintergrundtheorien denkbar. Beispiele für Hintergrundtheorien sind im mathematischen Kontext: Gleichheit und uninterpretierte Funktionen, Lineare Arithmetik (z.B.+,-) über reelle und ganze Zahlen, Divergenzlogik, Bitvektor-Theorie und Array-Theorie.

Programme zum Lösen des SMT-Problems werden als SMT-Solver bezeichnet.

Die meisten heute verwendeten SMT-Solver sind sog. indirekte SMT-Solver. Indirekte SMT-Solver kombinieren DPLL-SAT-Solver mit bestehenden Theorienlösern. Der SAT-Solver steuert den kombinierten Lösungsansatz, indem er atomare Formeln (aussagenlogische Variablen und Formeln in der Hintergrundtheorie) mit Booleschen Werten belegt und den Löser für die Hintergrundtheorie regelmäßig beauftragt, die Erfüllbarkeit der Formeln der Hintergrundtheorie zu überprüfen. Im Gegensatz dazu übersetzen direkte SMT-Solver die

Eigenschaften in aussagenlogische Formeln und verknüpfen die Hintergrundtheorie mit aussagenlogischen Formeln.

In der folgenden Abbildung wird die Arbeitsweise eines indirekten SMT-Solver für prädikatenlogische Formeln gezeigt (Haubelt & Teich, 2010: S.553):

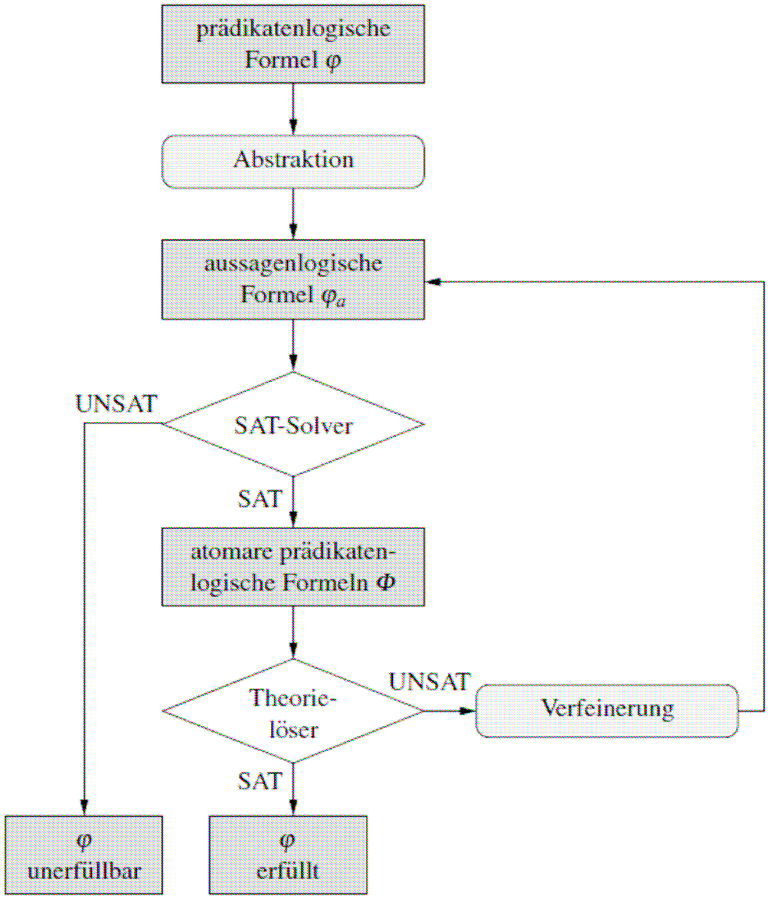

Abb. C.4. Indirekter SMT-Solver

Ausgehend von der prädikatenlogischen Formel φ, wird als erstes eine aussagenlogische Abstraktion durchgeführt. Dabei werden die atomaren prädikatenlogischen Formeln durch Boolesche Variablen repräsentiert. Die resultierende aussagenlogische Formel φ_a ist die Eingabe für den verwendeten SAT-Solver. Ergibt der SAT-Solver, dass φ_a nicht erfüllbar ist, so wird auch die prädikatenlogische Formel φ als unerfüllbar angesehen. Findet der SAT-Solver eine konsistente Belegung für die aussagenlogische Formel φ_a, werden die atomaren prädikatenlogischen Formeln Φ, die durch die Belegung impliziert werden, an den Theorielöser weitergereicht. Der Theorielöser versucht seinerseits für die Konjunktion der

übergebenen Formeln eine konsistente Belegung entsprechend der verwendeten Hintergrundtheorie T zu finden. Wenn eine solche konsistente Belegung gefunden wird, bildet die Vereinigungsmenge dieser Belegung zusammen mit der Belegung der atomaren aussagenlogischen Formeln aus φ. die konsistente Belegung für φ. Ansonsten führt der Theorielöser eine Konfliktanalyse durch und verfeinert die abstrahierte aussagenlogische Formel entsprechend (Haubelt & Teich, 2010: S.551 ff.).

3.1.5 Simulation von Max Weber

Im Kontext der Soziologie soll eine Serverfigur wie Max Weber, interpretiert als eine im Vergleich zu anderen immer höherwertigere Maschine, durch mehrere einfachere Maschinen oder Helfer (Theorielöser, Auswerter, Normenprüfer etc.) simuliert werden, und entsprechende Nutzen- / Kostenbilanzen als Leistungsmesser ablegen können. Eine Analogie dazu wird aus der theoretischen Informatik geliefert.

k-Band-Turingmaschinen Tk besitzen k separate Bänder mit einem Lese- / Schreibkopf für jedes Band. Die Köpfe können sich unabhängig voneinander bewegen. Jedes Band ist separat ansteuerbar. Ein- / Ausgabe steht immer auf dem 1. Band.
Folgerungen bzw. Thesen:
- mit einer Einband-TM T1 ist eine Mehrband-TM Tk simulierbar;
- mit einer Mehrband-TM Tk ist eine Einband-TM T1 simulierbar;
- mit einer Mehrband-TM Tk ist eine Registermaschine RM simulierbar;
- mit einer Registermaschine RM ist eine Turingmaschine simulierbar.

Um die Tk simulieren zu können, muss der Kopf der T1 die Arbeit von k separaten Köpfen realisieren. Würden die k Bänder der Tk mittels Klebeband aneinandergefügt werden, entstünde ein breiteres Band bestehend aus k Spuren (wir reduzieren Max Weber also, und nehmen Extrakosten in Kauf, um die entsprechende Simulation durchzuführen).
In diesem Modell achten wir darauf, dass sich die Zellen, auf die die k Köpfe der Tk zeigen, auf dem k-Spuren-Band der T1 (mit nur einem Kopf) übereinander liegen. Die Spalten des neuen Bandes werden aus Sicht der T1 als eine einzelne Zelle mit k Spuren interpretiert. Dieses sei unser neues Band für die T1, welche die Tk simuliert.

Trivialerweise können 1-Band-Turingmaschinen durch k-Band-Turingmaschinen simuliert werden, indem nur das erste der k Bänder benutzt wird. Dadurch wird die k-Band-Turingmaschine auf eine 1-Band-Turingmaschine reduziert (Max Weber ist also als Musiker in Sachen Religion leicht imitierbar).
Die Anzahl der benutzten Bandsymbole der Mehrspuren-TM wird am größten, wenn die Mehrband-TM in jedem Schritt einen Kopf nach rechts und einen nach links bewegt.
Benötigt die Mehrband-TM mehr Schritte, sind dies höchstens 2M+1 Symbole. Jeder Schritt der Mehrband-TM kann somit mit O(M) Schritten der Mehrspuren-TM simuliert werden.
Benötigt die Mehrband-TM M Schritte, benötigt die Mehrspuren-TM $O(M^2)$ Schritte (Kosten) (Althaus, 2009: S. 16 ff.).

Der soziologische Produktionsprozess kann Lehrstühle, Diplomanden, (Mafo-)Institute, Consulter, Verbände etc. als kleine Max Weber, als Akteure und eigenständige Architekturen identifizieren. Entsprechend dem Verlauf der empirischen Forschung (Patzelt, 1986: S.325 ff.) können Pipelines, Deadlines, logische Einheiten, CPUs, aktive Ressourcen, Leistungsmaxima, Protokolle, Middleware etc. identifiziert und soziologischen Realitäten zugeordnet werden. Dieses wissend, müssen Variationen von Architekturen zum Zwecke von Effizienzsteigerungen geprüft werden. Inwiefern sich Pipelining weiterentwickelt, oder Systemverteilung nachteilig auswirkt, soll durch Angabe des begrifflichen Hintergrundes im folgenden gezeigt werden.

3.2 Konzepte für Computerarchitekturen

Eine Programmiersprache kann als Spezifikationswerkzeug verstanden werden, das es ermöglicht, Berechnungsschritte zu formulieren, wobei ein bestimmtes Berechnungsmodell angenommen wird.
Eine Computerarchitektur kann als Werkzeug zur Implementation des Berechnungsmodells angesehen werden, oder zur Ausführung eines gegebenen Berechnungsschritts, der durch die Mittel der Programmiersprache ausgedrückt wird, wobei ein bestimmtes Berechnungsmodell, z.B. das Von-Neumann-Modell, vorgegeben wird (Glende, 2003: S.4). Analog wollen wir die Strukturierung der Sozial- und Marktforschung, deren Figuration etc. denken.

Computerarchitektur wird als diejenige Struktur eines Rechners bezeichnet, die ein Maschinensprachprogrammierer verstehen muss, um für eine Maschine ein korrektes (zeitunabhängiges) Programm zu schreiben.
Eine Computerarchitektur umfasst die Definitionen von Register, Speicher, Befehlssatz, Befehlsformat (z.B. Zuweisungsbefehle, Sprungbefehle), Adressierungsformen, Kodierung der Befehle, Implementierung und Realisierung der Befehle usw. (Lehrbücher der Soziologie, der Methoden, Kriterien der Wissenschaftlichkeit, Akteure / Organisation der Forschung, des Studiums etc. können analog gedacht werden).

Eine Multilevel-Beschreibung, hier mit vier Computerebenen, kann so aussehen:
4 „top level“-Beschreibung eines Computersystems, das auf der Spezifikation der Grundeinheiten basiert, wie z.B. Prozessor, Speicher, Bussystemen
3 Programmierebene
2 Logik Design Ebene
1 elektronische Schaltebene (Glende, 2003: S.15)

Die konkrete Architektur wird in ein logisches und physikalisches Design unterteilt (Glende, 2003: S.17):
– logisches Design:
Eine Abstraktion des physikalischen Designs, die oft vor dem physikalischen Design aufgestellt wird. Die Spezifikation sollte folgendes enthalten:
a) Vereinbarungen, wie sich die logischen Komponenten in der blackbox verhalten sollen (z.B. Register-Ausführungs-Einheiten);
b) Die Spezifikation ihres Zusammenwirkens (Protokolle);
c) Die Spezifikation der Sequenz der Informationsübertragung, die durch jede vereinbarte Operation eingeleitet wird (Mikrobefehle, Befehle).
Die Beschreibung kann in Blockdiagrammen oder ADL (Architektur-Beschreibungssprache) stattfinden.
– das physikalische Design, welches enthalten sollte:
a) Vereinbarungen über die benutzten Schaltelemente, was auch die Spezifikation der Signale beinhaltet;

b) Die Spezifikation ihres Zusammenwirkens;
c) Die Vereinbarung eingeleiteter Signalfolgen (Glende, 2003: S.17).

Die Struktur eines Computers mit sechs Ebenen inklusive des unterstützenden Programms in der rechten Spalte (Tanenbaum & Goodman, 2001: S.24):

Ebene 5	Problemorientierte Sprache	Übersetzer, Compiler
Ebene 4	Assemblersprache	Übersetzer, Assemblierer
Ebene 3	Betriebssystemmaschine	Teilinterpretation, OS
Ebene 2	Befehlssatzarchitektur, ISA	Interpretation, Mikroprogramm oder direkte Ausführung
Ebene 1	Mikroarchitektur	Hardware
Ebene 0	Digitale Logik	

Wissenschaftstheorie, Forschungsabläufe, Hand- und Lehrbücher werden im soziologischen Studium mitgeteilt. Bestimmte Kriterien, Kritiken, Verhaltensregeln und Mindeststandards, die in der Praxis die Akteure (Institute, wiss. Mitarbeiter, Hilfskräfte, Dateneinleser) in Konkurrenzsituationen strukturieren, sind offensichtlich. Auf jeder Ebene der Forschungsarchitektur, Ebenenbeziehungen inbegriffen, sollen analog der Informatik, Steigerungen einzuführen sein. Die Kostenfrage ist dabei immer zu stellen.

Dazu sollen aus dem Abschnitt spezielle Technologien von Glatz (2010: S.639 ff.) weitere Ausführungen zitiert werden. Daneben wird häufig das Vorlesungsskript Rechnerstrukturen von Spaniol et al. (2011) sowie das von Glende (2003) eingebunden.

3.3 Einprozessorsysteme und Speicher

Die Von Neumann Architektur

- Leitwerk (Control Unit): es holt die Maschinenbefehle aus dem Speicher, den Instruction Stream, interpretiert diese und setzt sie in zugehörige Steueralgorithmen um. Das Leitwerk übernimmt damit als Befehlsprozessor die Steuerung der Instruktionsausführung.
- Rechenwerk (Processing Unit): es holt die Daten aus dem Speicher bzw. von der Eingabe (Data Stream), transformiert diese Daten mittels unterschiedlicher Steueralgorithmen und legt sie im Speicher ab bzw. übergibt sie der Ausgabe. Als eigentlicher Datenprozessor realisiert es die logischen und arithmetischen Operationen. Es wird auch als arithmetisch-logische Einheit (ALU) bezeichnet.
- Speicher (memory): er enthält die Maschinenbefehle und die zu verarbeitenden Daten. Eine Folge von Maschinenbefehlen bezeichnen wir als Programm. Sowohl Befehle wie auch Daten befinden sich in einem gemeinsamen Adressraum. Der Speicher dient der kombinierten Ablage von Programmen und Daten.
- Ein und Ausgabe: Sie verbindet die Peripheriegeräte wie Tastatur, Monitor mit dem Rechenwerk, stellt also eine oder mehrere Schnittstellen zur Umwelt dar. Ursprünglich wurden das Eingabe- und Ausgabewerk als zwei getrennte Grundbaublöcke betrachtet.

Häufig werden Rechenwerk und Leitwerk als Prozessor oder CPU bezeichnet (analog dazu soll der Soziologieprofessor, der Lehrstuhl, das Institut oder eine andere Steuer- und Ausführungseinheit gesehen werden).

Es sind zwei (bzw. vier) Schritte, die nacheinander vom Prozessor abgearbeitet werden: Befehl aus dem Befehlsspeicher holen = Abholen und Interpretieren des Befehls; Daten aus dem Datenspeicher verknüpfen=Befehl ausführen.

Daneben existiert z.B. die Harvard Architektur: im Unterschied zum NR ist der Speicher in einem separaten Programm- und Datenspeicher aufgeteilt. Es stehen damit getrennte Zugriffspfade für Befehle und ihre Operanden zur Verfügung. Dies muss mit einem Hardware-Mehraufwand erkauft werden, da für den Datentransport doppelt so viele Leitungen notwendig sind (Glatz, 2010: S.640 f.). Klar ist, dass Architekturen einer Variation unterliegen, und diese Vor- und Nachteile aufweisen können; und dass sie etwa aufgrund von Verfassungen, Normen, Naturgesetzen etc. nicht immer möglich und realisierbar sind.

Der Prozessor verarbeitet Informationen im einfachsten, konventionellen Fall, also nach folgendem Ablauf:

1. Befehl (aus dem Speicher in das Befehlsregister) holen
2. Befehl interpretieren
3. Aufgrund der Interpretation des Befehls nötige Operanden holen
4. Befehl (Operation) ausführen

Anschließend wird dieser Ablauf mit dem nächsten Befehl durchgeführt (Spaniol et al., 2011: S.2). Diese Rangfolge zu ändern, weiter zu entwickeln und auszubauen, wird weiter unten angesprochen.

Nächste Punkte im Referat Computerarchitekturen sind der Speicher, darauf das Pipelining, die Ausführungssteuerung, die Datenübertragung und die Busse inkl. Schaltungen.

Man unterscheidet im Wesentlichen drei Arten von Speicher.

Der Hauptspeicher (Primary Memory, Main Memory) ist direkt adressierbar mit nahezu gleicher Zugriffszeit pro Zelle (typische Werte liegen bei einigen Nanosekunden). Er wird demnach auch als Random Access Memory (RAM) bezeichnet.

Einen Hintergrundspeicher (Secondary Memory) verwendet man zur Speicherung relativ selten benötigter Informationen. Er ist langsamer bezüglich der Zugriffszeit. Da er nicht die teuren und schnellen Speicherbausteine verwendet, ist er wesentlich billiger und daher auch in größerem Umfang verfügbar. Als Beispiel ist der Festplattenspeicher zu nennen.

Der Pufferspeicher (Cache Memory) zeichnet sich dadurch aus, dass er besonders schnell, dafür aber auch teuer ist. Aus diesem Grund ist er relativ klein. Seine besondere Eignung liegt in der Pufferung (vom Prozessor) besonders häufig benutzter Befehle und Daten.

Neben den Speichertypen gibt es die Register. Es handelt sich dabei um Zellen mit besonders kurzen Zugriffszeiten (Spaniol et al., 2011: S.3).

Analog können etwa die geistigen und finanziellen Voraussetzungen von Soziologen gedacht werden. Die Erfahrung bzw. Professionalität und Argumentationsstrategien des Soz-akteurs im Rahmen der Algorithmik (jenseits der Signifikanzrprüfung) werden durch laufende Theorienaneignungen etc. zu den immer kostspieligeren Faktoren gezählt.

3.4 Pipelining, Ausführungssteuerung, Datenübertragung und Leitungen, Busse, Schaltungen

Pipelining

Die Befehlsausführung nach dem Von-Neumann-Prinzip in vier Schritten kann beschleunigt werden, wenn man Schritt 1 und 2 des nachfolgenden Befehls mit den Schritten 3 und 4 des aktuell auszuführenden Befehls kombiniert (Spaniol et al., 2011: S.5). Von Neumann wird in dieser Arbeit etwa die selbe Gründervaterposition zugerechnet, wie sie Max Weber für die Soziologie bekanntermaßen innehat. Um keinen Rückschritt zu implementieren, wird die Von Neumann Leistung oder die von Max Weber, natürlich auf die aktuelle Rechen- bzw. Formulierungssituation der Sozialforschung approximativ oder imitativ angewandt. Dilettantisches Epigonentum nicht zu umgehen, ist per Definition das Problem der Imitation bzw. Simulation.

Ausführungssteuerung

Neben den Befehlen zum Datentransfer zwischen Prozessor und Speicher werden auch die Ein- / Ausgabegeräte (Input / Output, I / O) über E- / A-Befehle vom Prozessor angesteuert. Ihre Tätigkeit können solche Endgeräte dann weitgehend unabhängig vom Prozessor ausführen. Eine wichtige Aufgabe hat der Prozessor bei der Koordination dieser Geräte. Hierzu senden die Endgeräte sogenannte Interrupt-Signale, die den Prozessor in der Ausführung eines anderen Programms unterbrechen. Auf diese Weise kann von der normalen Befehlsausführung abgewichen werden.

Beispiele für derartige Unterbrechungen des Prozessors sind eine Alarmmeldung, die eine Reaktion innerhalb einer nicht zu überschreitenden Zeitspanne (Deadline) erfordert, oder eine Fertigmeldung eines langsamen Endgerätes. Der vergleichsweise schnelle Prozessor bedient normalerweise viele langsame Endgeräte gleichzeitig, er muss also zwischen gleichzeitig laufenden Programmen umschalten (Spaniol et al., 2011: S.6).

Datenübertragung und Leitungen

Die Kommunikation besteht im Wesentlichen im Transfer von Daten und Steuerinformationen zwischen den funktionellen Einheiten des Rechners, insbesondere zwischen Prozessor, Speicher und E- / A-Geräten. Sie erfolgt über Leitungen, welche die Einheiten verbinden. Physikalisch gesehen sind Leitungen meist drahtgebunden.

Aber auch andere Möglichkeiten existieren wie etwa Funkstrecken. Informationstransport auf einer Leitung erfolgt in der Regel sequenziell, d.h. ein Bit wird nach dem anderen auf die Leitung gebracht und ebenso empfangen. Der parallele Transport bedeutet, dass mehrere Bits gleichzeitig über ein Bündel von Leitungen gesendet werden. Ein solches Bündel wird auch als Bus bezeichnet.

Busse

Ein Rechnerbus besteht aus einer Menge von Leitungen. Ein Bus ist im Rechner als öffentliches Transportmedium für mehrere Einheiten aufzufassen, die über diesen miteinander verknüpft sind. Der Zugriff auf den Bus, also die Busbenutzung, wird daher auch als Multiple Access (MA) bezeichnet und muss durch Konventionen in Form von Protokollen geregelt werden.

Busse lassen sich nach den Einheiten charakterisieren, die sie miteinander verbinden. Ebenso gibt es verschiedene Busarchitekturen, welche die Zuordnung von Bussen an Einheiten beschreiben. Der Datenbus z.B. dient der Kommunikation zwischen Prozessoreinheit und dem Speicher.

Ein eigener Rechnerbus sorgt für die Kommunikation des Prozessors mit den E- / A-Geräten.

Einfacher, aber weniger leistungsfähig als diese Zwei-Bus-Architektur ist die Verwendung eines gemeinsamen Busses – so genannter Unibus – für alle Kommunikationsaufgaben zwischen allen Einheiten.

Ein grundlegendes Problem ist die unterschiedliche Geschwindigkeit, mit der die Komponenten arbeiten, die an einen gemeinsamen Bus angeschlossen sind, zum Beispiel: schnelle CPU und langsames E- / A-Gerät. Benutzt eine langsame Einheit den Bus, so ist er für eventuell längere Zeit für andere Einheiten blockiert. Dies erklärt auch, warum der Unibus nicht sehr leistungsfähig ist.

Erforderlich sind also Hilfsmittel, welche die Geschwindigkeitsunterschiede glätten, um weniger Busse verwenden zu können. Ein Lösungsansatz besteht im Einsatz von Pufferregistern zwischen dem Bus und den angeschlossenen Geräten. Auf diese Weise kann zum Beispiel eine schnelle CPU viele langsame Endgeräte wechselweise bedienen. Der Bus ist allerdings nur solange nicht blockiert, bis der Puffer voll ist (Spaniol et al., 2011: S.7 f.).

Es ist verständlich, dass die wenigsten Soziologen einen wissenschaftlichen Artikel die Woche, am Tag etc. anfertigen können. Tun sie dies, muss die Qualität etwa der Zeitungsebene angepasst werden. Wenn also bestimmte schnelle Prozessorleistungen nicht über langsame Helfergeräte etc. gehen, sondern unabhängig extern terminiert werden, würde hier die CPU Leistung ungeschmälert genutzt werden können. Das Angehen theoretischer Themen, ohne sich mit Variablenrechnung, Formatierungen etc. die Finger schmutzig zu machen, oder, auf der anderen Seite, zeitaufwendige Simulationsrechnungen, wären Beispiele dazu. Dass die Dauer einer Fernleihe die Arbeit von Diplomanden verzögern kann, ist häufig der Fall, bei kreativen, selbst gesteuerten Denkleistungen seltener.

Schaltungen

Schaltfunktionen und Bausteinsysteme sollen im folgenden die Ebene der digitalen Logik verständlich machen.

Die Aufgabe einer Schaltung besteht in der Transformation (binär dargestellter) Eingangssignale in (binär dargestellte) Ausgangssignale. Eine Schaltung ist daher mit einem Programm vergleichbar, mit dem Unterschied, dass die Transformation durch die Schaltung fest verdrahtet ist. Eine Schaltfunktion beschreibt die Zuordnung von Ausgangswerten zu Eingangswerten (Spaniol et al., 2011: S. 47).

Besonders einfache Schaltfunktionen lassen sich als Grundbausteine (Bausteinfunktionen) zur Zusammensetzung komplexerer Funktionen verwenden. So werden die Konjunktion (AND), die Disjunktion (OR) oder die Negation (NOT) als Beispiele genannt.

Eine Menge von Bausteinfunktionen, aus denen sich beliebige Schaltfunktionen zusammensetzen lassen, heißt Bausteinsystem.

Schaltfunktionen lassen sich mit diesen Grundbausteinen zu beliebigen neuen Schaltfunktionen zusammensetzen (Spaniol et al., 2011: S.50 f.).

Schaltkreise bestehen aus Eingangsleitungen, Schaltnetz und Ausgangsleitungen. Schaltwerke bestehen aus Schaltkreisen, Speicherelementen und Taktleitungen (Spaniol et al., 2011: S.72).
Wir haben hier also die digitale Logik angesprochen, die in ihrer Vielzahl logische Operationen simuliert. Eine solche Hardware ist äquivalent zu einer logischen Software, die verstanden worden ist.

3.5 Erweiterung des Von-Neumann-Berechnungsmodells

Das Basis-von-Neumann-Modell kann durch eine Menge von Unterklassen erweitert werden:
- Realisation von gleichzeitig ausführbaren Teilen der Berechnung, wie z.B. Prozesse oder Threads
- gleichzeitige Ausführung kann bereitgestellt werden für:
1. Kommunikation
2. Synchronisation
3. gemeinsame Nutzung von Daten;
zu 1.: der Kommunikationsmechanismus erlaubt die Übertragung von Daten zwischen den ausführenden Einheiten (execution units).
zu 2.: der Synchronisationsmechanismus erlaubt es, den Ausführungssequenzen Einschränkungen aufzuerlegen. Beispiele:
– Semaphore (semaphores)
– Signale (signals)
– Ereignisse (events)
– Warteschlangen (queues)
– Synchronisationssperren (barrier synchronization).
zu 3.: ein Mechanismus für die gemeinsame Nutzung von Daten ist notwendig, um zu vereinbaren, ob:
– parallel ausführbare Einheiten (Prozesse) auf einen gemeinsamen Datenbereich zugreifen dürfen;
– verschiedene Prozesse nur Zugriff auf ihre lokalen Datenräume haben. In diesem Fall sollte ein Kommunikationsmechanismus für den Zugriff auf Daten, die zu anderen Prozessen gehören, zur Verfügung gestellt werden (message passing concept) (Glende, 2003: S.10).

Parallele Be- / Verarbeitung auf Befehlsebene

Die Entwicklung des von-Neumann-Prozessors wird gekennzeichnet durch:

1. die Steigerung der Taktrate und der Dichte der Transistoren auf einem Chip
2. die funktionale Entwicklung durch die Steigerung des Parallelismusgrades auf Betriebsebene (operation level).

Die Steigerung des Parallelismus auf Betriebsebene fand wie folgt statt:

1. traditioneller von-Neumann-Prozessor (sequentielle Ausführung)
2. Scalar-Prozessoren, die multiple Ausführungseinheiten (EUs) und die Pipeline-Technologie umfassen
3. Superscalar-Prozessoren (VLIW-Prozessoren, Dual-Pipeline-Prozessoren).

Superscalar-Prozessoren waren erstmals implementiert als:

– statische festgelegte Very Long Instruction Word Architekturen (VLIW); für diese Architekturen muß ein „Trace Scheduling-Compiler“ einen langen Befehlswort-Code erzeugen;

– komplexere Prozessoren, die auf folgendes basieren:

a) dynamischer Planerstellungsmechanismus, der die Ausführung von mehreren Befehlen pro Zyklus erlaubt;

b) super-pipelining: eine Technik, um mehr als eine Pipeline beschäftigt zu halten (Glende, 2003: S.22).

Diese Entwicklungen (und natürlich andere) haben in den letzteren Jahren der Computerindustrie stattgefunden. Auf die Sozialtheorie oder deren Produktion bezogen, können Mitautorenschaften, Herausgeberschaften, Exzellenzgruppen, Arbeitskreise, Zunahme der Sponsoren etc. in einer ähnlichen, superskalaren Tonart ausgesprochen werden. Es wird aber nicht offensichtlich, dass das grundlegende Primat des einen Denkers oder aufgeklärten Soziologen, Diplomanden etc. abgesetzt worden wäre, welcher im Kontext von verteilten Systemen beständig eine architektonische Leistung, auch über seinen Schreibtisch hinaus, zu liefern in der Lage ist. Diese Rekonstruktion (auf einer höheren Ebene) sollte angestrebt werden, damit besprochene „Fortschritte“ in der E-Technik für die Sozialforschung Anwendung finden können.

3.6 Multiprozessorsysteme

Es folgt eine Einführung und später die Unterscheidung von sogenannten Multiprozessorsystemen. Dabei ist der Blickwinkel der Leistungssteigerung unablässig zu halten.

Klassifikation nach Flynn (Glatz, 2010: S.642 ff.)
Vier Wahrheitswerte und somit unterschiedliche Rechnerklassen sind gegeben durch:
Befehlsstrom =I,
0=die Maschine bearbeitet in einem gegebenen Zeitpunkt genau einen Befehl
1=die Maschine bearbeitet in einem gegebenen Zeitpunkt mehr als einen Befehl;
Datenstrom=D,
0=die Maschine bearbeitet in einem gegebenen Zeitpunkt genau einen Datenwert
1=die Maschine bearbeitet in einem gegebenen Zeitpunkt mehr als einen Datenwert.
Zusammengesetzt ergeben die ID Werte: 00,01,10,11: die entsprechenden Typen sind: SISD, SIMD, MISD, MIMD.
SISD: Einprozessorsysteme – Ein Befehls und ein Datenstrom;
SIMD: eine CU und mehrere PU (array processor, vector processor);
MIMD: verteilte Systeme; Multiprozessorsysteme: eine Reihe von CPUs, die ihre Befehls- und Datenströme aus einem (wenigstens teilweise) gemeinsamen Speicher holen; Supercomputer; eine Steigerung der Rechenleistung über eine derartige Parallelisierung durch die Verwendung von Standard MP, ist besonders kostengünstig;
MISD: mehrere Befehle, ein Datenwert (Glatz, 2010: S.642 ff.).

Eine feinere Unterteilung von MIMD Systemen ergibt sich, wenn man folgende Merkmale bedenkt:

- Homogenität der Prozessoren
- Aufgabenteilung der Prozessoren (z.B. Asymmetric Multiprocessing)
- Kopplung der Prozessoren
- Anzahl der Prozessoren (100 Prozessoren=massiv parallel) (Glatz, 2010: S.644 f.).

Die Rechenleistung paralleler Prozessoren kann man unterschiedlich operationalisieren, etwa über die Rechenarbeit (die Menge von Maschinenbefehlen, die ausgeführt werden muss); Leistung als Anzahl pro Zeit oder die CPU Auslastung. Nach dem Gesetz von Amdahl lässt sich die mittlere Leistung eines Multiprozessorsystems bei bekanntem seriellen Anteil s und einer Anzahl von Prozessoren n berechnen (Glatz, 2010: S.645 ff.).

Leistungsbewertung allgemein kann Hardware oder Software angehen:

Hardware: Hauptspeichergröße, Cache-Größe, Festplatten-Geschwindigkeit, Systembus, Peripheriebus etc.; Software: Betriebssystemversion, verwendete Compiler (Glatz, 2010: S.654).

Eine weitere Unterscheidung der MIMD Systeme ist wie folgt realisierbar:

- Mehrere Prozessoren im gleichen Chip
- Mehrere Prozessoren auf der gleichen Systemplatine
- Mehrere Prozessorkarten mit je einem CPU im gleichen Gehäuse
- Mehrere separate Computer lokal oder über Internet vernetzt (Rechner sind in der Regel Einprozessorsysteme) (Glatz, 2010: S.648).

Letzter Punkt wird einem realen Diskursmodell, erstere einem “Idealmodell des Diskurses” nahekommen. Dem soll später mehr Beachtung gewidmet werden.

Bei sogenannten Shared-Memory-Multiprozessoren können die Hardwarebauformen auf zwei Grundtypen zurückgeführt werden:

- Uniform memory access: der Zugriff auf den gesamten Speicher ist einheitlich schnell. Dies entspricht dem theoretischen Ideal.
- Non Uniform Memory Access: der Zugriff auf den gesamten Speicher ist, abhängig vom Adressbereich, unterschiedlich schnell. Diese Lösung ist nicht mehr ideal, aber kostengünstiger zu realisieren.

Für SMM sind drei Betriebssystemtypen denkbar:

- Jede CPU hat ihr eigenes OS, der Speicher wird in Partitionen pro CPU aufgeteilt.
- Master / Slave Multiprozessoren: das Betriebssystem läuft nur auf einer einzigen CPU.
- Symmetrische Multiprozessoren: nur eine einzige Kopie des Betriebssystems liegt im Speicher, diese ist von jeder CPU ausführbar (Glatz, 2010: S.650 ff.).

Eine weitere Gruppe, die Message-Passing-Multicomputer, auch Clustercomputer, cluster of workstations, tightly coupled multiprocessors genannt, soll kurz erwähnt werden.

Was den Nachrichtenaustausch, message passing angeht, verfolgt man unterschiedliche Ansätze:

- Low-Level-Kommunikationsoftware: primäres Ziel ist das Vermeiden des unnötigen Herumkopierens von Daten zwischen Betriebssystem und Anwenderprozessen, indem die Nachrichtendaten und Schnittstellencontroller direkt in die Adressräume der einschlägigen Anwenderprozesse eingeblendet werden.
- Kommunikationssoftware auf Benutzerebene: den Anwenderprozessen werden elementare Kommunikationsdienste in Form von zwei Systemaufrufen send und receive zur Verfügung gestellt.
- Remote Procedure Call: diese Softwareschicht erweitert die Fähigkeiten des Rechners derart, dass nicht nur lokale Prozeduren, sondern auch Prozeduren in entfernten Systemen aufgerufen werden können.
- Distributed Shared Memory: Diese Software simuliert einen gemeinsamen Speicher gegenüber den Applikationen (Glatz, 2010: S.652 f.).

Zuletzt wird auf das Konzept verteiltes System (loosely coupled multiprocessors) eingegangen. Jeder Knoten im verteilten System ist ein vollwertiger Computer. Ein wichtiger Nachteil ist das fehlende gemeinsame Programmiermodell. Eine Middleware als eine Zwischenschicht wird eingeführt, um an unterschiedlichen OS und Prozessoren aufzusetzen, und eine einheitliche Schnittstelle zu bieten (Glatz, 2010: S.653).
So werden z.B. in der Migrationsforschung zahlreiche Ergebnisse unterschiedlicher Forscher abgefragt, hinsichtlich der Integrationstheorie oder der theoretischen Modellierung können im Sinne der Komplexitätsbegrifflichkeit alle Ergebnisse auf einer allgemeineren Ebene unbrauchbar gemacht oder als wertlos postuliert werden, indem man z.B. Argumente ins Spiel bringt, die das System natürlicherweise überlasten, aufgrund einer fehlenden Middleware nicht akzeptieren etc. (siehe dazu die Integrationsprobleme bzw. –trivialität im Fragebogen im Anhang).

Vergleich der drei Multiprozessor-Grundmodelle (Glatz, 2010: S.654)

Element	Multiprozessor	Multicomputer	Verteiltes System
Knotenkonfiguration	CPU	CPU, Speicher, Netzwerk	Vollständiger Rechner
Knotenperipherie	Alle aufgeteilt	Geteilt, evtl. selbe Platte	Vollständiger Knoten
Ort	Selber Einschub	Selber Raum	Evtl. weltweit
Kommunikation der Knoten	Geteilter Speicher	Spezielle Verbindung	Herkömmliches Netz
Betriebssysteme	Eins, gemeinsam	Viele gleich	Evtl. alle verschieden
Dateisystem	Eins, gemeinsam	Eins, gemeinsam	Jeder Knoten eigenes
Administration	Eine Organisation	Eine Organisation	Viele Organisationen

Ein letzter Abschnitt über das Handheld Computing (Glatz, 2010: S.656 f.) wird den technischen Teil dieser Arbeit abschließen.

Vier zentrale Paradigmen werden in diesem Kontext betrachtet:

- Dezentralisierung: in klassischen zentralen Systemen ist die Intelligenz in einem Großrechner konzentriert, mit dem über dumme Datensichtstationen kommuniziert wird. In der PC-Welt verteilt sich die Intelligenz bereits auf ein zentrales System oder server, das Dienstleistungen anbietet, und dezentrale PCs oder clients. Letztere betreiben eigene, auf zentralisierte Dienstleistungen aufbauende Applikationen. Neu realisiert man heterogene Rechenanlagen, bestehend aus PCs, Großrechnern und HH-Computers, in denen die Intelligenz unterschiedlich verteilt sein kann = verteilte Systeme. Das gegenseitige Aufsynchronisieren von Informationsbeständen und die Aktualisierung verteilter Applikationen sind damit wichtige zu erfüllende Aufgaben aus betrieblicher Sicht (und ebenso aus Sicht der Sozialforschung).
- Diversifikation: gegenwärtig werden primär universell verwendbare Rechner gekauft. In Zukunft werden vermehrt spezialisierte Computer im Vordergrund stehen, die hauptsächlich für eine bestimmte Anwendung gebaut sind, z.B. Handys; eine

Diversifizierung der I- / Omedien ist zu erwarten, z.B. Kleintastatur, Touchscreen (z.B. allgemeine Groß-Soziologie vs. spezielle Bindestrichsoziologien und deren Unterteilungen; Datengewinnungsweisen).

- Konnektivität: entsprechend der Rechnerart in einer heterogenen Umgebung werden unterschiedliche Kommunikationssysteme eingesetzt, z.B. kabelgebunden, Funknetze; Standards zur Einheitlichen Darstellung von Informationen. Eine organisierte Themenvor- und Vergabe für die Soziologie kann in der Hinsicht gedacht werden.
- Einfachheit: neue Versionen bekannter Softwarepakete werden oft über zusätzliche Fähigkeiten als Verbesserung des Bestehenden verkauft. Damit wird meist die Bedienung der Produkte komplizierter, wobei der Durchschnittsbenutzer die Extrafähigkeiten kaum nutzt (versteht, in seine Arbeit integriert etc.) (Glatz, 2010: S.656 f.).

4. Diskurse und Inhalte

Der letzte Abschnitt des Arbeitspapiers versucht die zitierten Konzepte und Begriffe der Informatik in den Kontext der Sozialforschung zu stellen, und Schlussfolgerungen im Hinblick auf Vor- oder Nachteile, Verbesserung der Strukturen, des Diskurses, der machbaren oder gewünschten Potentiale etc. zu finden.

4.1 Kommunikation und Konversation

Einleitend wird ein sogenannter verbindungsorientierter Dienst zitiert, dem ein nachvollziehbares Beispiel zur Seite gestellt ist. Seine acht Operationen werden auf den Diskursbegriff angewendet.

1. Aufbau einer Verbindung anfordern
2. Gerufene Partei benachrichtigen
3. Anfrage durch gerufene Partei annehmen
4. Rufende Partei über die Annahme der Verbindung benachrichtigen
5. Mitteilen, dass die Daten gesendet werden können
6. Ankunft der Daten anzeigen
7. Anfrage, die Verbindung zu lösen
8. Partner benachrichtigen, dass die Verbindung gelöst wird.

Beispielinhalt:

1. Tantes Telefonnummer wählen
2. Ihr Telefon läutet
3. Sie nimmt den Hörer ab
4. Sie hören, dass das Klingelzeichen aufhört
5. Sie laden sie nächstes Wochenende zum Tee ein
6. Sie hört Ihre Einladung
7. Sie sagt, dass sie gerne kommt
8. Sie nehmen ihre Zusage zur Kenntnis
9. Sie legen den Hörer auf
10. Sie hört das und legt ebenfalls auf (Tanenbaum, 2000: S. 43 f.).

Schnell sieht man ein, dass hier sprachliches Kommunizieren auf technischer Ebene bzw. durch diese realisiert wurde. Der Kontext zur Argumentation oder Diskurs wird auf den nächsten Seiten ausgeführt. Dabei wird angenommen, dass wenn der (inhaltslose) Diskurs materialisiert und machbar ist, auch dessen Inhalt, Wert und Bedeutung durch Anwendung entsprechender Konzepte von technischen Mitteln bzw. Algorithmen etc. kumuliert werden kann.

Sprechakte und Sprechhandlungen

Als elementare Sprechakte kann man Sprechhandlungen wie etwas behaupten, ankündigen, androhen, fragen, versprechen oder verfluchen anführen. Die argumentative (komplexe) Sprechhandlung unterscheidet sich von diesen durch drei Merkmale:

1. es ist immer mehr als eine Proposition / ein Satz vorhanden
2. es gibt immer eine doppelte kommunikative Funktion des Sprechaktes
3. es liegt eine Interrelation der Sprechakte bzw. der notwendige Bezug zur Zielaussage vor (vgl. Mihajlovic, 2008: S.84 ff.).

Bedingungen sprachlichen Kommunizierens (Schöberle, 1984: S.23)

A kommuniziert verbal mit B, d.h. A erreicht, dass B versteht, was A mit einer sprachlichen Äußerung meint, wenn:

1. A verwendet einen sprachlichen Ausdruck y
2. A meint y als x (d.h. z.B. als Aufforderung, etwas zu tun, als Mitteilung, dass etwas der Fall ist etc.)
3. B nimmt an, dass y als x bzw. x‘ zählt
4. A nimmt an, dass y als x bzw. x‘ zählt
5. A nimmt an, dass 3. gilt
6. B nimmt an, dass 4. Gilt.

Diese Punkte sind im Kontext von Computernetzwerken bereits eingespeist. Während des Aufbaues von verteilten Systemen sind noch mehr Forderungen des argumentativen Wissenschaftsspiel mehr oder weniger gut in Hardware bzw. Software eingegangen oder übersetzt worden.

So z.B. die Konversationsmaximen von Grice, die jedoch keine besonderen Aspekte der Informationsaufnahme, des Anhörens erwähnen.

Allgemeines Kooperationsprinzip:

Make your conversational contribution such as is required, at the stage at which it occurs, by the accepted purpose or direction of the talk-exchange, in which you are engaged.

Submaximen in 4 Kategorien:

1. Quantity:

- ¸ Make your contribution as informative as is required (Variablendefinition, Minimierung von von Operationen, Berechenbarkeit, Compilierbarkeit (Verständlichkeit); ausgedehnt aber im Verschlüsselungskontext)
- ¸ Do not make your contribution more informative than is required (Redundanz)

5. Quality

- ¸ Do not say what you believe to be false (Verifizierung, Automat als Sprachenerkenner, bestimmtes Schaltungsmuster als Regel, Terminierung des Algorithmus)
- ¸ Do not say that for which you lack adequate evidence (Überlastung, Identifikation, Rauschen, Protokollmissbrauch)

3. Relation

- ¸ Be relevant

4. Manner

- ¸ Avoid obscurity of expression (Sprachen, Grammatiken, Entscheidbarkeit, Interpretation)
- ¸ Avoid ambiguity (Logik, Semantik, Compiler)
- ¸ Be brief (avoid unnecessary prolixity) (Laufzeit, Komplexität)
- ¸ Be orderly (Syntax).

Kurz: Wahrheit, Ernsthaftigkeit, Aufrichtigkeit und Wahrhaftigkeit, Verständlichkeit, Relevanz.

Halte eine bestimmte Anzahl von kombinierbaren Regeln ein, um die Konversation weiter abzuspulen oder nicht abbrechen zu lassen. Dies ist natürlich im technischen, funktionalen Rahmen, im Sport, in jedweder (symmetrischen) Beziehung (etwa das AGIL-Schema, These-Antithese-Syntheseabläufe etc.) leicht übersetzbar.

Sind sehr viele “Teilnehmer” an der Konversation beteiligt, wie bei komplexen Computernetzwerken der Fall ist, werden “technische” Maßnahmen erforderlich, um das Kommunikationssystem nicht überlasten zu lassen.

Maßnahmen in Bezug auf die Überlastungsüberwachung (Tanenbaum, 2000: S.408)

Schicht (ohne weitere Erklärungen)	Maßnahmen
Transporterschicht	• Erneute Übertragung • Zwischenspeichern von außer der Reihe gesendeten Paketen • Bestätigungen • Flußsteuerung • Timeout
Vermittlungsschicht	• Virtuelle Vebindungen gegenüber Datendiagrammen im Teilnetz • Warteschlangen und Dienste für Pakete • Verwerfen von Paketen • Routing-Algorithmus • Verwaltung der Lebensdauer von Paketen
Sicherungsschicht	• Erneute Übertragung • Zwischenspeichern von außer der Reihe gesendeten Paketen • Bestätigungen • Flußsteuerung

4.2 Die ideale Sprechsituation

Die ideale Sprechsituation nach Habermas (vgl. Hilgendorf, 1990: S.206 f.) wird gekontert durch die ideale (ebenfalls nicht reale) Hörsituation (die eine Komplexitätsexplosion, eine Endlosverschleifung auszeichnen soll). Zwei Personen (eine Gemeinschaft) und hundert oder mehr Personen (Gesellschaft) unterscheiden sich bei der Nachrichtenverifikation, deren Überwachung, Fehlerkorrektur usw. und begründen trivial die Konzepte Gemeinschaft, Gruppe, Gesellschaft, Weltsystem, Umwelt, Normen, realisierbare Handlungen etc., somit soziologische Grundbegriffe. Eine reale Person kann maximal n * xy täglich sprechen, anhören usw., ansonsten wird sie überlastet.

Eine Sprechsituation, in der Kommunikationen nicht nur nicht durch äußere kontingente Einwirkungen, sondern auch nicht durch Zwänge behindert werden, die sich aus der Struktur der Kommunikation selbst ergeben; und für alle Diskursteilnehmer eine symmetrische Verteilung der Chancen, Sprechakte zu wählen und auszuführen, gewährleistet wird, soll eine ideale heißen. So ist man als Hörer aber ganz klar benachteiligt, und zwar weil sich dies aus der Struktur der Kommunikation ergibt. Die symmetrische Verteilung der Chancen etc. kann überhaupt nicht realisiert werden, ist also nicht annähernd als eine rationale oder ideale zu verstehen. Die technischen Abläufe und Organisationen haben dies weiter oben andeuten können.

Der lebende menschliche Nutzer bzw. Architekt kann nach gewissen Architekturleistungen abgesetzt, und die Kriterien der in etwa idealen (Computer-) Sprechsituation endlos beibehalten werden (abgesehen von Stromzufuhren, Abnutzungen der Hardware, Übertragungsschichten oder Verlusten etc.), wenn dabei das Konzept der (gesteuerten) verteilten Systeme realisiert wird. Nachrichten, die übertragen (kopiert, weitergeleitet) werden sollen, heißen Argumentation, Gefühle, Wünsche, Meinung, Regel etc. Teilnehmer sind Nachrichtensender und Empfänger, eingebettete Compilierer usw. Der Diskurs (entsprechend das Handeln der Teilnehmer in Schleifen) kann durch Zufallsgeneratoren “angeheizt” werden. Computer als verteilte Systeme, auf anderer Ebene als Facebook, führen einen herrschaftsfreien Diskurs, so die Annahme. Die (humanoiden) Inhalte sind leider ausgeblieben. Computer als solche erkennen die Kriterien der Herrschaft oder Idealität als die selben an.

Kriterien der idealen Sprechsituation (der Begriff gleiche Chance weiter unten meint natürlich eine ungleich 0; 100 Teilnehmer könnten also zu Beginn 1% bekommen, 1 Diskurs zu eröffnen, und 99 Gegenreden zu bekommen; wer darf als erster?)

1. Alle potentiellen Teilnehmer eines Diskurses müssen die gleiche Chance haben, kommunikative Sprechakte zu verwenden, so dass sie jederzeit Diskurse eröffnen sowie durch Rede und Gegenrede, Frage und Antwort perpetuieren können (verteilte Systeme, Vergleich der Systeme, Middleware, Ressourcen, Protokolle, Standards, Vermeidung von Ausfällen durch Überlastung (z.B. jede Rede erhält eine Kurzform einer Gegenrede), Geschlossenheit der Netzwerke etc., Strom).

2. Alle Diskursteilnehmer müssen die gleiche Chance haben, Deutungen, Behauptungen, Empfehlungen, Erklärungen und Rechtfertigungen aufzustellen und deren Geltungsanspruch zu problematisieren, zu begründen oder zu widerlegen, so dass keine Vormeinung auf Dauer der Thematisierung und der Kritik entzogen bleibt (Algorithmik, Datenstrukturen, logische Argumentationen, Entscheidbarkeit von Problemen, Verifizierer, Automaten und einschlägige Sprachen, Compiler, Übersetzung, Verschlüsselungen als höchste Argumentationen, gemeinsame Speicher, gemeinsame Betriebssysteme, Provider, Vermittler etc.; Notwendigkeit der Komplexitätsexplosion durch jedermanns Kritik einer jeder Meinung, durch Wachstum des Wissens usw.).

3. Zum Diskurs sind nur Sprecher zugelassen, die als Handelnde gleiche Chancen haben, repräsentative Sprechakte zu verwenden, d.h. ihre Einstellungen, Gefühle und Wünsche zum Ausdruck zu bringen (Netzwerke über Client / Serverstrukturen: hier verfügen Clients über mehr separate (ungenutzte) Ressourcen, Verifizierung der intelligenten Gefühlsäußerung, P2P, “Archive der Einstellungen und Meinungen” durch Input über Nutzer; Strukturierung der dummen (fehlerhaften) und weniger dummen Automaten).

4. Zum Diskurs sind nur Sprecher zugelassen, die als Handelnde die gleiche Chance haben, regulative Sprechakte zu verwenden, d.h. zu befehlen und sich zu widersetzen, zu erlauben und zu verbieten, Versprechen zu geben und abzunehmen, Rechenschaft abzulegen und zu verlangen etc. (Diskurs wird betrachtet als Summe von gleichen Elementen, über Funknetze, Glasfaserkabel addiert; über Protokolle und Standards organisiert; Kooperation vs. Konkurrenz, alterierende Agent-Prinzipal-Systeme, Endlos-Schleifen, verteilte Systeme

(Dateien, Speicher) ohne “Körper”, Max Weber (im Sinne von: Gott diskutiert nicht) kann von jedermann zitiert werden).

4.3 Regelkanon für den allgemeinen praktischen Diskurs

Auch der Regelkanon für den allgemeinen praktischen Diskurs nach Alexy (zitiert über Hilgendorf, 1990: S.225 ff.), dürfte von jedem Computer, und erst recht von Informatikern, akzeptiert werden.

Grundregeln

- Kein Sprecher darf sich widersprechen (Logik, Boolesche Algebra, Hardware)
- Jeder Sprecher darf nur das behaupten, was er selbst glaubt (Automaten und entsprechende Sprachen / Grammatiken)
- Jeder Sprecher, der ein Prädikat P auf einen Gegenstand a anwendet, muss bereit sein, P auch auf jeden anderen Gegenstand, der a in allen relevanten Hinsichten gleicht, anzuwenden (im Sinne objektorientierter Programmierung)
- Verschiedene Sprecher dürfen den gleichen Ausdruck nicht mit verschiedenen Bedeutungen benutzen (nicht in der Pipeline, Variablenbedeutungen sind von Programm zu Programm verschieden, “Scherzfragen”).

Vernunftregeln

- jeder Sprecher muss das, was er behauptet, auf Verlangen begründen, es sei denn, er kann Gründe dafür anführen, die es rechtfertigen, eine Begründung zu verweigern (Compilierung; der Code ist die Begründung, die Lösung, die Regel, und muss den Schaltkreisen entsprechen; Schnittstellenproblem)
- Jeder, der sprechen kann, darf an Diskursen teilnehmen (Vernetzung, Protokolle, Schnittstellen)
- Jeder darf jede Behauptung problematisieren (Interpretation, Übersetzung, Akzeptanz der Sprache, des Problems; Laufzeit und Platz der Thematisierung)
- Jeder darf jede Behauptung in den Diskurs einführen (Nachrichten nach Uhrzeit senden)

- Jeder darf seine Einstellungen, Wünsche und Bedürfnisse äußern (100 Sender senden jeder zuerst 99 gleiche Ich-Nachrichten, empfangen 99 unterschiedliche Nachrichten, Beantwortung, Weiterleitung dieser Nachrichten, bedingte Nachrichten, Pfade, Bäume; Gedächtnis, Register der Einstellungen; Komplexitätsexplosion über die nächsten Stufen; Speicherlöschung)
- Kein Sprecher darf durch innerhalb oder außerhalb des Diskurses herrschenden Zwang daran gehindert werden, seine oben genannten Rechte wahrzunehmen (Bandbreitenproblem, wer zuerst kommt, darf zuerst; wer kann überhaupt?).

Argumentationslastregeln

- Wer eine Person A anders als eine Person B behandeln will, ist verpflichtet, dies zu begründen (etwa bei Überlastungssperren, dümmere Sprecher)
- Wer eine Aussage oder Norm, die nicht Gegenstand der Diskussion ist, angreift, muss hierfür einen Grund angeben (es werden Daten und Befehle im Speicher realisiert, parallele Prozesse)
- Wer ein Argument eingeführt hat, ist nur bei einem Gegenargument zu weiteren Argumenten verpflichtet (Statusmeldung; wer will (kann) Gegenargumente drechseln, wer muss?)
- Wer eine Behauptung oder eine Äußerung über seine Einstellungen, Wünsche oder Bedürfnisse in den Diskurs einführt, die nicht als Argument auf eine vorangegangene Äußerung bezogen ist, hat auf Verlangen zu begründen, weshalb er diese Behauptung oder diese Äußerung einführt (Interrupts).

Begründungsregeln

- Jeder muss die Konsequenzen der in einer von ihm behaupteten normativen Aussage vorausgesetzten Regel für die Befriedigung der Interessen einer jeden einzelnen Person auch für den hypothetischen Fall akzeptieren können, dass er sich in der Situation dieser Person befindet (Verbindung durch Middleware)
- Die Konsequenzen jeder Regel für die Befriedigung der Interessen eines jeden einzelnen müssen von allen akzeptiert werden können (Gleichheit der Automaten)

- Jede Regel muss offen und allgemein lehrbar sein (Automaten kennen verschiedene Sprachen)
- Solche moralischen Normen sollen eliminiert werden, die entweder schon seit ihrer Entstehung nicht „rational zu rechtfertigen“ waren oder aber diese „rationale Rechtfertigung“ zwar zunächst besaßen, inzwischen aber verloren haben (nicht compilierbar)
- Die den moralischen Auffassungen der Sprecher zugrunde liegenden moralischen Regeln müssen der Überprüfung ihrer individuellen Entstehungsgeschichte standhalten können. Eine moralische Regel hält einer solchen Überprüfung nicht stand, wenn sie nur aufgrund nicht zu rechtfertigender Sozialisationsbedingungen übernommen wurde
- Die faktisch gegebenen Grenzen der Realisierbarkeit sind einzuhalten (alle Probleme sind Schaltungsprobleme, reale Diskurse).

Übergangsregeln

- Es ist jederzeit jedem Sprecher möglich, in einen theoretischen (empirischen) Diskurs überzugehen
- Es ist jederzeit jedem Sprecher möglich, in einen sprachanalytischen Diskurs überzugehen
- Es ist jederzeit jedem Sprecher möglich, in einen diskurstheoretischen Diskurs überzugehen (es gebe also unendlich viele Diskurse, die gleichzeitig realisierbar sind; die Ebenenhierarchien von z.B. Computerarchitekturen sind dem allgemeineren Operator, nicht den elementaren Bausteinen zugänglich).

4.4 Diskussion, Argumentation, Ausblick

Das Diskussionsmodell nach Sökeland (Stammerjohann, 2009: S.31 ff.)

Zunächst wird die Argumentation dargestellt: Argumentationsregel ist eine statistische Hypothese, die ein Wenn-Dann-Satz mit Prozentwert ist. Diese führt vom Datum als Argument zur These, die mit den Daten zu belegen ist. Das Argument muss zur stat. These passen, damit die These belegt werden kann.

Dagegen besteht die Erklärung aus einem deterministischen Naturgesetz, welches auf eine vorliegende Tatsache angewendet wird, um ihre Ursache anzugeben. Sätze über Tatsachen sind Feststellungen, solche über wahrscheinliche Ereignisse Thesen.
Im Diskussionsmodell geht die Phase der Argumentation durch Akzeptanz in die Erklärung über. Zuerst stellt der Proponent die These auf, der Opponent greift sie an; jener belegt sie mit Daten, dieser argumentiert dagegen. Der Proponent kann andere Argumente mit anderen Regeln liefern; er liefert Gegenargumente der Gegenargumente oder korrigiert seine Argumentation entsprechend den Gegenerklärungen bis zur Akzeptanz.

Argumentationsverfahren nach Lumer (zitiert nach Stammerjohann, 2009: S.13 ff.)
In der logischen Argumentation folgt die These logisch aus den Argumenten. Die Wahrheitsbedingungen ergeben sich aus Ausdrücken, die mit logischen Wörtern verbunden sind. Es interessieren nur die Bedeutungen der logischen Wörter. Logische Wörter sind: der Negator „nicht“, die materiale (logische) Implikation, das Konditional, die Subjunktion „wenn dann“, die materiale (logische) Äquivalenz, das Bikonditional „genau dann, wenn“, die Disjunktion „oder“, die Konjunktion „und“. Nicht logische Wörter sind Variablen wie essen oder trinken. Bei Wörtern und Ausdrücken wird die Existenz und Nichtexistenz betrachtet, sodass sie auf wahr oder falsch gesetzt werden können (Wahrheitswerttabellen). Ein komplexer Ausdruck wird solange analysiert, bis elementare Ausdrücke übrig bleiben, in denen keine logischen Wörter mehr vorkommen. Durch Wahrheitswertevergleich der Teilausdrücke findet man heraus, ob der komplexe Ausdruck wahr, falsch oder kontingent ist. Wenn er wahr oder falsch ist, hat man eine logische Argumentation, ein logisches Gesetz bzw. Schluss.
Dabei haben wir das Schaltungssystem der Prozessoren, dass über eine Wissensbasis oder Input (wissenschaftlicher Art) durch Theorienlöser getaktet wird.

Die Übersetzung und Ausbreitung dieser letzten Abschnitte aus den Geisteswissenschaften in die Sprache der Informatik (das ganze Spektrum an Komplexität betreffend, Orakelautomaten, Wahrscheinlichkeitsalgorithmen, logischen Programmen, MIMD) bzw. der Berechenbarkeit von Problemen, Betriebssystemebenen, Mikroarchitekturen etc. wird als trivial angenommen. Akzeptanz im Rahmen des Diskurses wird als Speichern, Revision durch Korrekturen, Löschung, Modularisierung, Addons implementiert und interpretiert. Teilnehmer

der Diskussion befinden sich auf der selben “Platine” oder sind breiter verteilt; Daten oder Bits werden ausgetauscht, geschoben, geleitet, nach Regel durchoperiert, die digitale Logik getaktet, Effizienzen gemessen etc. Nachrichtenaustausch soll gleich Rede und Gegenrede sein, eine Informationsverwaltung nach Regeln der Diskussionsmodelle möglich sein.

Die Käseglocke des Begründungszusammenhanges, im Sinne einer Software, die den Behälter, der sie enthält, noch mal füllt, sei damit voll. Restliche Zusammenhänge soll die Max Weber Imitation ausgeben oder fingieren. Deren Prüfung kann die Vielzahl mehr oder weniger optimal organisierter Elemente im schlimmsten Fall überfordern, im besten jedoch effizienter machen, sowohl durch die Organisation als auch die Prüfungsleistungen an sich.

5. Verwendete Literatur

Althaus, Ernst (2009): Theoretische Grundlagen der Informatik I. Vorlesungsskript, Mainz. http://www.students.uni-mainz.de/scisnik/tgi1.pdf download am 15.06.2011.

Esser, Hartmut (1993): Soziologie. Campus Verlag, Frankfurt.

Fuhr, Norbert (2003): Rechnerstrukturen und Programmierparadigmen. Vorlesungsskript, Duisburg. http://www.is.informatik.uni-duisburg.de/courses/infoa_ss03/skript/skript.pdf download am 15.06.2011.

Glatz, Eduard (2010): Betriebssysteme. Dpunkt.verlag, Heidelberg.

Glende, Astrid (2003): Rechnerstrukturen. Inoffizielles Skript, Koblenz. http://www.uni-koblenz.de/~palmann/Rechnerstrukturen1.pdf download am 12.06.2011.

Haubelt, Christian & Teich, Jürgen (2010): Digitale Hardware / Software-Systeme. Springer, Heidelberg.

Hilgendorf, Eric (1990): Argumentation in der Jurisprudenz. Dissertation, Tübingen.

Hromkovic, Juraj (2004): Algorithmics for Hard Problems. Springer, Heidelberg.

Mihailovic, Milan (2008): Argumentation: Grundbegriffe, Logik und Anwendungen. Shaker Verlag, Aachen.

Mumdzhiev, Milko (2011): Das Praktikumskonzept von Mumdzhiev Datenanalyse und Statistik. Arbeitspapier, Nürnberg. https://sites.google.com/site/arbeitsgruppesocialeconomics/home/Dokumente download am 20.06.2011.

Patzelt, Werner J. (1986): Sozialwissenschaftliche Forschungslogik. R. Oldenbourg Verlag, München.

Renner, Rolf G. (1991): Denken, das die Welt veränderte. Band 2. Herder, Freiburg.

Schöberle, Wolfgang (1984): Argumentieren-Bewerten-Manipulieren. Julius Groos Verlag, Heidelberg.

Schurz, Gerhard (2006): Einführung in die Wissenschaftstheorie. Wissenschaftliche Buchgesellschaft, Darmstadt.

Socher, Rolf (2008): Theoretische Grundlagen der Informatik. Hanser, München.

Spaniol, O. et al. (2011): Rechnerstrukturen. Vorlesungsskript, Aachen. http://www.nets.rwth-aachen.de/content/teaching/scripts/downloads/rechnerstrukturen.pdf

download am 15.06.2011.

Stammerjohann, Ulrike (2009): Beschreibung, Argumentation, Kontext. Lit Verlag Dr. W. Hopf, Berlin.

Tanenbaum, Andrew S. (2000): Computernetzwerke. Pearson Studium, München.

Tanenbaum, Andrew S. & Goodman, James (2001): Computerarchitektur. Pearson Studium, München.

Wagner, Gerhard (1993): Gesellschaft aus politische Theologie? Duncker & Humblot, Berlin.

6. Anhang

Ein “spontaner” Fragebogenkatalog, der deutschen Soziologen vorgelegt wurde, und diese “außer Gefecht setzte”, ist wiedergegeben.

Max Weber:

1. Wie viele Werte bzw. Wertorientierungen gibt es?
2. Wie viele Zwecke bzw. Zweckorientierungen gibt es?
3. Wie viele Affekte bzw. Affektorientierungen gibt es?
4. Wie viele Traditionen bzw. Traditionorientierungen gibt es?
5. Wie viele Idealtypen gibt es?

Systemtheorie

1. Wie viele Gesellschaften gibt es?
2. Wie viele Gemeinschaften gibt es?
3. Wie viele Verhaltenssysteme gibt es?
4. Wie viele Strukturen gibt es?
5. Wie viele Kulturen gibt es?
6. Wie viele Akteure gibt es?

Sozialstruktur

Soziale Handlungen **aller** Akteure seien auf jeweils 3, 20, 100 Alternativen begrenzt. Nehmen Sie eine Sozialstruktur Deutschlands, und beschreiben Sie diese nach endlich vielen Handlungsschritten der oben gegebenen Akteure (z.B.nach 1000 Schritten).

Beschreiben Sie diese Sozialstruktur nach unendlich vielen Handlungsschritten.

Modellieren Sie obige Situationen, wobei die (zweite) Hälfte aller Akteure nur eine Handlungsmöglichkeit je Handlungssituation haben; daneben wenn alle Akteure nur eine Handlungsmöglichkeit sehen und praktizieren können.

Empirie

Schreiben Sie zur folgenden Frage nicht mehr als 200 Wörter, da dies (z.B. über Strafpunkte) negativ gewertet wird.

Matchen Sie (theoretisch) alle Ihnen bekannten oder über Archive zur Verfügung stehenden Datensätze (und imputieren so gewissermaßen). Tun Sie dieses für die letzten 500 Jahre. Nennen Sie ein Maximum an empirisch prüfbaren Hypothesen, ohne (neue) Datensammlungen zu unternehmen.

Integration

Ein Ehepaar aus der Türkei zieht nach Deutschland.

1. Die Frau wird assimiliert (vollständig integriert), der Mann überhaupt nicht (zu einem Drittel, zu zwei Drittel; gemessen anhand bestimmter Indikatoren).
2. Es kommt zur Scheidung. Formulieren Sie Funktionen, die Szenarien mit und ohne Scheidung durchspielen, wobei Integration optimiert wird (nehmen Sie daneben beliebige Zielfunktionen an) .
3. Ein weiteres Szenario ist durch den vollständigen Gedächtnisverlust des Mannes, und das Sterben der Frau (Unfall bei der Landung) gegeben. Daneben verlieren beide dauerhaft das Gedächtnis (Unfall bei der Landung). Formulieren Sie auch hierfür Modelle und optimieren Integrationseffekte.

Industrie

Nehmen Sie für Landwirtschaft, Industrie- und Dienstleistungsbranchen aktuelle Anteile des BIP an. Simulieren Sie unter der Bedingung, dass Koka, Opium etc. angebaut werden darf und die Produktion und Verwendung digitaler Schaltkreise (mehr oder weniger strikt) eingeschränkt wird, für Deutschland, die EU und Nordamerika (zyklische) Entwicklungen der Wirtschaft, Politik, Recht und Kultur. Beziehen Sie dabei eine Rassismusfrage ein, die den (heutigen) demokratischen, weißen Westen und gescheiterten Osten (Aphganistan, Südamerika etc.) an sogenannten Theorien der Moderne und Postmoderne orthogonalisiert.

Individualisierung

Das Individuum XY kennt weder das Aktiengesetz, diverse Prozessordnungen, das Handelsgesetzbuch etc., noch die meisten Paragraphen des Strafgesetzbuches (in Deutschland). XY hat auch nie gegen diese verstoßen. Zeigen Sie soziologische Theorien, welche diesen Sachverhalt beschreiben oder erklären.

Rassismus

300 Jahre nach einem erfolgreichen Holocaust leben und arbeiten in Deutschland nur „weiße Angehörige der arischen Rasse“. Auf deutschem Gebiet haben existierende Unternehmen, Universitäten, Bürokratie, Zivilgesellschaft etc. weder Zigeuner, noch Juden, Slaven usw. unter Vertrag.

Diskutieren Sie in diesem Zusammenhang die Sentenz: Rassismus ist tot, wir haben ihn begraben.

Sozialforschung

Die Anzahl der (natürlichen) Sozialforscher (an Universitäten, Instituten mit gemeinnützigen Zwecken u.ä.) wird halbiert, ihre Etats an die überbleibende Hälfte ausgeschüttet. Diskutieren Sie den weiteren Verlauf der Sozialforschung (Theorie und Praxis derselben); diesen für unterschiedlich begründete Selektionen.