10.06.2021

Ein neuer Ansatz für die Mengenlehre, Version 1.5

von Thomas Limberg

Zusammenfassung

Wir führen einen simplen Logikkalkül namens Limberg-Kalkül sowie zwei daran angefügte Herleitungssequenzen ein. Die Herleitungssequenzen enthalten dabei einen Beweis, dass Leer- und All-/Universalmenge (unter dem Limberg-Kalkül) existieren.

Informationen zum Autor

Name: Thomas Limberg
Land: Germany
Geburtsdatum: 27.11.1986
eMail-Adresse: Nurino@gmx-topmail.de

Einordnung gemäß „Mathematics Subject Classification 2010“

03-XX	Mathematical logic and foundations
03Bxx	General logic
03B10	Classical first-order logic
03Exx	Set theory
03E20	Other classical set theory (including functions, relations and set algebra)

Abkürzungen

$\in$	ist Element von
=	ist gleich
<	ist kleiner
>	ist größer
g.d.w.	genau dann, wenn
H.	Herleitung

HS.	Herleitungssequenz
S.	Satz
q.e.d.	quod erat demonstrandum (lat. für „was zu beweisen war“)
vkm.	verkettet mit

§1 Aussagesätze, Verum und Falsum

S.1: Aussagesätze müssen bzgl. **Rechtschreibung** und **Grammatik** korrekt sein.
S.2: Aussagesätze müssen **definiert** sein.
S.3: Z.B. sind die Aussagesätze "3 = 3." und "3 = 4." definiert, die Aussagesätze "3 $\in$ 4." und "3 + 2i < 4 + i." (mit "i" als imaginärer Einheit) hingegen nicht.
S.4: Sind die Aussagesätze A und B definiert, so sind auch Aussagesätze, wie z.B. nicht A, nicht B, A und B, A oder B, A oder nicht B, A impliziert B, A ist äquivalent zu B, definiert.
S.5: In der Mathematik müssen Aussagesätze zudem **eindeutig** sein.
S.6: Aussagesätzen muss auch sonst **eindeutig ein Wahrheitswert zugeordnet** werden können (, so dass z.B. das Lügner-Paradoxon nicht zum Tragen kommt).
S.7: Und schließlich müssen Aussagesätze **(semantisch) korrekt** sein.
S.8: Es gibt zwei Wörter, **Verum** (wahr) und **Falsum** (falsch), von denen Verum aufgestellt und Falsum nicht aufgestellt werden darf.

§2 Definitionen

S.1: Eine **Definition** ist die Angabe der Bedeutung einer Entität, also z.B. eines Wortes.
S.2: Es können 2 Fälle unterschieden werden.
S.3: Besitzt eine Entität bereits eine Bedeutung, so kann diese womöglich beschrieben werden.
S.4: In diesem Fall sprechen wir von einer **feststellenden/deskriptiven Definition**.
S.5: Wird hingegen einer Entität ohne Bedeutung eine Bedeutung zugewiesen, so sprechen wir von einer **festlegenden/festsetzenden Definition**.
S.6: Wegen §1.S.7 gilt folgendes: Eine festlegende Definition darf Falsum nicht enthalten und durch die Einführung einer festlegenden Definition darf es auch nicht zur Ableitbarkeit von Falsum kommen.
S.7: Ein **Widerspruch** ist ein Beispiel für einen Satz, aus dem Falsum abgeleitet werden kann.
S.8: Zudem seien **vollständige und unvollständige Definitionen** unterschieden!
S.9: Eine vollständige Definition ist z.B. "x = 5." oder "M ist Menge und 3 ist Element von M und sonst ist nichts Element von M.".
S.10: Eine unvollständige Definition ist z.B. "x > 5." oder "M ist Menge und 3 ist Element von M.".

§3 Ableitungen

S.1: Beim Ableiten von Aussagen seien nur **endlich viele Ableitungsschritte** zulässig!
S.2: Dies ist wichtig, damit rekursive Definitionen terminieren.

§4 Entitäten

S.1: Alles ist eine Entität.
S.2: S.1 ist eine vollständige feststellende Definition des Wortes "Entität".
S.3: "Entität" ist somit der oberste Oberbegriff.

§5 Klammern

S.1: Der Text nach einer öffnenden und vor der dazugehörigen schließenden Klammer sei Klammerinneres genannt!
S.2: Der Text vor einer öffnenden und nach der dazugehörigen schließenden Klammer sei Klammeräußeres genannt!
S.3: Für Anmerkungen in Sätzen seien im Folgenden, entgegen der üblichen Konvention, eckige Klammern anstatt runde Klammern verwendet!
S.4: Runde Klammern seien im Folgenden stattdessen zur Strukturierung von Sätzen verwendet!
S.5: Strukturierung bedeute hier:
S.5.1: Das Klammerinnere bilde eine Einheit, wohingegen das Klammeräußere vom Klammerinneren abgegrenzt werde!
S.5.2: Das Klammerinnere werde zuerst ausgewertet, dann erst das Klammeräußere!
S.6: Anmerkungs- und Strukturierungsklammern in Sätzen sollen verschachtelt werden dürfen.
S.7: Es folgt ein Beispiel.
S.8: Der folgende Satz ist mehrdeutig: Das Buch, das ich las, war nicht kurz und schön.
S.9: Er kann bedeuten, nennen wir es Bedeutung1: Das Buch, das ich las, war schön und nicht kurz.
S.10: Er kann aber auch bedeuten, nennen wir es Bedeutung2: Das Buch, das ich las, war nicht kurz oder nicht schön.
S.11: Um Bedeutung1 zu erhalten, kann folgendermaßen geklammert werden: Das Buch, das ich las, war (nicht kurz) und schön.
S.12: Um Bedeutung2 zu erhalten, kann folgendermaßen geklammert werden: Das Buch, das ich las, war nicht (kurz und schön).
S.13: Oder auch [Bedeutung2]: Das Buch (das ich las) war nicht (kurz und schön).

§6 Zeichen und Zeichenketten

S.1: Eine Zeichenkette, die keinen Backslash enthält, soll zitiert werden dürfen, indem sie von einem Backslash und einem O am Anfang ["O" steht für "opening"] und einem Backslash und einem C am Ende ["C" steht für "closing"] eingeschlossen wird.
S.2: Zeichenkettenzitate, wie in S.1 beschrieben, sollen verschachtelt werden dürfen.
S.3: Ein Beispiel: ' ' [Leerzeichen], ':' [Doppelpunkt], 'a', 'b' und 'c' sind Zeichen.
S.4: Noch ein Beispiel: „“ [leere Zeichenkette], „ “ [Leerzeichen], „Anton“ und „Bert“ sind Zeichenketten.
S.5: Noch ein Beispiel: \OSie sagte \OSein Name ist \OPeter\C.\C und \OEr ist 15 Jahre alt.\C.\C ist eine verschachtelte Zeichenkette.

§7 Zeichenkettentupel

S.1: Für jedes Zeichenkettenzitat SQ1 und jedes Zeichenkettenzitat SQ2 [wie in §6.S.1 und §6.S.2 beschrieben] gelte: '{' vkm. SQ1 vkm. ';' vkm. SQ2 vkm. '}' sei ein **Zeichenkettentupel**!
S.2: SQ1 heiße dabei **erste Komponente** des Zeichenkettentupels und SQ2 **zweite Komponente**!
S.3: **Ein Beispiel:** \O{ \OAnton\C ; \OBert\C }\C ist ein Zeichenkettentupel.

§8 Logikkalküle, H.-en und HS.-en

S.1: Das Nichts sei eine **HS.**!
S.2: Das Nichts als HS., nennen wir es **VDerS**, sei an jeden Logikkalkül **angefügt**!
S.3: Ein **Logikkalkül** sei ein Text, der Herleitungsregeln enthalten kann!
S.4: Eine **Herleitungsregel** eines Logikkalküls gibt an, wie aus einer HS. DerSB ["B" steht für "before"], die an den Logikkalkül angefügt ist, durch Verketten mit einer H. DerAdd eine HS. DerSA ["A" steht für "after"], die an den Logikkalkül **angefügt** sei, entsteht.
S.5: Eine **H.** sei ein Zeichenkettentupel!
S.6: Für jeden Logikkalkül LC1 und jede HS. DerS1, die an LC1 angefügt ist, gelte:
S.6.1: Wird eine Herleitungsregel DerR1 von LC1 auf DerS1 angewendet, so ist DerS1 die DerSB der Herleitungsregel.
S.6.2: Die erste Komponente von DerAdd sei eine Regelanwendungsbeschreibung, d.h. eine Beschreibung, die angibt, welche Regel von LC1 wie angewendet wird!
S.6.3: Die zweite Komponente von DerAdd sei der hergeleitete Aussagesatz [z.B. in Prädikatenlogik], der durch Anwendung von DerR1 entsteht!
S.7: Außer den in S.1 und S.4 definierten **HS.-en** gebe es keine HS.-en!
S.8: **Jede HS. ist an einen Logikkalkül angefügt.** [Beweis erfolgt nachfolgend in S.9 – S.12]
S.9: Gemäß S.7 sind die einzigen HS.-en diejenigen, die in S.1 und S.4 definiert werden.
S.10: Gemäß S.2 ist die in S.1 definierte HS. an einen Logikkalkül angefügt.
S.11: Auch die in S.4 definierten HS.-en sind per Definition an einen Logikkalkül angefügt.
S.12: Damit ist die in S.8 aufgestellte Behauptung bewiesen. [q.e.d.]
S.13: Für alle **HS.-en** DerS1 gelte: Gibt es in DerS1 eine H. Der1, deren zweite Komponente gleich \OFalsum\C ist, so heiße DerS1 **inkorrekt**, andernfalls **korrekt**!
S.14: Für jeden **Logikkalkül** LC1 gelte: Gibt es eine an LC1 angefügte inkorrekte HS., so heiße LC1 **inkorrekt**, andernfalls **korrekt**!
S.15: Für alle **HS.-en** DerS1 und DerS2 gelte: Ist DerS2 kürzer als DerS1, so **baue** DerS2 nicht auf DerS1 **auf**!
S.16: Für alle HS.-en DerS1 und DerS2 gelte: Sind DerS1 und DerS2 gleich lang, so baue DerS2 auf DerS1 auf g.d.w. DerS1 und DerS2 gleich sind!
S.17: Für alle HS.-en DerS1 und DerS2 und jede H. Der1 mit (DerS2 vkm. Der1) ist eine HS. gelte: Ist (DerS2 vkm. Der1) länger als DerS1, so baue (DerS2 vkm. Der1) auf DerS1 auf g.d.w. DerS2 auf DerS1 aufbaut!
S.18: **Jede HS. baut auf VDerS auf.** [Beweis erfolgt nachfolgend in S.19 – S.24]

S.19: Gemäß S.16 mit VDerS als DerS1 und VDerS als DerS2, baut VDerS auf VDerS auf, da VDerS = VDerS.
S.20: Gemäß S.17 mit VDerS als DerS1 gilt: Für jede HS. DerS2 und jede H. Der1 mit (DerS2 vkm. Der1) ist eine HS. gilt: Ist (DerS2 vkm. Der1) länger als VDerS, so baue (DerS2 vkm. Der1) auf VDerS auf g.d.w. DerS2 auf VDerS aufbaut.
S.21: (DerS2 vkm. Der1) ist länger als VDerS, da (DerS2 vkm. Der1) aus mindestens einer H. besteht, während VDerS aus 0 H.-en besteht.
S.22: Es gilt also für jede HS. DerS2 und jede H. Der1 mit (DerS2 vkm. Der1) ist eine HS.: (DerS2 vkm. Der1) baut auf VDerS auf g.d.w. DerS2 baut auf VDerS auf.
S.23: Insbesondere gilt für jede HS. DerS2 und jede H. Der1 mit (DerS2 vkm. Der1) ist eine HS.: Aus DerS2 baut auf VDerS auf folgt, (DerS2 vkm. Der1) baut auf VDerS auf.
S.24: Wenn man eine vollständige Induktion mit S.19 als Induktionsanfang und S.23 als Induktionsschritt durchführt, erhält man S.18. [q.e.d.]
S.25: Offensichtlich gilt: **VDerS ist korrekt.**
S.26: Für jeden Logikkalkül LC1 und jede HS. DerS1, die an LC1 angefügt ist, gelte: DerS1 heiße **Aufbau-korrekt** unter LC1, wenn jede HS. DerS2, die auf DerS1 aufbaut und an LC1 angefügt ist, korrekt ist, andernfalls heiße DerS1 **Aufbau-inkorrekt** unter LC1!
S.27: **Für jeden Logikkalkül LC1 und jede an LC1 angefügte HS. DerS1 gilt: Aus DerS1 ist inkorrekt folgt, DerS1 ist Aufbau-inkorrekt unter LC1.** [Beweis erfolgt nachfolgend in S.28 & S.29]
S.28: Für jeden Logikkalkül LC1 und jede an LC1 angefügte HS. DerS1 gilt: Aus DerS1 ist inkorrekt folgt, es gibt eine HS., die auf DerS1 aufbaut und zu LC1 gehört, nämlich DerS1 selbst [gemäß S.16 baut DerS1 auf DerS1 auf], und inkorrekt ist.
S.29: Daraus folgt die Behauptung in S.27. [q.e.d.]
S.30: **Für jeden Logikkalkül LC1 gilt: LC1 ist korrekt g.d.w. VDerS Aufbau-korrekt unter LC1 ist.** [Beweis erfolgt nachfolgend in S.31 – S.41]
S.31: Für jeden Logikkalkül LC1 gilt: Aus LC1 ist korrekt folgt, VDerS ist Aufbau-korrekt unter LC1. [Beweis erfolgt nachfolgend in S.32 – S.35]
S.32: Für jeden Logikkalkül LC1 gilt: Aus LC1 ist korrekt folgt gemäß S.14, jede HS., die an LC1 angefügt ist, ist korrekt.
S.33: Für jeden Logikkalkül LC1 gilt: Aus jede HS., die an LC1 angefügt ist, ist korrekt, folgt, jede HS., die an LC1 angefügt ist und die auf VDerS aufbaut, ist korrekt.
S.34: Für jeden Logikkalkül LC1 gilt: Aus jede HS., die an LC1 angefügt ist und die auf VDerS aufbaut, ist korrekt, folgt gemäß S.26 (wo wir für DerS1 VDerS einsetzen), VDerS ist Aufbau-korrekt unter LC1.
S.35: Fassen wir S.32 – S.34 zusammen, erhalten wir S.31. [q.e.d.]
S.36: Für jeden Logikkalkül LC1 gilt: Aus VDerS ist Aufbau-korrekt unter LC1 folgt, LC1 ist korrekt. [Beweis erfolgt nachfolgend in S.37 – S.40]
S.37: Für jeden Logikkalkül LC1 gilt: Aus VDerS ist Aufbau-korrekt unter LC1 folgt gemäß S.26, jede HS., die auf VDerS aufbaut und an LC1 angefügt ist, ist korrekt.
S.38: Für jeden Logikkalkül LC1 gilt: Aus jede HS., die auf VDerS aufbaut und an LC1 angefügt ist, ist korrekt folgt gemäß S.18, jede HS., die an LC1 angefügt ist, ist korrekt.
S.39: Für jeden Logikkalkül LC1 gilt: Aus jede HS., die an LC1 angefügt ist, ist korrekt folgt gemäß S.14, LC1 ist korrekt.
S.40: Fassen wir S.37 – S.39 zusammen, erhalten wir S.36. [q.e.d.]
S.41: Fassen wir S.31 und S.36 zusammen, erhalten wir S.30. [q.e.d.]
S.42: **Für alle HS.-en DerS1, DerS2 und DerS3 gilt: Aus DerS2 baut auf DerS1 auf und DerS3 baut auf DerS2 auf folgt, DerS3 baut auf DerS1 auf.** [Beweis erfolgt nachfolgend in S.43 – S.53]
S.43: Für alle HS.-en DerS2 und DerS3 mit (DerS3 und DerS2 sind gleich lang) gilt: Aus

DerS3 baut auf DerS2 auf folgt mittels S.16, DerS3 und DerS2 sind gleich.
S.44: Für alle HS.-en DerS1, DerS2 und DerS3 mit (DerS3 und DerS2 sind gleich lang) gilt: Aus DerS2 baut auf DerS1 auf und DerS3 baut auf DerS2 auf folgt mittels S.43, DerS3 und DerS2 sind gleich.
S.45: Für alle HS.-en DerS1, DerS2 und DerS3 mit (DerS3 und DerS2 sind gleich lang) gilt: Aus DerS2 baut auf DerS1 auf und DerS3 und DerS2 sind gleich folgt, DerS3 baut auf DerS1 auf.
S.46: Fassen wir S.44 und S.45 zusammen, gilt für alle HS.-en DerS1, DerS2 und DerS3 mit (DerS3 und DerS2 sind gleich lang) : Aus DerS2 baut auf DerS1 auf und DerS3 baut auf DerS2 auf folgt, DerS3 baut auf DerS1 auf.
S.47: Für alle HS.-en DerS1, DerS2 und DerS4 und jede H. Der1 mit (DerS2 vkm. Der1) ist eine HS. und (DerS4 vkm. Der1) ist länger als DerS2 gilt: Aus (DerS4 vkm. Der1) baut auf DerS2 auf folgt mittels S.17, DerS4 baut auf DerS2 auf.
S.48: Für alle HS.-en DerS1, DerS2 und DerS4 und jede H. Der1 mit (DerS2 vkm. Der1) ist eine HS. und (DerS4 vkm. Der1) ist länger als DerS2 gilt: Aus (aus DerS2 baut auf DerS1 auf und DerS4 baut auf DerS2 auf folgt, DerS4 baut auf DerS1 auf) folgt, aus (DerS2 baut auf DerS1 auf und (DerS4 vkm. Der1) baut auf DerS2 auf) folgt mittels S.47, (DerS2 baut auf DerS1 auf und DerS4 baut auf DerS2 auf) und daraus folgt, DerS4 baut auf DerS1 auf, und daraus folgt mittels S.17, (DerS4 vkm. Der1) baut auf DerS1 auf.
S.49: Fassen wir zusammen, erhalten wir, für alle HS.-en DerS1, DerS2 und DerS4 und jede H. Der1 mit (DerS2 vkm. Der1) ist eine HS. und (DerS4 vkm. Der1) ist länger als DerS2 gilt: Aus (aus DerS2 baut auf DerS1 auf und DerS4 baut auf DerS2 auf folgt, DerS4 baut auf DerS1 auf) folgt, (aus DerS2 baut auf DerS1 auf und (DerS4 vkm. Der1) baut auf DerS2 auf folgt, (DerS4 vkm. Der1) baut auf DerS1 auf).
S.50: Wenn man eine vollständige Induktion mit S.46 als Induktionsanfang und S.49 als Induktionsschritt durchführt, erhält man: Für alle HS.-en DerS1, DerS2 und DerS3 mit DerS3 ist länger oder gleich lang wie DerS2 gilt: Aus DerS2 baut auf DerS1 auf und DerS3 baut auf DerS2 auf folgt, DerS3 baut auf DerS1 auf.
S.51: Für alle HS.-en DerS1, DerS2 und DerS3 mit DerS3 ist kürzer als DerS2 gilt: (DerS3 baut auf DerS2 auf) ist wegen S.15 nicht erfüllt.
S.52: Daraus folgt für alle HS.-en DerS1, DerS2 und DerS3 mit DerS3 ist kürzer als DerS2: Aus DerS2 baut auf DerS1 auf und DerS3 baut auf DerS2 auf folgt, DerS3 baut auf DerS1 auf.
S.53: Fassen wir S.50 und S.52 zusammen, erhalten wir S.42. [q.e.d.]
S.54: Die **Herleitungssequenzmengen** seien folgendermaßen [S.55 & S.56] definiert.
S.55: Für jeden Logikkalkül LC1 gelte: **HSM**(LC1;0) sei die Menge der HS.-en, die VDerS enthält und sonst keine HS.!
S.56: Für jeden Logikkalkül LC1 und jede natürliche Zahl n gelte: HSM(LC1;n+1) sei die Menge der HS.-en [DerSA], die durch Anwenden einer Herleitungsregel von LC1 auf ein Element von HSM(LC1;n) [DerSB] entsteht!
S.57: Offensichtlich **gilt für jeden Logikkalkül LC1 und jede natürliche Zahl n: Jedes Element von HSM(LC1;n) ist an LC1 angefügt.**
S.58: Wir definieren nun die Länge **L** von HS.-en.
S.59: Es sei L(VDerS) = 0 !
S.60: Für jede HS. DerS1 und jede H. Der1 mit (DerS2 vkm. Der1) ist eine HS. gelte: L(DerS1 vkm. DerS) = L(DerS1) + 1 !
S.61: **Für jeden Logikkalkül LC1, jede natürliche Zahl n und jedes Element DerS1 von HSM(LC1;n) gilt: L(DerS1) = n .** [Beweis erfolgt nachfolgend in S.62 – S.64]
S.62: Mit S.55 und S.59 folgt, für jeden Logikkalkül LC1 und jedes Element DerS1 von HSM(LC1;0) gilt: DerS1 = VDerS und L(DerS1) = L(VDerS) = 0 .

S.63: Für jeden Logikkalkül LC1 und jede natürliche Zahl n gilt: Aus (für jedes Element DerS1 von HSM(LC1;n) gilt L(DerS1) = n) folgt, jedes Element von HSM(LC1;n+1) besteht aus (einer HS. DerS2, die an LC1 angefügt ist, und einer H. Der1) und gemäß S.56 ist DerS2 Element von HSM(LC1;n) und somit ist L(DerS2) = n und gemäß S.60 ist L(DerS2 vkm. Der1;n+1) = L(DerS2) + 1 = n + 1.
S.64: Führt man eine vollständige Induktion mit S.62 als Induktionsanfang und S.63 als Induktionsschritt durch, so erhält man S.61. [q.e.d.]
S.65: Offensichtlich **gilt für jeden Logikkalkül LC1 und jede an LC1 angefügte HS. DerS1: DerS1 ist in HSM(LC1;L(DerS1)) enthalten.**
S.66: Zudem **gilt** offensichtlich **für jeden Logikkalkül LC1 und jede natürliche Zahl n: Jedes Element von HSM(LC1;n+1) baut auf einem Element von HSM(LC1;n) auf.**
S.67: **Für jeden Logikkalkül LC1 und alle natürlichen Zahlen m und n mit m ist kleiner gleich n gilt: Jedes Element von HSM(LC1;n) baut auf einem Element von HSM(LC1;m) auf.** [Beweis erfolgt nachfolgend in S.68 – S.70]
S.68: S.67 gilt offensichtlich für m = n.
S.69: Für jeden Logikkalkül LC1 und alle natürlichen Zahlen m und n mit m ist kleiner gleich n gilt: Aus jedes Element von HSM(LC1;n) baut auf einem Element von HSM(LC1;m) auf folgt, jedes Element von HSM(LC1;n+1) baut wegen S.66 auf einem Element von HSM(LC1;n) auf und gemäß S.42 baut jedes Element von HSM(LC1;n+1) auf einem Element von HSM(LC1;m) auf.
S.70: Führt man eine vollständige Induktion mit S.68 als Induktionsanfang und S.69 als Induktionsschritt durch, so erhält man S.67. [q.e.d.]

§9 Der Limberg-Kalkül

S.1: Im folgenden [S.2 – S.14] sei der simple Logikkalkül namens **Limberg-Kalkül** definiert!
S.2: „PIsEl" und „PIsNotEl" seien die einzigen beiden **Prädikate** des Limberg-Kalküls!
S.3: 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9' und '_' seien die einzigen **Zeichen vom Typ 1**!
S.4: Ein Zeichen vom Typ 1 sei eine **Zeichenkette vom Typ 1**!
S.5: (Eine Zeichenkette vom Typ 1 vkm. einem Zeichen vom Typ 1) sei wiederum eine Zeichenkette vom Typ 1!
S.6: Ansonsten gebe es keine Zeichenkette vom Typ 1!
S.7: Für jede Zeichenkette Str1 vom Typ 1 gelte: ('F' vkm. Str1) sei eine **freie Variable** des Limberg-Kalküls!
S.8: Für jede Zeichenkette Str1 vom Typ 1 gelte: ('Q' vkm. Str1) sei eine **quantisierte Variable** des Limberg-Kalküls!
S.9: Ansonsten gebe es keine Variablen des Limberg-Kalküls!
S.10: **Herleitungsregel 1:** Für jedes Prädikat P1 und alle freie Variablen F1 und F2 gelte: Die zweite Komponente von DerAdd darf (P1 vkm. '(' vkm. F1 vkm. ';' vkm. F2 vkm. ')') sein, sofern DerSA Aufbau-korrekt unter dem Limberg-Kalkül ist.
S.11: **Herleitungsregel 2:** Für jedes Prädikat P1 und jede quantisierte Variable Q1 und jede freie Variable F1 gelte: Die zweite Komponente von DerAdd darf (' ∀ ' vkm. Q1 vkm. ':' vkm. P1 vkm. '(' vkm. Q1 vkm. ';' vkm. F1 vkm. ')') sein, sofern DerSA Aufbau-korrekt unter dem Limberg-Kalkül ist.
S.12: **Herleitungsregel 3:** Für jedes Prädikat P1 und jede quantisierte Variable Q1 und

alle freien Variablen F1 und F2 gelte: Enthält DerSB eine Herleitung, deren zweite Komponente gleich (' ∀ ' vkm. Q1 vkm. ':' vkm. P1 vkm. '(' vkm. Q1 vkm. ';' vkm. F1 vkm. ')') ist, so darf die zweite Komponente von DerAdd (P1 vkm. '(' vkm. F2 vkm. ';' vkm. F1 vkm. ')') sein.
S.13: **Herleitungsregel 4:** Für jedes Prädikat P1 und alle quantisierten Variablen Q1 und Q2 mit Q1 ungleich Q2 und jede freie Variable F1 gelte: Enthält DerSB eine Herleitung, deren zweite Komponente gleich (' ∀ ' vkm. Q1 vkm. ':' vkm. P1 vkm. '(' vkm. Q1 vkm. ';' vkm. F1 vkm. ')') ist, so darf die zweite Komponente von DerAdd (' ∃ ' vkm. Q2 vkm. ':' vkm. ' ∀ ' vkm. Q1 vkm. ':' vkm. P1 vkm. '(' vkm. Q1 vkm. ';' vkm. Q2 vkm. ')') sein.
S.14: **Herleitungsregel 5:** Für alle freien Variablen F1 und F2 gelte: Enthält DerSB eine Herleitung, deren zweite Komponente gleich („PIsEl(“ vkm. F1 vkm. ';' vkm. F2 vkm. ')') ist, und enthält DerSB eine Herleitung, deren zweite Komponente gleich („PNotIsEl(“ vkm. F1 vkm. ';' vkm. F2 vkm. ')') ist, so darf die zweite Komponente von DerAdd gleich \OFalsum\C sein.
S.15: Durch Herleitungsregel 1 und Herleitungsregel 2 können bestimmte festsetzende Definitionen aufgestellt werden.
S.16: Durch die Forderung in Herleitungsregel 1 und in Herleitungsregel 2, dass DerSA Aufbau-korrekt unter dem Limberg-Kalkül zu sein hat, wird §2.S.6 genüge getan.
S.17: In Herleitungsregel 3 speit der Allquantor etwas aus [bildlich gesprochen].
S.18: In Herleitungsregel 4 wird von einem Beispiel auf die Existenz geschlossen.
S.19: In Herleitungsregel 5 wird von einem Widerspruch auf Falsum geschlossen.
S.20: **Der Limberg-Kalkül ist korrekt.** [Beweis erfolt nachfolgend in S.21 – S.25]
S.21: Zu beweisen ist gemäß §8.S.14: Jede HS., die an den Limberg-Kalkül angefügt ist, ist korrekt.
S.22: Es seien die folgenden zwei Fälle unterschieden.
S.23: Fall 1, HSM(Limberg-Kalkül;1) ist leer:
S.23.1: Dann ist die einzige an den Limberg-Kalkül angefügte HS. VDerS, welche gemäß §8.S.25 korrekt ist.
S.24: Fall 2, HSM(Limberg-Kalkül;1) ist nicht leer:
S.24.1: Auf VDerS kann offensichtlich nur Herleitungsregel 1 oder Herleitungsregel 2 angewendet werden.
S.24.2: Demzufolge ist jede an den Limberg-Kalkül angefügte HS. aus HSM(Limberg-Kalkül;1) Aufbau-korrekt unter dem Limberg-Kalkül.
S.24.3: VDerS ist gemäß §8.S.2 an den Limberg-Kalkül angefügt und ist gemäß §8.S.25 korrekt.
S.24.4: Für jede an den Limberg-Kalkül angefügte HS. DerS1 mit DerS1 ist ungleich VDerS gilt: L(DerS1) ist größer gleich 1.
S.24.5: Aus §8.S.67 folgt [mit LC1 = Limberg-Kalkül und m = 1]: Für jede an den Limberg-Kalkül angefügte HS. DerS1 mit DerS1 ist ungleich VDerS gilt: Jedes Element von HSM(Limberg-Kalkül;L(DerS1)) baut auf einem Element von HSM(Limberg-Kalkül;1) auf.
S.24.6: Mit §8.S.65 folgt, für jede an den Limberg-Kalkül angefügte HS. DerS1 mit DerS1 ist ungleich VDerS gilt: DerS1 baut auf einem Element von HSM(Limberg-Kalkül;1) auf.
S.24.7: Mit S.24.2 folgt, für jede an den Limberg-Kalkül angefügte HS. DerS1 mit DerS1 ist ungleich VDerS gilt: DerS1 ist korrekt.
S.24.8: Aus S.24.3 und S.24.7 folgt, jede an den Limberg-Kalkül angefügte HS. ist korrekt.
S.25: Aus S.23.1 und S.24.8 folgt S.21 und damit S.20. [q.e.d.]

§10 2 Anwendungsbeispiele des Limberg-Kalküls

S.1: \O { \O
S.1.1: Es werde Herleitungsregel 2 des Limberg-Kalküls angewendet!
S.1.2: In P1 sei \OPIsNotEl\C eingesetzt!
S.1.3: In Q1 sei \OQ1\C eingesetzt!
S.1.4: In F1 sei \OFEmptySet\C eingesetzt!
S.1 [weiter]: \C ; \O
S.1.5: ∀ Q1:PIsNotEl(Q1;FEmptySet)
S.1 [weiter]: \C } \C sei H.1 !
S.2: \O { \O
S.2.1: Es werde Herleitungsregel 4 des Limberg-Kalküls angewendet!
S.2.2: Sie werde auf H.1 angewendet!
S.2.3: In P1 sei \OPIsNotEl\C eingesetzt!
S.2.4: In Q1 sei \OQ1\C eingesetzt!
S.2.5: In Q2 sei \OQEmptySet\C eingesetzt!
S.2.6: In F1 sei \OFEmptySet\C eingesetzt!
S.2 [weiter]: \C ; \O
S.2.7: ∃ QEmptySet: ∀ Q1:PIsNotEl(Q1;QEmptySet)
S.2 [weiter]: \C } \C sei H.2 !
S.3: H.1 sei HS.1 !
S.4: H.1 vkm. H.2 sei HS.2 !
S.5: \O { \O
S.5.1: Es werde Herleitungsregel 2 des Limberg-Kalküls angewendet!
S.5.2: In P1 sei \OPIsEl\C eingesetzt!
S.5.3: In Q1 sei \OQ1\C eingesetzt!
S.5.4: In F1 sei \OFUniversalSet\C eingesetzt!
S.5 [weiter]: \C ; \O
S.5.5: ∀ Q1:PIsEl(Q1;FUniversalSet)
S.5 [weiter]: \C } \C sei H.3 !
S.6: \O { \O
S.6.1: Es werde Herleitungsregel 4 des Limberg-Kalküls angewendet!
S.6.2: Sie werde auf H.3 angewendet!
S.6.3: In P1 sei \OPIsEl\C eingesetzt!
S.6.4: In Q1 sei \OQ1\C eingesetzt!
S.6.5: In Q2 sei \OQUniversalSet\C eingesetzt!
S.6.6: In F1 sei \OFUniversalSet\C eingesetzt!
S.6 [weiter]: \C ; \O
S.6.7: ∃ QUniversalSet: ∀ Q1:PIsEl(Q1;QUniversalSet)
S.6 [weiter]: \C } \C sei H.4 !
S.7: H.3 sei HS.3 !
S.8: H.3 vkm. H.4 sei HS.4 !
S.9: HS.1 ist Aufbau-korrekt unter dem Limberg-Kalkül. [Beweis erfolgt nachfolgend in S.10 – S.]
S.10: Zu beweisen ist gemäß §8.S.26: Jede HS., die an den Limberg-Kalkül angefügt ist und auf HS.1 aufbaut, ist korrekt.
S.11:
S.: HS.3 ist Aufbau-korrekt unter dem Limberg-Kalkül. [Beweis erfolgt nachfolgend in S. – S.]