Mathematische Logik

Zusammenfassung 2000 / 2001 von Klaus Ridder und Matthias Hensler

1. AUSSAGENLOGIK

1.1. Syntax und Semantik der Aussagenlogik

Aussagenvariablen: 0/1, $\tau = \{x0, x1, x2, ...\}$

$\wedge$ AND
$\vee$ OR
$\neg$ NOT
$\rightarrow$ Folge, impliziert

Formeln: (AL = aussagenlogische Formel)
(1) 0,1 $\in$ AL
(2) $\tau \in$ AL
(3) $\psi, \varphi \in$ AL $\rightarrow$ $(\psi\wedge\varphi), (\psi\vee\varphi), (\neg\psi), (\psi\rightarrow\varphi) \in$ AL.

Semantik:
y = Formel, z.B. **(X1** Ù **X2)**
t**(**y**)** = alle in ψ enthaltenen Variablen, hier **X1** und **X2**.
s = Menge von Aussagenvariablen {X1,X2,X3}

[Formel]^J = **Interpretation** einer Formel = Variabllenbelegung.
(kann man dann nach 0 oder 1 auflösen).
Interpretation „**passend zu** y“ $\rightarrow$ ψ **ergibt 1 oder 0** bei Interpretation.

[y $\rightarrow$ j **]** º **[**Øy Ú j **]**: nur falsch bei „1-0“.

Konzidenzlemma (klar): gleiche Interpretation[1] $\rightarrow$ gleiche Auflösung (0/1)

Modell = Interpretation, die Formel wahr macht.
Formel **erfüllbar** = Modell existiert.
Tautologie = Formel **immer wahr**, egal welche Interpretation.
<u>Lemma</u>: ψ erfüllbar $\leftrightarrow \neg\psi$ keine Tautologie (d.h. ψ nicht immer 0)

Algorithmus für Erfüllbarkeit: alle Variablen durchprobieren $\rightarrow$ $\mathbf{2^n}$.

y º j : **Formeln äquivalent** = alle möglichen Interpretationen liefern gleichen Wahrheitswert.

[1] d.h. alle Variablen haben in beiden Interpretationen den gleichen Wert

Äquivalenzen:

(2) DeMorgan: $\neg(\psi\wedge\varphi) \equiv \neg\psi\vee \neg\varphi$
$\neg(\psi\vee\varphi) \equiv \neg\psi\wedge \neg\varphi$

(3) Distributiv: oder / und ausklammerbar.

1.2. Aussagenlogik, Boolsche Funktionen

Boolsche Funktion: kriegt Werte (0/1) → 0/1.

z.B..: n=2, B^2 : (B^n)

f_{0001} (x,y) = $x \wedge y$:
0001 stellt hier die 4 Möglichkeiten bei einer boolschen Funktion dar:
00, 01, 10, 11. Die Formel $x \wedge y$ ist nur wahr, wenn Fall 4 eintritt. Daher steht hier: 0001.

ANZAHL DER FUNKTIONEN:

Es gibt bei **n** Eingabevariablen $\mathbf{2^n}$ mögliche Eingaben. Jede dieser Möglichkeiten auf eine bestimme Ausgabe gibt eine Funktion:
- alle Möglichkeiten ergeben 0
- nur die letzte ergibt 1
- ...
- aller ergeben 1:

Das sind wieder $\mathbf{2^{Mögliche\_Eingaben}}$ = **2^2^n** boolsche Funktionen.

z.B.: 1 Eingabe = 2 Möglichkeiten, den einen Platz zu besetzen (0/1) = 4 Funktionen
3 Eingaben = 2^3 Möglichkeiten, die 3 Plätze zu besetzen = 2^2^3 Funktionen.

Boolsche Funktion vorhanden « AL Funktion vorhanden.

z.B.: 00→0
01→0
10→0
11→1 → AL: $\boxed{A \wedge B}$

Boolsche Fkt. → AL: Herleitbar wie in Rechnerstrukturen: DNF! ☺

1.2.1. Disjunktive / Konjunktive Normalform:

Diskunktion = ODER = $\vee$
Konjunktoin = UND = $\wedge$

DNF: $(A\wedge A\wedge A) \vee (A\wedge A\wedge A)$: dickes ODER in der Mitte.
KNF: $(A\vee A\vee A) \wedge (A\vee A\vee A)$: dickes UND.

zu jeder Formel ist eine DNF vorhanden! DNF → KNF möglich!

1.2.2. Def. 1.10 Funktional vollständige Mengen:

Funktional vollständige Menge = aus diesen Funktionen lassen sich alle anderen Funktionen herleiten.

Funktional vollständig sind z.B.: $\{\wedge,\neg\}$, $\{\vee,\neg\}$, $\{\rightarrow,\neg\}$, $\{\rightarrow,0\}$
(eingekreistes + = XOR)

1.3. Horn-Formeln:

ψ = eine **KNF**, wobei in jeder Klammer **maximal 1 Literal ohne** Ø dasteht:

z.B.: (¬A ∨ ¬B ∨ **C**) ∧ (¬D ∨ ¬E ∨ **F**)

es gilt: **(ØA Ú ØB Ú C) ° (A Ù B → C):** beides nur 0, wenn C=0 und Rest=1.
zuerst immer alle Formelteile nach dieser Regel umformen → !

Beispiel:
y **= (1→A)** Ù **(A** Ù **B → C)** Ù **(A → B)** Ù **(B** Ù **C → 1)**

ERFÜLLBAR?
wir haben ja UND-Verknüpfungen.
Um die erste Klammer wahr zu machen, muss (1→A) muss wahr sein. (1→A = A).
Aus A=1 folgt dritteKlammer=B → B=1,
daraus zweiteKmlammer = C, C=1,
vierte Klamme: 1→0 ≡ 0, also Horn-Formel nicht erfüllbar.

Bem.: dieser Algorithmus findet immer das kleinste Modell.
Kleinstes Modell = ein Teil der Variablen kriegt feste Werte, der Rest darf beliebig sein.
Beachte: **erfüllbar** ↔ hat **kleinstes Modell**.

1.4. Der Kompaktheitssatz der Aussagenlogik

F = Formelmenge. **Modell J** = macht <u>alle</u> Formeln wahr.

aus F folgt y: jedes Modell für F ist auch Modell für y. („semantische Folgebeziehung")

$\boxed{\text{F } |= \text{ y .}}$

<u>SATZ:</u>

(i) F **erfüllbar** « **jede endliche Teilmenge** F_0 **von** F **erfüllbar.**

„→" klar: wenn alle Formeln erfüllbar sind, ist es auch ein Teil davon.
„←": zu schwer.

(ii) F |= y « **es ex. eine endliche Teilmenge** F_0 Í F **mit** F_0 |= y.

„←" klar: geg.: Jedes Modell, das alle Formeln aus F_0 erfüllt, erfüllt auch ψ.
Die Modelle, die ALLE Formeln aus Φ erfüllen (nicht nur die aus Φ_0), sind nun natürlich nur ein Teil davon. Natürlich erfüllen die dann auch noch ψ (sind ja dieselben).

„→": setze (i) als bewiesen voraus. - Widerspruchsbeweis:
...kann für jede Teilmenge Φ0 aus Φ eine Interpretation finden, die Φ0 erfüllt, aber nicht Φ.
daraus folgt: Φ0 erfüllbar, ψ nicht erfüllbar → $\Phi0 \cup \neg\psi$ immer erfüllbar.
aus (i) folgt: → ganz $\Phi \cup \neg\psi$ auch erfüllbar. → $\Phi \models \neg\psi$ → Widerpspruch.

1.4.1. Anwendung: das Lemma von König.

Baum mit endlichen Wegen → unendlicher Weg vorhanden.

1.5. Aussagenlogische Resolution

KNF → Die Werte einer jeder Klammer in je eine Menge = 1 Klausel.
Doppelte fallen dann weg.

Beweis von Unerfüllbarkeit:

1.) Alle Klauseln hinschreiben: kommt in 2 Klauseln ein Ausdruck einmal negiert und einmal nicht negiert vor, so kann man ihn „resolvieren", d.h. einfach alle Ausdrücke mit ODER verknüpfen.

$(<.......> \vee A) \wedge (<.......> \vee \neg A) \quad \rightarrow \quad <.....> \vee <....>.$

Um dies zu erfüllen, muss ja entweder ein Argument aus der linken <....> oder eins aus der rechten <....> erfüllt sein. Also können wir einfach beide mit ODER verknüpfen

Wir resolvieren jetzt so lange wie wir können. Kommt irgendwo eine leere Klausel raus (d.h. einer der Wertemengen = 0), so ist die Formel unerfüllbar, sonst ist sie erfüllbar.
<u>BEISPIEL:</u> siehe S. 17 oben.

BEACHTE: für undendliche Klauselmengen kann es passieren, dass das nicht funktioniert.

Für **Horn-Formeln** (Wir betrachten hier nur die „$\vee$“ – Form, nicht die „$\rightarrow$“ Form):
(Test muss das ganze in n Schritten abbrechen.)

Die Horn-Formel kann nur dann nicht erfüllbar sein, wenn mindestens eine Klausel (d.h. eine Klammer) mit nur einem Literal da ist.
Wenn wir jetzt den Resolutionstest nur mit „Einheitsresolution“ anwenden, d.h. bei jedem Resolutionsschritt MUSS eine Klausel mit genau 1 Element beteiligt sein, und damit eine leere Klausel erhalten, so ist die Formel nicht erfüllbar. Sonst ist sie erfüllbar.

Beweis: wie der Erfüllbarkeitstest.

1.6. Der aussagenlogische Sequenzkalkül

Sequenz: G Þ D (das sind endliche Formelmengen).
G = „Antezedens“
D = „Sukzedens“

Diese Sequenz ist gültig, wenn jedes Modell von Γ mindestens 1 Formel aus Δ wahr macht.
d.h.: Interpretation macht alle Formeln aus Γ wahr $\Rightarrow$ diese Int. macht auch mind. eine Formel aus Δ wahr.

Wenn es also eine Interpretation gibt, die alle Formeln aus Γ wahr macht und alle aus Γ falsch, ist das nicht gültig. Diese Interpretation „falsifiziert“ dann diese Sequenz.

Dies kann man in sogenannten „Schussregeln“ verwenden:
diese Schlussregeln kann man sich einmal herleiten und dann benutzen. Man liest sie von unten nach oben:

Wenn man irgendeine Formel hat, und wissen möchte, ob sie erfüllbar ist, sucht man irgendeine Schlussregel, bei der der Teil unter dem Strich auf die Formel passt.

Beispiel: **X, XÚY Þ XÙY, Y**.
das bedeutet: links stehen 2 Formeln, rechts eine. Es muss ja JEDE Interpretation, die alle Formeln links wahr macht, mindestens eine rechts wahr machen.
dazu können wir jetzt die Vorlagen „$\vee\Rightarrow$“ oder „$\Rightarrow\wedge$“ nehmen. Wir nehmen oBdA mal „$\vee\Rightarrow$“
Wir haben unter dem Strich stehen: $\Gamma, \psi \vee \vartheta \Rightarrow \Delta$.
Dies bedeutet: Γ ist eine beliebige Formelmenge, $\psi \vee \vartheta$ unsere „spezielle“ Formel.
Rechts steht nur eine Formelmenge, nämlich Δ.
(beachte: G, D = Formelmengen, y , J = Literale).
X, XÚY Þ XÙY, Y

Nach der Schlussregel gilt ja: G, y $\Rightarrow$ D   G, J $\Rightarrow$ D., hier also
X,**X** $\Rightarrow$ **XÙY,Y**   **X**,**Y** $\Rightarrow$ **XÙY,Y**.
WARUM ? – Nun, was oben und unten steht, muss ja äquivalent sein. Wir gehen bei „$\vee\Rightarrow$“ ja davon aus, dass links eine Formelmenge steht, inklusive unserer „betrachteten“ Formel, und rechts auch eine Formelmenge. Das ganze ist ja nur bei $1\Rightarrow0$ falsch, und genau diesen Fall müssen wir bertrachten: das ganze ist hier also – ausgehend davon, dass Γ=1 und Δ=0 ist (sonst ist es eh egal da immer richtig) – genau dann falsch, wenn $X\vee Y$ wahr ist.
Dies müssen wir jetzt ohne Junktoren ($\vee$, $\wedge$, $\neg$) schreiben. Dann sieht das so aus:

$\Gamma, \psi \Rightarrow \Delta \quad \Gamma, \vartheta \Rightarrow \Delta$
WIESO? → Ganz einfach: Das hier (d.h. also eins davon) wird genau dann falsch (wieder unter der Voraussetzung, dass Γ=1 und Δ=0 ist), wenn ψ=1 oder ϑ=1 ist. Und genau das wollten wir ja haben ☺

Am Schluss haben wir nur noch Blätter, in denen nur noch Axiome sind. So ein Axiom ist falsch, wenn links und rechts disjunkt sind; dann können wir nämlich einfach alles links mit 1 bewerten, und alles rechts mit 0 → $1 \Rightarrow 0$ → falsch. Es reicht, wenn ein solch ein Blatt falsch ist, dann ist alles falsch.
→ FERTIG.

2. Strukturen und Homomorphismen

Strukturen aller Art (Algebraische, geometrische, Räume, Datenstrukturen)
Beziehungen zwischen Strukturen
Abbildungen zwischen Strukturen (**Strukturehaltend**)
Kongruenzrelation
Quotientenstrukturen

2.1. Einige Grundbegriffe

Kartesisches Produkt:
$A \times B$: Paar; eins aus A, eins aus B.
A^n: n Elemente aus A, sortiert !

Relationen und Funktionen
n-stellige Relation = Teilmenge von A^n.
Aussagen = Nullstellige Relationen: $\emptyset$ = false, { } = true.

n-stellige Funktion aus A: A^n → A: Tupel aus A → ein A.
f{*f*} → A = Konstante (wenn man nur das leere Tupe hat, kommt ja auch imemr dasselbe raus).

Graph: Jedem Tupel wird ein Wert zugeordnet.

injektiv = alle Funktionswerte verschieden,
durjektiv = ganzer Funktionsbereich abgedeckt,
bijektiv = inj + sur.

gleichmächtig = gleich viele Elemente = bijektive Abbildung existiert.
abzählbar = es gibt f:N → A.

Lemma 2.1: Menge abzählbar → endlich oder gleichmächtig zu N.

Potenzmenge = alle Teilmengen von A.

Satz 2.2: Keine Menge ist gleichmächtig zu ihrer Potenzmenge. (völlig klar).

2.2. Strukturen

Struktur = Universum (1,2,3,4), Fuinktionen (+, *), Relationen (<, >)

Beispiele: Körper, Graphen additive Gruppe N.

Signatur = **Symbolmenge** = **Vokabular**: Menge von Namen (bzw. Symbole) der Funktionen und Relationen
R^n = nur n-stellige Relationssymbole.

funktionale, algebraische Signatur = Signatur hat nur Funktionen, keine Relationen.
relationale Signatur = Signatur hat nur Relationen, keine Funktionen.

P,Q,R: Relationen
f,g,h : Funktionen
c,d,e: Konstanten
s, t**:** Signaturen
$\mathbf{R}^{\mathbf{gothisch\_A}}$ = konkrete Relationn
$\mathbf{f}^{\mathbf{gothisch\_A}}$ = konkrete Funktion.

Def. 2.4**:** t**-Sturktur**: Universum + Relationen + Funktionen.
t**-Algebra** = τ - Struktur ohne Relationen.

Struktur: **gothisch-A**.
Universum: **A.**

Def. 2.5: **Substruktur**, **Erweiteung**: weniger / mehr Symbole:
2n ist Substruktur von n. 2n+1 istr nicht Substruktur von n, da nicht {+}-abgeschlossen.

Def. 2.6: **Redukt**, **Expansion**: Struktur mit weniger / mehr Relationen bzw. Funktionen.

2.3. Ein Zoo von Strukturen

Beispiele.

2.4. Mehrsortige Strukturen

2.5. Homomorphismen und Isomorphismen

Homomorphismus = lineare Abbildung = strukturerhaltende Abbildung.

Def. 2.8: Im **Homomorphismus** gilt:

(1) Die Elemente sind nach der Abbildung immer noch in Relation (d.h. sind geordnet)
(2) Man kann entweder zuerst abbilden und die Funktion anwenden, oder andersherum.

Def. 2.9**: starker Homomorphismus** = Umkehrabbildung gilt auch.
Def. 2.10: **Isomorphismus** = bijektiver starker Homomorphismus.

Automorphismus = Isomorphismus auf sich selbst.

NOTATIONEN:
Pi ist Isomorphismus = A Pfeil mit Tilde B
Pi ist Einbettung = A geschwungener Pfeil B

Symmetriegruppe = Autotmrphismen einer Struktur = Gruppe mit neutralem Element
starrer Automotphismus = Id().

3. Syntax und Semantik der Prädikatenlogik

Prädikatenlogik ist ausdrucksstärker als die Aussagenlogik.

3.1. Syntax der Prädikatenlogik

Signatur τ: aus dem Alphabet τ machen wir Formeln, Terme, ...

zusätzlich zu: Variablen, Junktoren und Klammern kommen jetzt:
- Relations- und Funktionssymbole
- =
- ( )

Def. 3.1: Terme.
(1) jeder Variable ist ein τ-Term (Grundterme)
(2) f(τ) ist auch ein τ-Term.

Term $\mathbf{t(x_1,...,x_n)}$ = Term mit genau den versch. Variablen x_1 bis x_n.

<u>Beispiel</u>: Signatur mit den Symbolen:
f für Funktion mit 1 Argument,
g für Funktion mit 2 Argumenten
c für Konstante (d.h. Funktion ohne Argument)

ggccfx = g(g(c,c),f(x)), auch als Baum darstellbar.

Def. 3.2: Aufbau von **Formeln**:
(1) **t1 = t2** ist Formel.
(2) **Relationssymbol** und dahinter Terme ist eine τ-Formel.
(3) Ø erlaubt.
(4) Ù, Ú erlaubt
(5) y τ-Formel, → $ **x** y, " **x** y sind auch Formeln.

atomare Formel = nur „=“ und Relationssymbole (also keine Junktoren und Quantoren):
quantorenfreie Formeln = ohne $\exists$ und $\forall$.

Meistens benutzen wir ab jetzt die INFIX - Notation (a+b statt +ab).

S. 44

<u>Freie und gebundene Variablen</u>
gebunden = kommt in $\exists$ oder $\forall$ vor (können wir ja nicht frei belegen).
Wenn nun da steht: $\forall$ x(......x......), (lies: für alle x gilt ()),
dann ist auch das x in der Klammer gebunden.
(Nicht aber die x, die nicht im Bereich eines Quantors sind!)

Def. 3.3:
t ist Term, ψ ist Formel. Var(t), Var(ψ) = alle dort auftretenden Variablen.
Frei(ψ) ist: (alles logisch)

(1) für atomare Formel ist Frei(ψ) = Var(ψ).
(2) Frei(¬ψ) = Frei(ψ)
(3) Frei(ψoφ) = Frei(ψ) ∪ Frei(φ) (o = ∧, ∨, →):
Verknüpft man 2 Formeln durch ∧, ∨, →, so kann man die freien Variabeln einfach zusammennehmen.
(4) Die Freien Variablen in ψ in dem Ausdruck Frei(∀xψ) sind alle freien Variablen in ψ ohne alle x. (dasselbe gilt für Frei(∃xψ))

$\psi(x_1,..., x_k)$: maximal $x_1,..., x_k$ kommt in der Formel ψ vor.

Mächtigkeit: abzählbar sind:

- Signatur τ,
- Alphabet Alph(τ),
- Term T(τ),
- Formelmenge FO(τ)

FO und T sind unendlich.

3.2. Semantik der Prädikatenlogik

Die Modellbeziehung

Def. 3.4:
τ - Interpretation: Paar: **Struktur** + **Variablenbegelgung** dazu.
(z.B.: x:=irgendeine_Zahl, nicht nur x:=TRUE/FALSE!)

S. 44, S.45

<u>BEISPIEL:</u> wir geben jeder Formel und jeder freien Variablen eine Bedeutung:
t = fxgy = f(x,g(y)). (also $a=(N, f^a, g^a)$) (a ist die Signatur?)
Bewertung der **Formeln**: f:=+; g:= $n \rightarrow n^2$
Bewertung der **Variablen**: x→3, y→2, und wir können ausrechnen:
$f(x,g(y)) = f(3,g(2)) = f(3,2^2) = 3+2^2 = 3+4 = 7$.

SIEHE SKRIPT S. 45:
Interpretation von Termen : (intuitiv).
Interpretation von Formeln:
(3): kann bei = nur 0 oder 1 geben. Ist 1, wenn die Interpretation beider Seiten gleich ist.
(4): Relation: 1, wenn alle Elemente in der Relation sind.
(5) – (8): intuituiv, genau wie in der Aussagenlogik.

Blind mitgetippt:
ψ=∃ z (E xy ∧ E zy), φ = ∀ x ∀ y (E xy → ψ)

$J=(a,b)$ $a=(N, E^a)$ $E^a = \{ (m,n)\text{: } m|n \text{ (“m teilt n”')}, m \neq n \}$
ß: x → 2, y → 36 $[[\psi]]^J = 1$

Satz = Formel ohne freie Variablen.

Def 3.6: sei Φ eine Menge von τ-Sätzen (Formeln ohne freie Variablen)
Die Modelklasse von Φ, kurz Mod(Φ), ist die Klasse aller τ-Strukturen a, so dass a $\models \Phi$.

Beispiel 1: ungerichtete Graphen
$\Phi_{Graph} = \forall x \neg\exists xx, \forall x \forall y (Exy \rightarrow Eyx \}$
es gibt keine Kante von x nach x. Aber wenn eine von x nach y geht, geht auch eine von y nach x.

Beispiel 2: Gruppe:
$G=(G, o, e, {}^{-1})$

$\Phi_{Gruppe} = \{ \forall x\forall y\forall z ((x*y)*z = x*(y*z)),- \ldots.$
also die bekannten Gesetze einer Gruppe.

Def. 3.7: Semantische Folgerungsbeziehung

$\Phi \models \psi$ bedeutet: Wenn eine Interpretation ALLE Formeln links wahr macht, muss die auch die Formel rechts wahr machen Beispiel:
$\Phi_{Gruppe} \models x^{-1}$ o $x = 1$ bedeutet: egal was man für x einsetzt, man bekommt IMMER die 1.

$(N, E^a) \models \psi(2, 36)$ ist wahr: 2 ist Teiler von 36.
$(N, E^a) \models \varphi$?
$(R, K) \models \varphi \rightarrow$ wahr.

Def. 3.8:
Formel ψ / Formelmenge Φ **erfüllbar** = $\exists$ Modell (= Interpretation, die Formel wahr macht.)
Allgemeingültig: „$\models$ y" : jede Interpretation macht Formel wahr.
Logisch Äquivalent: „$\psi \equiv \Phi$" (Input-Output-Verhalten links und rechts ist gleich.)

Def. 3.9:
Existentieller Abschluss: $\exists x_1, ..., x_k\ \psi$ es existiert x_1 bis x_k , so dass ψ gilt.
Universeller Abschluss: $\forall x_1, ..., x_k\ \psi$: für alle x_1 bis x_k gilt ψ.
x_1 bis x_k sind alle freien Variablen.

Lemma 3.10:
erfüllbar $\Leftrightarrow$ existentieller Abschluss ist erfüllbar. (logisch.)
allgemeingültig $\Leftrightarrow$ universeller Abschluss gilt.

3.3. Substitutionen
t[x/s]: alle x durch s ersetzen. (s kann auch ganzer Ausdruck sein).
(x darf auch in s vorkommen).

ACHTUNG: NIEMALS eine gebundene Variable ersetzen !!!!!!! N I E M A L S !!!!!
wäre ja auch Quatsch.

Def .3.11:
nur endliche viele Variablen bleiben frei, der Rest wird festgesetzt.
Dies ist eine Funktion. Man muss ggf. gebundene Variablen „umbenennen“, um Kollisionen zu vermeiden.

Notation:
vereinfachte Schreibweise: statt in ψ[x1/t1,...] schreiben wir: ψ(t1, ...), wenn es klar ist, dass alle x... durch t... ersetzt werden.

Lemma 3.12 (Substitutionslemma)
Für Term, Formel, Substitution, Interpretation (t, ψ, ρ, $_{\text{geschwungen}}$I) gilt:

(i) egal, ob « Substitution → Interpretation » oder « Substitution → Interpretation ».
(ii) ??

also irgendwie heißt das: beim Substituieren macht man nie was kaputt.

3.4. Normalformen:

(in etwa):
Normalform = Äquivalenzrelation von Formel, die alle dasselbe tun.

Lemma 3.13 (Ersetzungslemma):
Da die Formeln der Menge alle dasselbe tun, können wir sie wild durcheinander austauschen, und es gilt immer noch alles (¬, ∧,∨, →, ∃, ∀, und ersetzen kann man damit).

Reduzierte Formeln: (logisch) (wir hören gerade.Nirvana El Bosco – geniales Lied! ☺)
wir können folegnde Symbole rausschmeißen und durch andere ersetzen:

∧, →, ↔, ∀ durch folgende Formeln:

A und B = Nicht (Nicht A oder Nicht B)
A folgt B = Nicht A oder B
Für alle A folgt B = Ex existiert kein x so, dass B nicht gilt.

Lemma 3.14: („Lemma p**“)** **Mrew!** ☺
klar: man kann zu jeder Formel eine reduzierte konstruieren.

Def. 3.15: Negationsnormalform (NNF): nur Ø**,** Ù**,** Ú**,** $**,** "**.** (auch kein → !)

Satz 3.16: man kann jede Formel in NNF darstellen.

Termreduzierte Formeln: keine Tiefe ≥ 2.

Lemma 3.17: man kann jede Formel termreduziert darstellen. (durch Substitution).
Beispiel: f(g(h(x))) | h(x) = y, g(y)=z → f(z).

Die Pränex-Normalform

Lemma 3.18:
(i)
Bei ∃ kann man ∨ ausklammern: Es ex. ein x so dass A und (es ex. ein x so dass) B gilt.
Bei ∀ kann man ∧ ausklammern: Für alle x gilt A und (für alle x gilt) B

(ii) wenn man eine Aussage hat und noch eine „ex existiert“-Aussage, kann man das „es existiert“ auch ganz nach außen packen:

$A \vee \exists xB \equiv \exists\, x\,(A \vee B)$:

klar: wenn A gilt, kann x ja sein wozu es gerade Lust hat, ist ja egal. x ist nur relevant, wenn es um B geht.(logisch, wenn man mal drüber nachdenkt.)

$A \wedge \exists xB \equiv \exists\, x\,(A \wedge B)$:
$A \vee \forall xB \equiv \forall\, x\,(A \vee B)$:
$A \wedge \forall xB \equiv \forall\, x\,(A \wedge B)$:

(iii) wenn kein x esistiert, für das A gilt, gilt für alle x „nicht A“.
$\neg\exists xA \equiv \forall x\neg A$.
$\neg\forall xA \equiv \exists x\neg A$.

(iv) $\exists x\exists y$ und $\exists y\exists x$ ist logischerweise dasselbe.

bereinigte Formel:
- keine Variable gebunden UND frei vorkommt.
- jede Variable wird *maximal* einmal quantifiziert (∀, ∃)

Def. 3.19: Pränex-Normalform:
zuerst Quantonen (∃ und ∀), dann eine Formel ohne Quantoren. und bereinigt (s.o.)
also: $\forall x\forall y\exists z(A\wedge B\wedge C\vee D\wedge\neg E)$

Satz 3.20 (Satz über Pränex-Normalform):
Jede Formel lässt sich in Pränex-Normalform darstellen.
Die Umwandlung funktioniert folgendermaßen:

$\psi \equiv \neg\forall x\neg Rxx\wedge\forall x\exists y(Rxy\wedge\exists x(\neg Ryy\wedge Ryx))$ | 1. Schritt : $\neg\forall x\neg Rxx \rightarrow \exists xRxx$
$\psi \equiv \exists x\ Rxx\wedge\forall x\exists y(Rxy\wedge\exists x(\neg Ryy\wedge Ryx))$ | 2. Schritt : $A \wedge \exists x(B) \rightarrow \exists z(A\wedge B)$.
$\psi \equiv \exists x\ Rxx\wedge\forall x\exists y\exists z(Rxy\wedge(\neg Ryy\wedge Ryz))$
$\psi \equiv \exists u\ Ruu\wedge\forall x\exists y\exists z(Rxy\wedge(\neg Ryy\wedge Ryz))$ | 3. Schritt : rot auch wieder nach vorne.
$\psi \equiv \exists u\forall x\exists y\exists z\ Ruu\wedge (Rxy\wedge\neg Ryy\wedge Ryz)$ → fertig. :-)

Satz 3.21: (Satz über die Skolem-Normalform):
Wie Pränex, aber vorn nur ∀ (kein ∃).
Umwandlung der Formel, so dass gilt:
(i) diese neue Formel hat die Pränex-Form, also z.B. $\forall x\forall y\exists z(A\wedge B\wedge C\vee D\wedge\neg E)$
(ii) Die gleichen Variablen sind frei.
(iii) $\varphi \models \psi$: jede Interpretation. die die neue Formel wahr macht, macht auch die alte wahr. (soll also die gleiche Formel wie vorher sein).
(iv) Zu jedem Modell von ψ existiert eine Expansion, welche Modell von φ ist. (häää?)

3.5. Spieltheoretische Semantik

Auswertungsspiel: MC($_{\text{geschwungen-}}$I, ψ)
Verifiziererin: V, Falsifizierer: F.

Wie lösen während des Spiels die Formel weiter auf, und bereits interpretierte Variablen fallen raus.

Bei einem „A und B" prüft der Valsifizierer eins davon, es reicht ja, wenn eins falsch ist.
Bei „A oder B" sucht die Verifiziererin sich eins aus; es reicht ja, wenn eins richtig ist.

Bei $\exists$ muss sich die Verifiziererin eins aussuchen (reicht ja, wenn EINS existiert.)
Bei $\forall$ muss der Falsifizierer eins aussuchen (reicht ja, wenn EINS falsch ist.)

4. MALO – 4. Kapitel

- Fortsetzung -

4.1. CTL

E=$\exists$, A=$\forall$, X = next, U = until.

EXΨ = am einem neXten Punkt gilt Ψ.
AXΨ = an allen neXten Punken gilt Ψ.
E(φUΨ) („until") = es gibt einen Pfad, auf dem irgendwann Ψ gilt, und vorher φ.
A(φUΨ) („until") = auf allen Pfaden gilt irgendwann Ψ, und vorher φ.
FΨ = irgendwann auf dem Pfad gilt mal Ψ.
GΨ = Überall auf dem Pfad gilt Ψ.

Bsp. 1: EFΨ = Es existiert ein Pfad, auf dem irgendwann Ψ gilt. (Ψ ist erreichbar)
Bsp. 2: AG $\neg(P \wedge Q)$ = an allen erreichbaren Zuständen schließen sich P und Q aus.
Bsp. 3: AFΨ = auf allen Pfaden erreiche ich irgendwann Ψ
Bsp. 4: AGAFΨ: auf allen Pfaden gilt unendlich oft Ψ.

Eigenschaften von CTL:
Das Model Checking Problem für CTL ist effizient lösbar.

CTL ist invariant unter Bisimulation (Übung)
CTL hat die EME
ML <-> FO^2

```
H                      He
Li Be B  C  N  O  F  Ne
Na Mg Al Si P  S  Cl Ar
```

5. 5. Definierbarkeit der Prädikatenlogik

5.1. 5.1. Definierbarkeit

Def.: Str(τ): Klasse aller τ - Strukturen

K ist FO-axiomatisierbar, wenn eine Menge FI von FO-Sätzen dieser Signatur existiert, so dass diese Klasse K gerarde die Menge derjenoigen Struktureder Signatur Tao ist, welche Modell sind von PSI.
Wenn es also ein solches Modell gibt, dann ist die Klasse FO axiomatisierbar.
Wenn FI endlich ist, dann können wir K durch einen einzigen Satz axiomatisieren, nämlich $\Psi = \bigwedge \{\varphi : \varphi$ elem PSI$\}$ → K endlich axiomatisierbar.

14:04 30.01.01

Satz von Ehrenfreucht und Fraisée:

τ endlich, relational; *A*,*B* τ-Strukturen