Dr. Winfried Teschers
Anton-Günther-Straße 26c
91083 Baiersdorf
winfried.teschers@t-online.de

Projektdokument

ASBA

Axiome, Sätze, Beweise und Auswertungen

Projekt zur maschinellen Überprüfung von mathematischen Beweisen und deren Ausgabe in lesbarer Form

Winfried Teschers

1. Juli 2017

Es wird ein Programmsystem beschrieben, das zu eingegebenen Axiomen, Sätzen, und Beweisen letztere prüft, Auswertungen generiert und zu gegebenen Ausgabeschemata eine Ausgabe der Elemente in üblicher Formelschreibweise im LaTeX-Format erstellt.

Inhaltsverzeichnis

1. Analyse

In der Mathematik gibt es eine unüberschaubare Menge an Axiomen, Sätzen, Beweisen, *Fachbegriffen*[1)] und *Fachgebieten*[2)]. Zu den meisten Fachgebieten gibt es noch ungelöste Probleme.

Es fehlt ein System, das einen Überblick bietet und die Möglichkeit, Beweise automatisch zu überprüfen. Außerdem sollte all dies in üblicher mathematischer Schreibweise ein- und ausgegeben werden können. In diesem Dokument werden die Grundlagen für das zu entwickelnde Programmsystem, das **A**xiome, **S**ätze, **B**eweise und **A**uswertungen behandeln kann (ASBA) behandelt.

Ein Programmsystem mit ähnlicher Aufgabenstellung findet sich im GitHub Projekt Hilbert II (siehe [18, 19]). Einige Ideen sind von dort übernommen worden.

1.1. Fragen

Einige der Fragen, die in diesem Zusammenhang auftauchen, werden hier formuliert:

1. *Grundlagen*: Was sind die Grundlagen? Z. B. welche Logik und Mengenlehre.
2. *Basis*: Welche wichtigen Axiome, Sätze, Beweise, Fachbegriffe und Fachgebiete gibt es? Welche davon sind Standard?
3. *Axiome*: Welche Axiome werden bei einem Satz oder Beweis vorausgesetzt? Allgemein anerkannte oder auch strittige, wie z. B. den *Satz vom ausgeschlossenen Dritten* (*tertium non datur*) oder das *Auswahlaxiom*.
4. *Beweis*: Ist ein Beweis fehlerfrei?
5. *Konstruktion*: Gibt es einen konstruktiven Beweis?
6. *Vergleiche*: Welcher Beweis ist besser? Nach welchem Kriterium? Z. B. elegant, kurz, einsichtig oder wenige Axiome. Was heißt eigentlich *elegant*?
7. *Definitionen*: Was ist mit einem Fachbegriff jeweils genau gemeint? Z. B. *Stetigkeit*, *Integral* und *Analysis*.
8. *Abhängigkeiten*: Wie heißt ein Fachbegriff in einer anderen Sprache? Ist wirklich dasselbe gemeint? Was ist mit Fachbegriffen in verschiedenen Fachgebieten?
9. *Überblick*: Ist ein Axiom, Satz, Beweis oder Fachbegriff schon einmal – ggf. abweichend – definiert, formuliert oder bewiesen worden?
10. *Darstellung*: Wie kann man einen Satz und den zugehörigen Beweis – ggf. auch spezifisch für ein Fachgebiet – darstellen?
11. *Forschung*: Welche Probleme gibt es noch zu erforschen.

1) *Fachbegriffe* sind Namen für mathematische Elemente und Konstruktionen, z. B. Axiome, Sätze, Beweise und Fachgebiete. Symbole können als spezielle Fachbegriffe aufgefasst werden.

2) Ein *Fachgebiet* ist ein Teil der Mathematik mit einer zugehörigen Basis an Axiomen, Sätzen und spezifischen Fachbegriffen und Darstellungen. Z. B. *Logik*, *Mengenlehre* und *Gruppentheorie*. Ein Fachgebiet kann sehr klein sein und im Extremfall kein einziges Element enthalten. *Umgebung* wäre eine bessere Bezeichnung, ist aber schon ein verbreiteter Fachbegriff, so dass hier die Bezeichnung Fachgebiet verwendet wird.

1.2. Eigenschaften

Ausgehend von den Fragen in Abschnitt 1.1 auf der vorherigen Seite soll ASBA entwickelt werden, das die folgenden Eigenschaften hat:

1. *Daten*: Axiome, Sätze, Beweise, Fachbegriffe und Fachgebiete können in formaler Form gespeichert werden – auch nicht oder unvollständig bewiesene Sätze. Dabei soll die übliche mathematische Schreibweise verwendet werden können.
2. *Definitionen*: Es können Fachbegriffe für Axiome, Sätze, Beweise und Fachgebiete – letztere mit eigenen Axiomen, Sätzen, Beweisen, Fachbegriffen und über- oder untergeordneten Fachgebieten – definiert werden. Die Definitionen dürfen wiederum an dieser Stelle schon bekannte Fachbegriffe und Fachgebiete verwenden.
3. *Prüfung*: Vorhandene Beweise können automatisch geprüft werden.
4. *Ausgaben*: Die Axiome, Sätze und Beweise können in üblicher Schreibweise – abhängig von Sprache und Fachgebiet – ausgegeben werden.
5. *Auswertungen*: Zusätzlich zur Ausgabe der gespeicherten Daten sind verschiedene Auswertungen möglich, unter anderem für die meisten der unter Abschnitt 1.1 auf der vorherigen Seite behandelten Fragen.

Damit ASBA nicht umsonst erstellt wird und möglichst breite Verwendung findet, werden noch zwei Punkte angefügt:

6. *Lizenz*: Die Software ist *Open Source*.
7. *Akzeptanz*: ASBA wird von Mathematikern akzeptiert und verwendet.

Tabelle 1.1 zeigt, wie sich die Eigenschaften zu den Fragen in Abschnitt 1.1 auf der vorherigen Seite verhalten. Mit einem X werden die Spalten einer Zeile markiert, deren zugehörige Eigenschaften zur Beantwortung der entsprechenden Frage beitragen sollen. Idealerweise sollte die Erfüllung aller angegebenen Eigenschaften alle gestellten Fragen beantworten, was allerdings illusorisch ist.

Frage \ **Eigenschaft**		1 Daten	2 Definitionen	3 Prüfung	4 Ausgaben	5 Auswertungen	6 Lizenz	7 Akzeptanz
1	Grundlagen	X	X	-	X	X	-	-
2	Basis	X	X	-	X	X	-	-
3	Axiome	X	X	-	X	X	-	-
4	Beweis	X	-	X	X	-	-	-
5	Konstruktion	X	-	-	X	-	-	-
6	Vergleiche	X	-	-	-	X	-	-
7	Definitionen	X	X	-	X	-	-	-
8	Abhängigkeiten	X	-	-	X	-	-	-
9	Überblick	X	-	-	-	X	-	-
10	Darstellung	-	X	-	X	-	-	-
11	Forschung	X	-	-	-	X	-	-

Tabelle 1.1.: 1.1 Fragen → 1.2 Eigenschaften

1.3. Ziele

Um die Eigenschaften von Abschnitt 1.2 auf der vorherigen Seite zu erreichen, werden für ASBA die folgenden Ziele[3)] gesetzt:

1. *Daten*: Es enthält möglichst viele wichtige Axiome, Sätze, Beweise, Fachbegriffe, Fachgebiete und Ausgabeschemata[4)].
2. *Form*: Die Daten liegt in formaler, geprüfter Form vor.
3. *Eingaben*: Die Eingabe von Daten erfolgt in einer formalen Syntax unter Verwendung der üblichen mathematischen Schreibweise.
4. *Prüfung*: Vorhandene Beweise können automatisch geprüft werden.
5. *Ausgaben*: Die Ausgabe kann in einer eindeutigen, formalen Syntax gemäß vorhandener Ausgabeschemata erfolgen.
6. *Auswertungen*: Zusätzlich zur Ausgabe der Daten sind verschiedene Auswertungen möglich. Insbesondere kann zu jedem Beweis angegeben werden, wie viele Beweisschritte und welche Axiome und Sätze[5)] er verwendet.
7. *Anpassbarkeit*: Fachbegriffe und die Darstellung bei der Ausgabe können mit Hilfe von – gegebenenfalls unbenannten – untergeordneten Fachgebieten angepasst werden.
8. *Individualität*: Axiome und Sätze können für jeden Beweis individuell vorausgesetzt werden. Dabei sind fachgebietsspezifische Fachbegriffe erlaubt.
9. *Internet*: Die Daten können auf mehrere Dateien verteilt sein. Ein Teil davon – oder sogar alle – können im Internet liegen.
10. *Kommunikation*: Die Kommunikation mit ASBA kann mit den Fachbegriffen der einzelnen Fachgebiete erfolgen.
11. *Zugriff*: Der Zugriff auf ASBA kann lokal und über das Internet erfolgen.
12. *Unabhängigkeit*: ASBA kann online und offline arbeiten.
13. *Rekursion*: Es kann rekursiv über alle verwendeten Dateien – auch solchen, die im Internet liegen – ausgewertet werden.
14. *Bedienbarkeit*: ASBA ist einfach zu bedienen.
15. *Lizenz*: Die Software ist *Open Source*.

Der Punkt 16 wurde noch eingefügt, damit ASBA effizient arbeiten kann und um die Akzeptanz zu erhöhen:

16. *Zwischenspeicher*: Wichtige Auswertungen können an vorhandenen Dateien angehängt oder separat in eigenen Dateien gespeichert werden.

Die Tabelle 1.2 auf der nächsten Seite zeigt wieder, wie sich die Ziele zu den Eigenschaften in Abschnitt 1.2 auf der vorherigen Seite verhalten. Mit einem X werden wieder die Spalten einer Zeile markiert, deren zugehörige Ziele zur Sicherstellung der entsprechenden Eigenschaft beitragen sollen. Idealerweise sollte durch Erreichen aller aufgestellten Ziele ASBA alle angegebenen Eigenschaften aufweisen, was wahrscheinlich ebenfalls illusorisch ist.

3) Es sind eigentlich Anforderungen. Da dieser Begriff auch im Kapitel 3 auf Seite 28 verwendet wird, werden die Anforderungen hier *Ziele* genannt.

4) Um den Punkt 4 von Abschnitt 1.2 auf der vorherigen Seite erfüllen zu können, werden noch fachgebietsspezifische Ausgabeschemata benötigt, welche die Art der Ausgaben beschreiben.

5) Sätze, die quasi als Axiome verwendet werden.

	Eigenschaft \ Ziel	1 Daten	2 Form	3 Eingaben	4 Prüfung	5 Ausgaben	6 Auswertungen	7 Anpassbarkeit	8 Individualität	9 Internet	10 Kommunikation	11 Zugriff	12 Unabhängigkeit	13 Rekursion	14 Bedienbarkeit	15 Lizenz	16 Zwischenspeicher
1	Daten	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-
2	Definitionen	X	-	X	-	-	-	-	-	-	-	-	-	-	-	-	-
3	Prüfung	-	-	-	X	-	-	-	-	-	-	-	-	-	-	-	-
4	Ausgaben	-	-	-	-	X	-	-	-	-	-	-	-	-	-	-	-
5	Auswertungen	-	-	-	-	-	X	-	-	-	-	-	-	-	-	-	-
6	Lizenz	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	-
7	Akzeptanz	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Tabelle 1.2.: 1.2 Eigenschaften → 1.3 Ziele

1.4. Zusammenfassung

	Frage \ Ziel	1 Daten	2 Form	3 Eingaben	4 Prüfung	5 Ausgaben	6 Auswertungen	7 Anpassbarkeit	8 Individualität	9 Internet	10 Kommunikation	11 Zugriff	12 Unabhängigkeit	13 Rekursion	14 Bedienbarkeit	15 Lizenz
1	Grundlagen	X	X	X	-	X	X	x	-	-	-	-	-	-	-	-
2	Basis	X	X	X	-	X	X	x	x	-	-	-	-	-	-	-
3	Axiome	X	X	X	-	X	X	x	-	-	-	-	-	-	-	-
4	Beweis	X	X	X	X	X	-	-	x	-	-	-	-	-	-	-
5	Konstruktion	X	X	X	-	X	-	-	x	-	-	-	-	-	-	-
6	Vergleiche	X	X	X	-	-	X	-	x	-	-	-	-	-	-	-
7	Definitionen	X	X	X	-	X	-	x	-	-	-	-	-	-	-	-
8	Abhängigkeiten	X	X	X	-	X	-	x	-	-	-	-	-	-	-	-
9	Überblick	X	X	X	-	-	X	x	-	-	-	-	-	-	-	-
10	Darstellung	X	-	X	-	X	-	x	-	-	-	-	-	-	-	-
11	Forschung	X	X	X	-	-	X	x	-	-	-	-	-	-	-	-
Die nächsten beiden Punkte sind Eigenschaften aus Abschnitt 1.2 auf Seite 5:																
6	Lizenz	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X
7	Akzeptanz	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Tabelle 1.3.: 1.1 Fragen → 1.3 Ziele

Die Tabelle 1.3 ist eine Kombination aus den Tabellen 1.1 auf Seite 5 und 1.2 und zeigt, wie sich die Ziele in Abschnitt 1.3 auf der vorherigen Seite zu den Fragen in Abschnitt 1.1 auf Seite 4 verhalten. Auch hier werden mit einem X die Spalten einer Zeile markiert, deren zugehörige Ziele für die Beantwortung der entsprechenden Frage nötig sind. Mit einem kleinen x werden sie markiert, wenn sie zur Beantwortung der Fragen nicht nötig, aber von Interesse sind. Idealerweise sollte das Erreichen aller aufgestellten Ziele alle gestellten Fragen beantworten, was natürlich auch illusorisch ist.

1.5. Umgebung von ASBA

In der Abbildung 1.1 wird beschrieben, welche Interaktionen ASBA mit der Umgebung hat, d. h. welche Ein- und Ausgaben existieren und woher sie kommen bzw. wohin sie gehen.

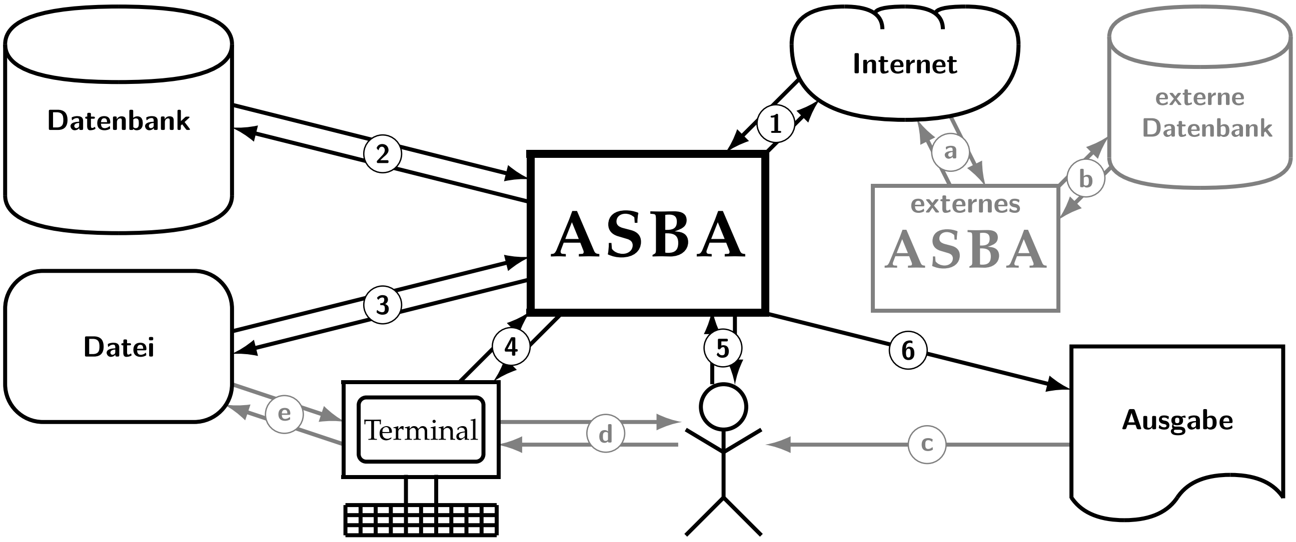

Abbildung 1.1.: Die Umgebung von ASBA

In den in Abbildung 1.1 abgebildeten Datenflüssen (1) bis (6) und (a) bis (e) werden die folgenden Daten übertragen:

(1) **ASBA → Internet** Inhalte der Datenbank.

Internet → ASBA Inhalte der externen Datenbank.

(2) **Datenbank → ASBA** Inhalte der Datenbank und Antworten auf Datenbankanweisungen.

ASBA → Datenbank Inhalte der Datei, der externen Datenbank und Datenbankanweisungen.

(3) **Datei → ASBA** Inhalte der Datei.

ASBA → Datei Die Datei wird um zusätzliche Auswertungen ergänzt, z. B. ob die Beweise korrekt sind, welche Axiome und Sätze – auch externe aus dem Internet – verwendet wurden, Anzahl der Beweisschritte usw.

(4) **Terminal → ASBA** Anweisungen, Daten und Batchprogramme.

ASBA → Terminal Antworten auf Anweisungen, Auswertungen usw.

Außerdem interaktive Ein- und Ausgabe durch einen Anwender, wie in (5) beschrieben.

(5) **Anwender ↔ ASBA** Interaktive Ein- und Ausgaben durch einen Anwender mit Komponenten von (3), (4) und (6). – Die Kommunikation läuft i. Alg. über ein Terminal.

(6) **ASBA → Ausgabe** Inhalte von Datei und Datenbank in lesbarer Form, u. a. mit Hilfe von Ausgabeschemata auch in Formelschreibweise. Die Ausgabe kann auch in eine Datei erfolgen, z. B. im LaTeX-Format.

(a) Nur die für ASBA relevanten Daten:

Internet → externes ASBA Inhalte der Datenbank.

externes ASBA → Internet Inhalte der externen Datenbank.

(b) Nur die für ASBA relevanten Daten:

externe Datenbank → externes ASBA Inhalte der externen Datenbank.

externes ASBA → externe Datenbank Inhalte der Datenbank.

(c) **Ausgabe → Anwender** Alle Daten der Ausgabe.

(d) **Anwender ↔ Terminal** Interaktive Ein- und Ausgabe durch einen Anwender, wie in (5) beschrieben.

(e) **Terminal ↔ Datei** Erstellen und Bearbeiten der Datei durch einen Anwender. – siehe (d)

Die Datenflüsse (a) bis (e) erfolgen außerhalb von ASBAund werden nicht weiter behandelt.

Die Datenbank und die Datei enthalten im Prinzip die gleichen Daten, wobei sie in der Datei im Textformat in lesbarer Form und in der Datenbank in einem internen Format vorliegen. Zudem enthält die Datenbank i. Alg. sehr viel mehr Daten. Es handelt sich dabei jeweils um die folgenden Daten:

Axiome Ein Axiom ist eine Aussage oder Behauptung, die nicht aus anderen Aussagen abgeleitet werden kann. Es können wie bei Sätzen Voraussetzungen vorhanden sein, aber keine Beweis.

Sätze Ein Satz besteht aus einer Anzahl von Voraussetzungen, einer Behauptung und einem Beweis, der die Behauptung aus den Voraussetzungen ableitet. Letztere können Axiome und andere Sätze sein, auf die dann verwiesen wird.

Beweise Ein Beweis besteht aus einer Folge von Beweisschritten, die aus gegebenen Voraussetzungen eine Behauptung ableitet.

Fachbegriffe Ein Fachbegriff ist ein Name für ein Prädikat in einem bestimmten Fachgebiet.

Fachgebiete Ein Fachgebiet ist ein Teil der Mathematik mit einer zugehörigen Basis von Axiome, Sätze, Fachbegriffe und Ausgabeschemata, quasi eine untergeordnete Datenbank.

Ausgabeschemata Eine Beschreibung, wie ein bestimmtes mathematisches Objekt ausgegeben werden soll. Dies kann z. B. ein Stück LaTeX-Code mit entsprechenden Parametern sein.

Auswertungen Statistische und andere Auswertungen, die bestimmten Elementen der Datei bzw. Datenbank zugeordnet sind. Z. B. können zu einem Satz alle für einen Beweis notwendigen Axiome angegeben werden – als Verweise.

Die Daten können interne und externe Verweise enthalten.

1.6. Basis von Beweisen

Da ein Computerprogramm erstellt werden soll, muss die Grundstruktur des Vorgehens bei Beweisen definiert werden.[6)]

Die logischen Darstellung von mathematischen Aussagen, wozu auch Axiome und Sätze gehören, erfolgt, da es sich immer um Formeln handelt, an besten mit Zeichenketten.[7)] Mehrdimensionale Formeln, wie z. B. Matrizen, Baumstrukturen, Funktionsschemata und anderes, können auch als (eindimensionale) Zeichenketten dargestellt werden. Beweise sind letztendlich nichts anderes, als erlaubte Transformationen dieser Zeichenketten.

6) siehe [30]

7) Die interne Darstellung der Zeichenketten kann zur Optimierung des Programms von der logischen abweichen.

Zeichen oder Buchstaben, aus denen diese Zeichenketten bestehen dürfen, müssen definiert werden. Außerdem Regeln, wie aus diesen Buchstaben Zeichenketten (Formeln) gebildet werden dürfen. Gebraucht werden also:

- *Bausteine*, also Grundelemente, auch (Satz-)Buchstaben genannt, aus denen komplexe Ausdrücke (Zeichenketten) zusammengesetzt werden können.
- *Formationsregeln*, mit denen festgelegt wird, wie man aus den Bausteinen komplexe Ausdrücke erzeugen kann.

Sätze lassen sich als eine Menge von Formeln (Zeichenketten), den Voraussetzungen, wozu auch Axiome gehören können, einer weiteren Menge von Formeln (Zeichenketten), den Folgerungen, und der Angabe eines Beweises darstellen.

Beweise zu gegebenen Voraussetzungen und Folgerungen lassen sich als Folge von zulässigen Transformationen, beginnend mit den Voraussetzungen und endend mit den Folgerungen, darstellen.

Transformationsregeln definieren, welche Transformationen von gegebenen Formelmengen zulässig sind.[8)]

8) siehe [1, 33, 34]

2. Mathematische Grundlagen

Die mathematischen Grundlagen werden einerseits gebraucht, um die erlaubten Beweisschritte zu definieren (siehe Abschnitt 2.3 auf Seite 14), andererseits dienen sie auch zum Testen von ASBA. Daher behandelt dieses Kapitel die mathematischen Grundlagen viel ausführlicher, als für die Erstellung von ASBA erforderlich ist. Alle hier aufgeführten Axiome, Sätze und Beweise sollen dazu kodiert und die Beweise von ASBA verifiziert werden.

2.1. Metasprache

Wenn man über eine Sprache spricht, braucht man auch eine Sprache, in der Aussagen über die erstere getroffen werden können. Wenn die zuerst genannte Sprache die der Mathematik ist, nimmt man üblicherweise die natürliche Sprache als Metasprache. Leider ist diese oft ungenau, nicht immer eindeutig und abhängig vom Zusammenhang, in dem sie gesprochen wird[1)]. Um diese Probleme in den Griff zu bekommen, wird die Metasprache zum Teil formalisiert. Durch diese Formalisierung erinnert sie dann teilweise schon an mathematische Formeln. Die Sprachebenen sollten aber sorgfältig unterschieden werden.

2.1.1. Metasprachliche Ausdrücke

Ein *metasprachlicher Ausdruck* ist eine in normaler Sprache verfasste Aussage, wie z. B. (a) „Morgen scheint die Sonne.“, (b) „Ich bin 1,83 m groß.“, (c) „Ich habe ein rotes Auto und das kann 200 km/h schnell fahren.“, usw. In einem erweiterten Sinne gehören auch Relationen einschließlich ihrer Operanden dazu[2)], wie z. B. „$A = A$“, „$A \equiv B$“, „$A < B$“, usw.

Während die Beispiele (a) und (b) einfache, nicht mehr zerlegbare metasprachliche Ausdrücke sind, ist Beispiel (c) zusammengesetzt. Für alle drei Aussagen lässt sich feststellen, ob sie richtig sind oder nicht. Das kann man für den zweiten Teil von (c) aber nicht, wenn man nicht weiss worauf sich „das“ bezieht. Natürlich muss auch der Zusammenhang, in dem ein metasprachlicher Ausdruck formuliert wird, bekannt sein, denn z. B. ist die Bedeutung von „Ich“ nur dann bekannt, wenn man weiss von wem die Aussage ist. Auf eine exakte Definition von „metasprachlicher Ausdruck“ wird verzichtet, weil das intuitive Verständnis hier ausreicht. In erster Näherung können aber alle sprachlichen Ausdrücke, die im Prinzip überprüft werden können, als metasprachliche Ausdrücke betrachtet werden.

---

1) Man betrachte die beiden Aussagen „Studenten und Rentner zahlen die Hälfte.“ und „Studenten oder Rentner zahlen die Hälfte.“, die beide das gleiche meinen. – Entnommen aus [1] Abschnitt 1.2 Bemerkung 1.
Ein weiteres Problem ist, dass man unauflösbare Widersprüche formulieren kann, z. B. „Der Barbier ist der Mann im Ort, der genau die Männer im Ort rasiert, die sich nicht selbst rasieren.“. Und der Barbier? Wenn er sich selbst rasiert, dann rasiert er sich nicht selbst, und wenn er sich nicht selbst rasiert, dann rasiert er sich selbst. Was denn nun? – Quelle unbekannt) – Das Problem ist verwandt mit dem Problem der „Menge aller Mengen, die sich nicht selbst enthalten“.

2) Wird statt des Symbols der Name der zugehörigen Relation verwendet, ist dies unmittelbar einleuchtend. So wird z. B. aus der Formel „$A < B$“ die metasprachliche Aussage „A ist kleiner als B“.

Zusammengesetzte metasprachliche Ausdrücke wie (c) können zum Teil formalisiert werden. Dies wird mit den folgenden Definitionen erreicht:

$A \Rightarrow B$	steht für „*Wenn* A [gilt] *dann* [gilt] [auch] B“.
$A \Leftarrow B$	steht für „A [gilt] *sofern* B [gilt]“.
$A \Leftrightarrow B$	steht für „A [gilt] *genau dann wenn* B [gilt]“.
A && B	steht für „[Es gilt] A *und* B“.
$A \ \|\| \ B$	steht für „[Es gilt] A *oder* B“.

Bei den Schlussregeln[3] wird '|' statt '&&' bei unterschiedlicher Priorität[4] verwendet.

$A \mid B$	steht für „A *und* B“.

Offensichtlich sind das alles ebenfalls metasprachliche Ausdrücke, jetzt aber teilweise formalisiert. (c) lässt sich dann ausdrücken als „‚Ich habe ein rotes Auto‘ && ‚das kann 200 km/h schnell fahren.‘“.

Um Verwechslungen mit den logischen Symbolen zu vermeiden, werden für „und“ und „oder“ die Symbole '&&' und '||' verwendet. A und B können als Operanden von '$\Leftrightarrow$', '&&', '||' und '|' ohne Bedeutungsveränderung vertauscht werden. Wird in einer Formel nur einer der Operatoren '&&', '||' oder '|' verwendet, können die Operanden beliebig permutiert werden, so dass dann auch eine Klammerung überflüssig ist. – Ein Symbol für „nicht“ wird hier nicht gebraucht.

Metasprachliche Ausdrücke können auch geklammert werden, um die Reihenfolge der Auswertung eindeutig zu machen. '$\Rightarrow$', '$\Leftarrow$', '$\Leftrightarrow$', '&&', '||' und '|' heißen *metasprachliche Operatoren*. Ihre Prioritäten werden im Unterabschnitt 2.4.2 auf Seite 22 zusammen mit anderen Operatoren definiert.

Sollen zwei metasprachliche Ausdrücke miteinander verglichen werden, muss klar sein auf welche Art; ob z. B. als Zeichenfolgen – mit oder ohne Wertung der Zwischenräume –, als Wahrheitswerte oder auf sonstige Art. Wenn die Art des Vergleichs implizit oder explizit klar ist und sich die beiden Ausdrücke auf diese Art vergleichen lassen, heißen sie *vergleichbar*.

2.1.2. Mit Gleichheit verwandte Symbole

2.1.2.1. Allgemeine Voraussetzungen

In diesem und allen weiteren Abschnitten wird vorausgesetzt:

- Wenn mehrere der im Folgenden definierten Operatoren '$:\Leftrightarrow$', '$:=$', '$=$', '$\neq$' und '$\equiv$' verwendet werden, dann im selben Zusammenhang[5].
- Soweit verwendet sind die *interessierenden Eigenschaften* für '$=$' und '$\equiv$' bekannt. Dabei muss jede interessierende Eigenschaft für '$\equiv$' auch eine für '$=$' sein.
- Soweit verwendet sind die jeweiligen Operanden von '$=$' und '$\equiv$' vergleichbar.

[3] siehe Abschnitt 2.3 auf Seite 14
[4] siehe Tabelle 2.4 auf Seite 24
[5] Statt von einem *Zusammenhang* könnte man auch von einer *Umgebung* sprechen. Diese Bezeichnung ist aber auch ein verbreiteter Fachbegriff, so dass auf seine Verwendung verzichtet wird. Die Exaktheit der Begriffe in diesem Dokument soll für Erstellung von ASBA ausreichen; was darüber hinausgeht, ist nicht Inhalt dieses Dokuments.

2.1.2.2. Definition der mit Gleichheit verwandten Symbole

Unter den Voraussetzungen von Paragraph 2.1.2.1 auf der vorherigen Seite werden die folgenden (metasprachlichen) Operatoren definiert:

$=$ ***Gleichheit*** „$A = B$“ heißt, dass A und B sich in den interessierenden Eigenschaften für '$=$' nicht unterscheiden.[6] – „A ist *dasselbe* wie B“ oder „A ist *identisch* zu B“ – Inwieweit die Begriffe *Gleichheit* und *Identität* korrelieren, wird hier nicht erörtert. siehe [29]

$\neq$ ***Ungleichheit*** „$A \neq B$“ heißt, dass A und B sich in mindestens einer der interessierenden Eigenschaften für '$=$' unterscheiden. „A ist *nicht dasselbe* wie B“ (aber vielleicht das gleiche) oder „A ist *nicht identisch* zu B“.

$\equiv$ ***Äquivalenz*** „$A \equiv B$“ heißt, dass A und B sich in den interessierenden Eigenschaften für '$\equiv$' nicht unterscheiden. – „A ist *das gleiche* wie B“ oder „A ist *so wie* B“.

$\not\equiv$ ***Kontravalenz*** „$A \not\equiv B$“ heißt, dass A und B sich in mindestens einer der interessierenden Eigenschaften für '$\not\equiv$' unterscheiden. – „A ist *nicht das gleiche* wie B“ oder „A ist *nicht so wie* B“.

$:\Leftrightarrow$ ***Metadefinition*** „$A :\Leftrightarrow B$“ heißt, dass der Metaausdruck A *definitionsgemäß gleich* dem Metaausdruck B ist, wobei B auch eine Definition in natürlicher Sprache sein kann. A und B können sich gegenseitig ersetzten. B darf dabei von A weder direkt noch indirekt abhängen, d. h. A darf in B und zugehörigen Definitionen noch nicht vorkommen.

Üblicherweise ist A hier eine Bezeichnung und B eine Aussage, so dass man „$A :\Leftrightarrow B$“ auch als „*A steht für B*“ lesen kann. Oft wird damit Gleichheit ('$=$'), Äquivalenz ('$\equiv$') oder eine andere Relation definiert.

$:=$ ***Definition*** „$A := B$“ heißt, dass der Ausdruck A *definitionsgemäß gleich* dem Ausdruck B ist. Gewissermaßen ist A nur eine andere Schreibweise für B. A und B können sich gegenseitig ersetzten.[7] B darf dabei von A weder direkt noch indirekt abhängen, d. h. A darf in B und zugehörigen Definitionen noch nicht vorkommen. – Man beachte, dass '$:\Leftrightarrow$' und '$:=$' verschiedene Sprachebenen sind.

Üblicherweise ist A hier eine Variable und B eine Formel, so dass man „$A := B$“ auch als „*A steht für B*“ lesen kann.

Es sei noch

$$\mathcal{M} := \{|, \&\&, ||, \Rightarrow, \Leftrightarrow, \Leftarrow, =, \neq, \equiv, \not\equiv, :\Leftrightarrow, :=\}$$

die Menge der metasprachlichen Operatoren und der mit Gleichheit verwandten Symbole.

2.2. Formale Elemente

Ein *formales Element* kann z. B. eine Menge, Zeichenfolge, Zahl, Formel, usw. sein. Zwei formale Elemente A und B sind *vergleichbar*, wenn beide von derselben Art sind, d. h. wenn z. B. jeweils beide Mengen, Zeichenfolgen, Zahlen oder formale Elemente – die vergleichbare Ergebnisse liefern – sind.

Intuitiv scheint klar zu sein, was damit gemeint ist. Wenn aber entschieden werden muss, ob z. B. (a) „1+1“ gleich „2“ oder (b) „1+1“ gleich „1 + 1“ ist, muss man erst entscheiden, von welcher Art die beiden zu vergleichenden Ausdrücke sind, d. h. *wie* verglichen wird. Wenn sie als jeweiliges Ergebnis der beiden Formeln verglichen werden, dann ist (a) richtig. Wenn sie als Formeln, d. h.

[6] Z. B. sind zwei logische Operatoren gleich, wenn sie stets denselben *Wahrheitswert* liefern.

[7] Nach den Definitionen von '$:\Leftrightarrow$' und '$:=$' sind zwei Ausdrücke P und Q schon dann gleich, wenn nach der Ersetzung aller Vorkommen von A durch B sowohl in P als auch in Q die resultierenden Ausdrücke $\overline{P}$ und $\overline{Q}$ gleich sind.

als Zeichenfolgen, verglichen werden ist (a) falsch. Wenn die Ausdrücke in (b) als Zeichenfolgen verglichen werden, ist (b) dann richtig, wenn der Zwischenraum zwischen den einzelnen Zeichen nicht zählt. Wenn er aber zählt, ist (b) falsch.

Im Zusammenhang mit binären Relationen werden noch einige Verabredungen getroffen. Dazu seien '$\sim$', '$\simeq$', '$\lhd$', '$\rhd$', '$\unlhd$' und '$\unrhd$' Beispielsymbole für Relationen und '$=$' und '$\neq$' die Symbole für Gleichheit und Ungleichheit. Wenn dann nichts anderes gesagt wird gelte stets:

$$\begin{aligned} ((A \sim B) \ || \ (A = B)) \quad &\Leftrightarrow \quad (A \simeq B) \\ (A \lhd B) \quad &\Leftrightarrow \quad (B \rhd A) \\ (A \unlhd B) \quad &\Leftrightarrow \quad (B \unrhd A) \end{aligned} \tag{2.1}$$

Mit der Definition einer Relation der einen Seite ist damit automatisch auch die der anderen Seite erfolgt, mit der Ausnahme, dass man „$A \sim B$“ so nicht mit Hilfe von „$A \simeq B$“ definieren kann. Dies könnte man zwar mit Hilfe des Ansatzes

$$(A \sim B) \quad \Leftrightarrow \quad (A \simeq B) \ \&\& \ (A \neq B) \tag{2.2}$$

versuchen, aber die so definierte Relation '$\sim$' kann, muss aber nicht mit der in (2.1) übereinstimmen. Allerdings lässt sich (2.1) aus (2.2) ableiten und wenn „$(A = B) \Rightarrow (A \simeq B)$“ gilt, auch (2.2) aus (2.1). – Auf einen Beweis wird hier verzichtet.

Es sei noch angemerkt, dass wegen (2.2) die Definition von '$\Leftarrow$' in Abschnitt 2.1.1 auf Seite 11 überflüssig ist und wegen der Klammerregeln (siehe Unterabschnitt 2.4.2 auf Seite 22) auch alle Klammern in diesem Abschnitt 2.2. Die Prioritäten der Operatoren '$\lhd$', '$\rhd$', '$\unlhd$' und '$\unrhd$' unterscheiden sich normalerweise nicht; ebenso wenig die der Operatoren '$\sim$' und '$\simeq$', die aber durchaus verschieden von den Prioritäten von '$=$' und '$\neq$' sein können.

Als Beispielsymbol für binäre Operatoren wird '$\circ$' verwendet. Mit '$\circ$' zusammenhängende Verabredungen werden hier nicht getroffen.

2.3. Schlussregeln

Die Regeln zur Formulierung und Prüfung der Beweise müssen fest codiert werden. Sie sind quasi die Axiome von ASBA und sollten daher möglichst wenig voraussetzen. Dazu wird ein *Genzen-Kalkül* verwendet, so wie er in [1] Kapitel 1.4 beschrieben ist (siehe auch [34, 33]).

Ein Beweis in ASBA besteht aus n formalen Elementen V_i für $1 \leqslant i \leqslant n$ (den *Voraussetzungen*), einer Folge von zulässigen Transformationen, mit der neue formale Elemente generiert werden, bis alle m formalen Elemente F_j für $1 \leqslant j \leqslant m$ (die *Folgerungen*) abgeleitet sind. n kann auch gleich 0 sein, für m ist dass nicht sinnvoll.

Die zu beweisende Aussage (z. B. ein mathematischer Satz) kann dann auch folgendermaßen formuliert werden:[8)]

$$\frac{V_1 \mid V_2 \mid \ldots \mid V_n}{F_1 \mid F_2 \mid \ldots \mid F_m} \qquad \text{(formaler Satz)} \tag{FS}$$

Zum Beweis müssen aus den V_i durch zulässige Transformationen die F_i abgeleitet werden.

In diesem Abschnitt geht es um die zulässigen Transformationen, d. h. die allgemeingültigen Schlussregeln. Dazu gehören zunächst die Basisregeln. Dann aber auch alle aus den Basisregeln und den bis dahin allgemeingültigen Schlussregeln korrekt abgeleiteten neuen Schlussregeln. Die Schlussregeln haben die Form eines Formalen Satzes.

8) '|' steht für „und“ bzw. '&&', bindet aber wesentlich schwächer. *siehe auch* 2.1.1 auf Seite 11

2.3.1. Basisregeln

Gemäß [1] Kapitel 1.4 *Ein vollständiger aussagenlogischer Kalkül* werden sechs Basisregeln definiert. Zuvor werden aber noch einige Definition gebraucht. Dazu seien n, m, k und l natürliche Zahlen (auch 0), α, α_i, β und β_j formale Elemente, X, X_i, Y und Y_j Mengen von formalen Elementen und

$$\begin{array}{lll} X & := & X_1 \cup X_2 \cup ... \cup X_n \cup \{\alpha_1, \alpha_2, ..., \alpha_m\} \\ Y & := & Y_1 \cup Y_2 \cup ... \cup Y_k \cup \{\beta_1, \beta_2, ..., \beta_l\} \end{array}$$

X und Y können auch die leere Menge sein. Damit wird definiert:

$\alpha \vdash \beta \quad :\Leftrightarrow$ β ist mittels schrittweiser Anwendung *zulässiger Transformationen* (siehe weiter unten) aus α ableitbar. Sprechweise: Aus α ist β *ableitbar* oder *beweisbar*; kurz: „α *ableitbar* β“ bzw. „α *beweisbar* β“ – Es kann auch 'α' durch 'X' und/oder 'β' durch 'Y' ersetzt werden.

$\vdash \beta \quad :\Leftrightarrow \quad \varnothing \vdash \beta$ ('$\vdash$' kann dann auch ganz entfallen)

$X_1, X_2, ..., X_n, \alpha_1, \alpha_2, ..., \alpha_m \quad \vdash \quad Y_1, Y_2, ..., Y_n, \beta_1, \beta_2, ..., \beta_m \quad :\Leftrightarrow \quad X \vdash Y$

Eine *zulässige Transformation* ist die Anwendung einer *Substitution*[9] (siehe unten), einer *Basisregel* (siehe unten) oder einer davon abgeleiteten sonstigen *Schlussregel*, z. B. aus Unterabschnitt 2.3.3 auf Seite 17. Bei den Schlussregeln und der Substitution ($\leftarrow\!\!\prec$) soll das Komma stärker binden als '$\vdash$', '$\leftarrow\!\!\prec$' und '|', wobei '|' für „und“ bzw. '&&'[10] steht und schwächer bindet als '$\vdash$' und '$\leftarrow\!\!\prec$'.[11]

Zur der Auswahl der Basisregeln, der Formulierung und der Bezeichnungen wird auf [1, 34] zurückgegriffen. Wie in [34] steht 'E' für „-Einführung“ und 'B' für „-Beseitigung“ (oder „-Elimination“) von Operatoren.[12]

Im Folgenden seien α und β wieder stets formale Elemente und X und Y Mengen von formalen Elementen. Für die sechs Basisregeln werden dann nur noch die logischen Operatoren '$\neg$' und '$\wedge$' benötigt. Bei den weiteren Schlussregeln wird noch '$\rightarrow$' gemäß der Definition 2.5 auf Seite 26 verwendet.

9) siehe 2.3.2 auf der nächsten Seite

10) siehe Unterabschnitt 2.1.1 auf Seite 11

11) siehe Fußnote 3 von Tabelle 2.4 auf Seite 24

12) In der Monotonieregel wird hier, anders als in [1], „X, Y“ statt „Y , für $Y \supseteq X$“ genommen. Das ist gleichwertig, vermeidet aber den Zusatz „ , für $Y \supseteq X$“. Außerdem werden bei den Bezeichnungen „($\wedge$1)“ und „($\wedge$2)“ gemäß [34] durch „($\wedge$E)“ bzw. „($\wedge$B)“ ersetzt.

$$\frac{}{\alpha \vdash \alpha} \quad \text{(Anfangsregel)} \tag{AR}$$

$$\frac{X \vdash \alpha}{X, Y \vdash \alpha} \quad \text{(Monotonieregel)} \tag{MR}$$

$$\frac{X \vdash \alpha, \neg\alpha}{X \vdash \beta} \quad \text{(Einführung/Beseitigung der Negation Teil 1)} \tag{$\neg$1}$$

$$\frac{X, \alpha \vdash \beta \mid X, \neg\alpha \vdash \beta}{X \vdash \beta} \quad \text{(Einführung/Beseitigung der Negation Teil 2)} \tag{$\neg$2}$$

$$\frac{X \vdash \alpha, \beta}{X \vdash \alpha \wedge \beta} \quad \text{(Einführung der Konjunktion)} \tag{$\wedge$E}$$

$$\frac{X \vdash \alpha \wedge \beta}{X \vdash \alpha, \beta} \quad \text{(Beseitigung der Konjunktion)} \tag{$\wedge$B}$$

In einer Schlussregel werden die formalen Elemente[13)] über dem Querstrich als *Voraussetzungen* und die unter dem Querstrich als *Folgerung* der Regel bezeichnet. Eine Schlussregel steht für die Aussage, dass mit ihren Voraussetzungen auch auch ihre Folgerungen gelten. – Im Gegensatz zu den weiteren Schlussregeln werden die oben aufgelisteten Basisregeln nicht weiter hinterfragt, d. h. sie gelten quasi als Axiome.

2.3.2. Identitätsregeln

Die zulässigen Transformationen, d. h. die Anwendung der Schlussregeln, erfordern zulässige Substitutionen. Damit wird dem Gleichheits- oder Identitätszeichen ′=′ dann mittels Einführungs- und Beseitigungsregel eine Bedeutung verliehen.[14)] – Es seien α, β und γ wieder formale Elemente. Zunächst wird definiert:

$\Gamma(\alpha \leftarrow\!\!\prec \beta)$ $:=$ Das formale Element, dass man erhält, wenn in einem formalen Element Γ alle oder nur einige Vorkommen eines weiteren formalen Elements α durch ein drittes, mit α *vergleichbaren* formales Element β ersetzt werden. Gegebenenfalls muss noch die Auswahl der Ersetzungen angegeben werden, andernfalls werden alle Vorkommen ersetzt. Letzteres heißt dann eine *vollständige* Substitution. (2.3)

Bei obiger Definition der Substitution bleibt noch offen, unter welchen Voraussetzungen sie angewendet werden darf. Das soll hier nicht allgemein erläutert werden. In diesem Abschnitt genügt es, das nur vollständige Substitutionen verwendet werden. In dem Fall sind beliebige Substitutionen von Formeln durch *vergleichbare*[15)] erlaubt.

[13)] hier: Aussagen in einer formalen Form.
[14)] siehe [34]
[15)] sieheEnde von Unterabschnitt 2.1.1 auf Seite 11

Nun können die beiden Identitätsregeln definiert werden:

$$\frac{}{\alpha = \alpha} \qquad \text{(Einführung der Identität)} \tag{=E}$$

$$\frac{\alpha = \beta \mid \gamma}{\gamma(\alpha \leftarrow\!\!\prec \beta)} \qquad \text{(Beseitigung der Identität)}(=\mathrm{B}) \tag{=B}$$

Die Identitätsregeln werden hier eingeführt, um die Substitution zu rechtfertigen. Wie die Basisregeln gelten sie als Axiome, würden also eigentlich dazu gehören. Da sie aber nicht weiter verwendet werden, werden sie hier nicht zu den Basisregeln gezählt.

2.3.3. Weitere Schlussregeln

In [1] werden aus den Basisregeln mittels zulässiger Transformationen weitere Schlussregeln abgeleitet.[16)] Man vergleiche auch mit [34].

$$\frac{X, \neg\alpha \vdash \alpha}{X \vdash \alpha} \qquad \text{(Beseitigung der Negation; Indirekter Beweis)} \tag{¬3}$$

$$\frac{X, \neg\alpha \vdash \beta, \neg\beta}{X \vdash \alpha} \qquad \text{(Reductio ad absurdum)} \tag{¬4}$$

$$\frac{X, \alpha \vdash \beta}{X \vdash \alpha \rightarrow \beta} \qquad \text{(Einführung der Implikation)} \tag{→E}$$

$$\frac{X \vdash \alpha \rightarrow \beta}{X, \alpha \vdash \beta} \qquad \text{(Beseitigung der Implikation)} \tag{→B}$$

$$\frac{X \vdash \alpha \mid X, \alpha \vdash \beta}{X \vdash \beta} \qquad \text{(Schnittregel)} \tag{SR}$$

$$\frac{X \vdash \alpha \mid \alpha \rightarrow \beta}{X \vdash \beta} \qquad \text{(Abtrennungsregel–}\textit{Modus ponens}\text{)} \tag{TR}$$

Dabei werden zum Beweis der Schlussregeln in [1] folgende Basisregeln verwendet:

$$\begin{aligned}
\neg 3 &: \ \text{AR, MR, } \neg 2 \\
\neg 4 &: \ \text{AR, MR, } \neg 1, \neg 2 \\
\rightarrow\!\text{E} &: \ \text{AR, MR, } \neg 1, \neg 2, \wedge\text{E} \\
\rightarrow\!\text{B} &: \ \text{AR, MR, } \neg 1, \neg 2, \wedge\text{B} \\
\text{SR} &: \ \text{AR, MR, } \neg 1, \neg 2 \\
\text{TR} &: \ \text{AR, MR, } \neg 1, \neg 2, \wedge\text{E}
\end{aligned}$$

16) In [1] werden die Identitätsregeln zwar weder aufgeführt noch angewandt, ohne Substitution geht es aber nicht.

2.3.4. Beispiel einer Ableitung

Als Beispiel wird hier die Schnittregel aus den Basisregeln abgeleitet.[17] Dazu wird verabredet, dass in der Tabelle 2.1 auf der nächsten Seite der Inhalt der Zelle in der Zeile i und der Spalte (X_n) mit X_i bezeichnet wird. Zur kürzeren Darstellung wird statt auf die Spaltenüberschriften nur auf die dort notierten (X_n) verwiesen. Für die ausgefüllten Felder wird nun definiert:

$$R_i := \begin{cases} \text{- „Voraussetzung“ = Die Aussage } A_i \text{ ist eine Voraussetzung.} \\ \text{- „Folgerung“ = Die Aussage } A_i \text{ ist eine Folgerung.} \\ \text{- „Annahme“ = Die Aussage } A_i \text{ wird vorübergehend als zutreffend angenommen.} \\ \text{- } j \text{ = Verweis auf die Schlussregel } \overline{R}_j \text{ für ein } j < i. \\ \text{- Verweis (ohne Klammern) auf eine allgemeingültige Schlussregel.} \end{cases}$$

$S_i :=$ Die anzuwendende Substitution.

$\overline{R}_i :=$ Das Ergebnis der Substitution S_i auf die Schlussregel R_i

$Z_i :=$ Die Indizes j (mit $j < i$) als Verweise auf eine oder mehrere Aussagen A_j, welche zusammen genau die Voraussetzungen der Schnittregel $\overline{R}_i$erfüllen.

$A_i :=$ Folgerung(en) der Schlussregel $\overline{R}_i$ – auch in Form der Angabe von einem oder mehreren „Aj“ (mit $j < i$). In der Ergebniszeile kann hier auch die bewiesene Aussage als Schlussregel stehen.

$D_i :=$ die Indizes der A_j, von denen A_i abhängig ist.

Bis zur Zeile i hat man die folgende Schlussregel bewiesen:

$$\frac{A_{i_1} \mid A_{i_2} \ldots}{A_i} \quad \text{, für alle } i_j \in D_i$$

Sei nun

$$\Gamma_i := \begin{cases} \text{leer} & \text{für } R_i = \text{„Voraussetzung“} \\ \text{leer} & \text{für } R_i = \text{„Folgerung“} \\ \text{leer} & \text{für } R_i = \text{„Annahme“} \\ \overline{R}_j & \text{für } R_i = j \\ \text{die Schlussregel} & \text{für } R_i = \text{Verweis auf eine Schlussregel} \end{cases}$$

Damit gilt für die Einträge in einer Zeile i:

- Wenn Γ_i nicht leer ist, ist R_i eine allgemeingültige Schlussregel und es gilt:
  - WennS_i nicht leer ist, ist $R_i = \Gamma_i(S_i)$[18].
  - Sonst ist $R_i = \Gamma_i$.
- Wenn A_i nicht leer ist, ist $R_i = \dfrac{A_{z_1} \mid A_{z_2} \mid \ldots}{A_i}$ (alle $z_j \in Z_i$).
- Wenn A_i nicht leer ist, ist die Schlussregel $\dfrac{A_{d_1} \mid A_{d_2} \mid \ldots}{A_i}$ (alle $d_j \in D_i$) bewiesen.

Z_i und D_i dürfen dabei auch leer sein.

Die Erzeugung einer Tabelle analog zu 2.1 auf der nächsten Seite wird im folgenden beschrieben. Zellen, für die kein Inhalt angegeben wird, bleiben leer. Rückwärts-Referenzen auf schon ausgefüllte Zellinhalte sind jederzeit möglich. Das Eintragen der Zeilennummer i wird nicht extra erwähnt. – Die Tabelle und die Beschreibung sind so ausführlich, damit man daraus leicht ein Computerprogramm erstellen kann.

[17] Die Form der Tabelle ist angelehnt an [34] Kapitel 2.2.4 *Eine Beispielableitung*.
[18] sieheDefinition 2.3 auf Seite 16

Zeile (n)	**Regel** (R_n)	**Substitu-tionen** (S_n)	**erzeugte Regel** ($\overline{R}_n$)	**angewendet auf ...** (Z_n)	**Aussage** (A_n)	**Abhängig-keiten** (D_n)
1	Voraus-setzung				$X \vdash \alpha$	1
2	Voraus-setzung				$X, \alpha \vdash \beta$	2
3	Folge-rung				$X \vdash \beta$	3
4	MR		$\dfrac{X \vdash \alpha}{X, Y \vdash \alpha}$			
5	4	$Y \longleftarrow \neg\alpha$	$\dfrac{X \vdash \alpha}{X, \neg\alpha \vdash \alpha}$	1	$X, \neg\alpha \vdash \alpha$	1
6	AR		$\dfrac{}{\alpha \vdash \alpha}$			
7	6	$\alpha \longleftarrow \neg\alpha$	$\dfrac{}{\neg\alpha \vdash \neg\alpha}$		$\neg\alpha \vdash \neg\alpha$	
8	4	$\alpha \longleftarrow \neg\alpha$	$\dfrac{X \vdash \neg\alpha}{X, Y \vdash \neg\alpha}$			
9	8	$X \longleftarrow \neg\alpha$	$\dfrac{\neg\alpha \vdash \neg\alpha}{Y, \neg\alpha \vdash \neg\alpha}$			
10	9	$Y \longleftarrow X$	$\dfrac{\neg\alpha \vdash \neg\alpha}{X, \neg\alpha \vdash \neg\alpha}$	7	$X, \neg\alpha \vdash \neg\alpha$	
11	$\neg 1$		$\dfrac{X \vdash \alpha, \neg\alpha}{X \vdash \beta}$			
12	11	$X \longleftarrow X, \neg\alpha$	$\dfrac{X, \neg\alpha \vdash \alpha, \neg\alpha}{X, \neg\alpha \vdash \beta}$	5, 10	$X, \neg\alpha \vdash \beta$	1
13	$\neg 2$		$\dfrac{X, \alpha \vdash \beta \mid X, \neg\alpha \vdash \beta}{X \vdash \beta}$	2, 12	$X \vdash \beta$	1, 2
14	AR, MR, $\neg 1$, $\neg 2$		$\dfrac{A_1 \mid A_2}{A_3}$		$\dfrac{X \vdash \alpha \mid X, \alpha \vdash \beta}{X \vdash \beta}$	

Tabelle 2.1.: Ableitung der Schnittregel aus den Basisregeln

1. In den ersten Zeilen werden zuerst Voraussetzungen, dann zu beweisende Folgerungen und schließlich Annahmen aufgeführt.[19)] Jede der drei Gruppen kann auch leer sein und es ist auch möglich, die Zeilen an anderen Stellen der Tabelle anzugeben, spätestens aber, wenn darauf verwiesen wird. Für jede Voraussetzung, Folgerung und Annahme gibt es eine Zeile:
   a) $R_i =$ „Voraussetzung“, „Folgerung“ oder „Annahme“.
   b) $A_i =$ Die aktuelle Voraussetzung, Folgerung oder Annahme.
   c) $D_i = i$ (ein Verweis auf A_i).
2. In den nächsten Zeilen werden die Beweisschritte aufgeführt, für jeden Schritt eine Zeile.

   Zunächst kann R_i kann auf zwei Arten erzeugt werden:
   a) i. $R_i = j$, wenn die schon bewiesene Schlussregel $\overline{R}_j$ (mit $j < i$) angewendet werden soll.
      ii. $S_i =$ Die auf die Schlussregel R_i anzuwendende Substitution.
      iii. $\overline{R}_i =$ Das Ergebnis der Substitution S_i auf die Schlussregel R_i. – Sie wird hier als *interne* Schlussregel bezeichnet.

19) Die Angabe ist dann erforderlich, wenn darauf verwiesen wird. Durch die Auflistung hat man aber einen vollständigen Überblick über die Voraussetzungen und Folgerungen eines Beweises und die Zwischenannahmen. Auf jede nötige Voraussetzung und jede verwendete Zwischenannahme wird in der Spalte (Z_n) mindestens einmal verwiesen, so dass sie auch aufgeführt werden müssen. Die Angabe der Folgerungen erleichtert die Erstellung einer *Ergebniszeile* (siehePunkt 3).

oder

b) i. R_i = Verweis auf eine allgemeingültige Schlussregel.

ii. $\overline{R}_i$ = Die Schlussregel, auf die verwiesen wird. – Sie wird hier als *externe* Schlussregel bezeichnet.

Man beachte, dass die Schlussregel $\overline{R}_i$, stets allgemeingültig ist, da sie ausschließlich aus allgemeingültigen Schlussregeln mittels Substitutionen abgeleitet worden ist. Daher gibt es auch keine Beschränkung weiterer Substitutionen durch irgendwelche Abhängigkeiten.

Nun kann die Zeile beendet werden, oder es geht weiter mit:

c) Z_n = Die Indizes aller A_j (mit $j < i$), die eine Voraussetzung der Schlussregel $\overline{R}_i$ sind, möglichst in der verwendeten Reihenfolge. – Für jedes angegebene j werden noch die Abhängigkeiten D_j den Abhängigkeiten D_i hinzugefügt.

d) A_i = Folgerung(en) der Schlussregel $\overline{R}_i$. – Wenn diese Folgerungen schon als Aussagen A_j (mit $j < i$) vorhanden sind, können diese auch einfach mit „A_j" eingetragen werden. Damit werden die Zusammenhänge und der Abschluss des Beweises besser ersichtlich.

e) D_i = Die Verweise wurden schon in (2c) eingetragen.[20)]

Der Beweis muss so lange fortgeführt werden, bis alle Folgerungen als Aussagen in der Spalte (A_n) erschienen und dort jeweils nur von Voraussetzungen abhängig sind.

3. In einer *Ergebniszeile*, die dann die letzte ist, kann noch die bewiesene Behauptung in Form einer Schlussregel formuliert und in einer passenden Spalte notiert werden. Zusätzlich können dort auch noch alle verwendeten Schlussregeln gesammelt werden. Dies kann z. B. folgendermaßen geschehen:

   a) (R_n) = Verweise auf alle verwendeten externen Schlussregeln.

   b) $(\overline{R}_n)$ = Die bewiesene Behauptung als Schlussregeln, wobei alle A_i, die Voraussetzungen sind, als Voraussetzung und alle A_j, die Folgerungen sind, als Folgerung eingesetzt werden, jeweils in der Form „A_i" bzw. „A_j". Das ergibt dann:

$$\frac{A_{i_1} \mid A_{i_2} \mid \ldots}{A_{j_1} \mid A_{j_2} \mid \ldots}$$

   c) $(A_n) = \overline{R}_i$, wobei die Voraussetzungen und Folgerungen aufgelöst werden.

   d) (D_n) = Die Vereinigung aller Abhängigkeiten der Folgerungen, vermindert um die Voraussetzungen. – Wenn das Feld dabei nicht leer bleibt, ist der Beweis missglückt!

Ein weiteres Beispiel in der Tabelle 2.2 auf der nächsten Seite soll verdeutlichen, wie Abhängigkeiten von Zwischenannahmen wieder beseitigt werden können.[21)]

> > > Beispiel mit Reduzierung von Abhängigkeiten vervollständigen < < <

20) Wenn D_n leer ist, dann ist A_n allgemeingültig.
21) siehe [34], Kapitel 2.2.4 *Eine Beispielableitung*

Zeile (n)	**Regel** (R_n)	**Substitutionen** (S_n)	**erzeugte Regel** ($\overline{R}_n$)	**angewendet auf ...** (Z_n)	**Aussage** (A_n)	**Abhängigkeiten** (D_n)
1	Folgerung				$(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)$	1
2	Annahme				$\alpha \to \beta$	2
3	Annahme				$\neg\beta$	3
4	Annahme				α	4
5	$\to$B		$\frac{X \vdash \alpha \to \beta}{X, \alpha \vdash \beta}$			
6	-1	$X \leftarrowtail \emptyset$	$\frac{\alpha \to \beta}{\alpha \vdash \beta}$	2	$\alpha \vdash \beta$	2
7	SR		$\frac{X \vdash \alpha \mid X, \alpha \vdash \beta}{X \vdash \beta}$			
8	-1	$X \leftarrowtail \emptyset$	$\frac{\alpha \mid \alpha \vdash \beta}{\beta}$	4, 6	β	4, 6
9	$\neg$1		$\frac{X \vdash \alpha, \neg\alpha}{X \vdash \beta}$			
10	-1	$X \leftarrowtail \emptyset$	$\frac{\alpha, \neg\alpha}{\beta}$			
11	-1	$\alpha \leftrightarrows \beta$	$\frac{\beta, \neg\beta}{\alpha}$			
12	-1	$\alpha \leftarrowtail \neg\alpha$	$\frac{\beta, \neg\beta}{\neg\alpha}$	8, 3	$\neg\alpha$	2, 3, 4
13	$\to$E		$\frac{X, \alpha \vdash \beta}{X \vdash \alpha \to \beta}$			
14	-1	$X \leftarrowtail \emptyset$	$\frac{\alpha \vdash \beta}{\alpha \to \beta}$			
15	-1	$\alpha \leftrightarrows \beta$	$\frac{\beta \vdash \alpha}{\beta \to \alpha}$			
16	-1	$\alpha \leftarrowtail \neg\alpha$	$\frac{\beta \vdash \neg\alpha}{\beta \to \neg\alpha}$			$\to$E
17	-1	$\beta \leftarrowtail \neg\beta$	$\frac{\neg\beta \vdash \neg\alpha}{\neg\beta \to \neg\alpha}$	3, 12, ???	$\neg\beta \to \neg\alpha$	2, 3, 4, ???
18	$\to$E+1	$\alpha \leftarrowtail \gamma$	$\frac{\gamma \vdash \beta}{\gamma \to \beta}$			$\to$E
19	-1	$\beta \leftarrowtail \delta$	$\frac{\gamma \vdash \delta}{\gamma \to \delta}$			$\to$E
20	-1	$\gamma \leftarrowtail \alpha \to \beta$	$\frac{\alpha \to \beta \vdash \delta}{(\alpha \to \beta) \to \delta}$			$\to$E
21	-1	$\delta \leftarrowtail \neg\beta \to \neg\alpha$	$\frac{\alpha \to \beta \vdash \neg\beta \to \neg\alpha}{(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)}$	2, 17	$(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)$	2, 3, 4, ???
22	$\to$E, $\to$B, SR		$\frac{}{A_1}$		$\frac{}{(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)}$	

Tabelle 2.2.: Ableitung der Kontraposition aus allgemeingültigen Schlussregeln

2.4. Aussagenlogik

2.4.1. Konstante und Operatoren

Die Tabelle 2.3 auf der nächsten Seite[22)] definiert für die zweiwertige Logik Konstanten- und Operatorsymbole über die Wahrheitswerte ihrer Anwendung. So ergeben sich, abhängig von den Wahrheitswerten der Operanden A und B[23)], die in der Tabelle angegebenen Wahrheitswerte für die Operationen. Die mit 0, 1 und 2 benannten Spalten werden jeweils nur für die 0-, 1- und 2-stelligen Operatoren, d. h. für die Konstanten, die unären und die binären Operatoren ausgefüllt. Dabei werden die Konstanten als 0-stellige Operatoren angesehen. Hat der Inhalt einer Zelle keine Relevanz, steht dort ein Minuszeichen, ist kein Wert bekannt, so bleibt sie leer.

Für einige Junktoren, Namen und Sprechweisen sind auch Alternativen angegeben. Die durchgestrichenen (d. h. negierten) Symbole sind ungebräuchlich und nur aus formalen Gründen aufgeführt. Wenn für eine bestimmte Kombination von Wahrheitswerten mehr als eine Zeile angegeben ist, so sind die zugehörigen Operationen in der zweiwertigen Aussagenlogik alle gleich. Bei der formalen Definition wird aber keine Zweiwertigkeit vorausgesetzt, so dass je nach Definition die Operationen verschiedene Ergebnisse liefern können.

Um vollständig zu sein, d. h. alle 22 möglichen Kombinationen von Wahrheitswerten für höchstens zwei Variable zu berücksichtigen, enthält die Tabelle auch viele ungebräuchliche Junktoren und Operationen. Die Zeilen mit den Klammern und den gebräuchlichsten Junktoren sind in der Tabelle grau hinterlegt. Hellgrau hinterlegt sind Zeilen mit weniger gebräuchlichen Junktoren. Die restlichen Operationen sind uninteressant und brauchen daher keine Priorität.

2.4.2. Klammerregeln

Zur Klammerersparnis werden die üblichen Regeln verwendet, d. h. dass Operatoren mit höherer Priorität stärker binden, als solche mit niedrigerer Priorität.

Für die Operatoren derselben Priorität gilt Rechtsklammerung[24)]. Im Folgenden wird nur noch ein Teil der logischen Operatoren aus der Tabelle 2.3 auf der nächsten Seite und der metasprachlichen Operatoren aus Unterabschnitt 2.1.1 auf Seite 11 berücksichtigt. Diese werden in der Tabelle 2.4 auf Seite 24 mit abnehmender Priorität aufgelistet.

Die Prioritäten der logischen Operatoren wurden aus [1] Kapitel 1.1 Seite 5 entnommen und ergänzt und die der metasprachlichen Operatoren daran angeglichen. Wie üblich bindet ein Operator *stärker* als jeder andere mit einer niedrigeren Priorität und *schwächer* als jeder andere mit höherer Priorität.

2.4.3. Formalisierung

Da sie die Grundlage – quasi das Fundament – des mathematischen Inhalts von ASBA sind, müssen die Axiome, Sätze, Beweise, usw. der Aussagenlogik in streng formaler Form vorliegen. Die Formalisierung stützt sich auf [28]; siehe auch [21, 24]. Da Computerprogramme mit der *Polnischen Notation*[25)]

22) Die Tabelle basiert auf den Wahrheitstafeln in [27] Kapitel 2.2 und [1] Kapitel 1.1 Seite 3.

23) Im Gegensatz zu Paragraph 2.4.3.1 auf Seite 24 können A und B hier beliebige Aussagen – auch Formeln – sein.

24) Unäre Operatoren stehen hier stets links *vor* dem Operanden, so dass es nur Rechtsklammerung geben kann. Zur Rechtsklammerung bei binären Operationen ein Zitat aus [1] Kapitel 1.1 Seite 5: „Diese hat gegenüber Linksklammerung Vorteile bei der Niederschrift von Tautologien in →, [...]“

25) Bei der *Polnischen Notation* wird eine zweistellige Operation $(A \circ B)$ dargestellt als $\circ AB$. Eine Zwischenstufe ist $\circ(A, B)$, bei der noch die redundanten Gliederungszeichen Komma und Klammern – auch andere als die runden – hinzukommen, so dass die Operationen optisch besser getrennt und dadurch für Menschen besser lesbar werden. Durch einfaches Weglassen der Gliederungszeichen ergibt sich dann die Polnische Notation.

A	-	W	F	W	W	F	F	-	Aussage A	-
B	-	-	-	W	F	W	F	-	Aussage B	-
Junktor[1]	**0**	**1**		**2**				**Name**	**Sprechweise**[2]	**Prio**
$\top$	W	-	-	-	-	-	-	Verum	Wahr	-
$\bot$	F	-	-	-	-	-	-	Falsum	Falsch	-
	-	W	W	-	-	-	-			-
$(\ldots)$	-	W	F	-	-	-	-	Klammerung[3]	A ist geklammert	6[4]
$\neg$	-	F	W	-	-	-	-	Negation	Nicht A	5[5]
	-	F	F	-	-	-	-			-
	-	-	-	W	W	W	W	Tautologie		-
$\vee$	-	-	-	W	W	W	F	Disjunktion; Adjunktion; Alternative	A oder B	3
$\leftarrow \Leftarrow \subset$	-	-	-	W	W	F	W	Replikation; Konversion; konverse Implikation	A folgt aus B	2
$\rfloor$	-	-	-	W	W	F	F	Präpendenz	Identität von A	-
$\rightarrow \Rightarrow \supset$	-	-	-	W	F	W	W	Implikation; Subjunktion; Konditional	Wenn A so B; Aus A folgt B; A nur dann wenn B	2
$\lfloor$	-	-	-	W	F	W	F	Postpendenz	Identität von B	-
$\leftrightarrow \Leftrightarrow$	-	-	-	W	F	F	W	Äquivalenz; Bijunktion; Bikonditional	A genau dann wenn B; A dann und nur dann wenn B	1
$\wedge$ & $\cdot$	-	-	-	W	F	F	F	Konjunktion	A und B; Sowohl A als auch B	4
$\uparrow \barwedge \mid$	-	-	-	F	W	W	W	NAND; Unverträglichkeit; Sheffer-Funktion	Nicht zugleich A und B	4
$+ \dot{\vee} \veebar \oplus$	-	-	-	F	W	W	F	XOR; Antivalenz; ausschließende Disjunktion	Entweder A oder B	3
$\nleftrightarrow \nLeftrightarrow \not\equiv$	-	-	-	"	"	"	"	Kontravalenz		-
$\lceil$	-	-	-	F	W	F	W	Postnonpendenz	Negation von B	-
$\nrightarrow \nRightarrow \not\supset$	-	-	-	F	W	F	F	Postsektion		-
$\rceil$	-	-	-	F	F	W	W	Pränonpendenz	Negation von A	-
$\nleftarrow \nLeftarrow \not\subset$	-	-	-	F	F	W	F	Präsektion		-
$\downarrow \overline{\vee}$	-	-	-	F	F	F	W	NOR; Nihilation; Peirce-Funktion	Weder A noch B	3
	-	-	-	F	F	F	F	Kontradiktion		-

[1] *Operatorsymbole*. Sie stehen meistens für die Operatoren selbst. Der Einfachheit halber werden auch die beiden Konstanten $\top$ und $\bot$ als Junktoren bzw. Operatoren bezeichnet.
Die Operatoren '$\subset$', '$\supset$', '$\not\subset$' und '$\not\supset$' haben hier nicht die Bedeutung der entsprechenden Operatoren der Mengenlehre und dürfen nicht damit verwechselt werden; entsprechendes gilt für '$+$' und '$\cdot$' mit Addition und Multiplikation.

[2] Ist eine Zelle in dieser Spalte leer, so ist die zugehörige Zeile nur vorhanden, um alle binären Operationen aufzuführen.

[3] Klammerung ist genau genommen kein Operator und wird nicht nur bei logischen, sondern auch bei anderen Ausdrücken verwendet.

[4] Die Priorität der Klammern ist größer als die aller Operatoren.

[5] Die Priorität der unären Operatoren muss größer sein als die aller mehrwertigen, also auch der binären Operatoren. Wenn alle unären Operatoren auf derselben Seite des Operanden stehen, brauchen sie eigentlich keine Priorität, da die Auswertung nur von innen (dem Operanden) nach außen erfolgen kann. Nur wenn es sowohl links-, als auch rechtsseitige unäre Operatoren gibt, muss man für diese Prioritäten definieren.

Tabelle 2.3.: Definition von aussagenlogischen Symbolen.

Klammern	$($ $)$
Unäre logische Operatoren	$\neg$
Binäre logische Operatoren	$\wedge$ $\cdot$ $\uparrow$ $\vee$ $+$ $\downarrow$ $\leftarrow$ $\rightarrow$ $\leftrightarrow$
Mit Gleichheit verwandte Symbole; ihre Prioritäten untereinander sind nicht eindeutig und bleiben daher undefiniert.	$=$ $\neq$ $\equiv$ $\not\equiv$
Ableitungsrelation[1] Substitution[1] Definition	$\vdash$ $\leftarrowtail$ $:=$
Metasprachliche Operatoren[2]	&& || $\Leftarrow$ $\Rightarrow$ $\Leftrightarrow$ |
Metadefinition	$:\Leftrightarrow$
Strukturelemente der natürlichen Sprache, z. B. Satzzeichen[3]	. , ; usw.

[1] siehe Unterabschnitt 2.3.1 auf Seite 15
[2] Für '|' siehe Unterabschnitt 2.1.1 auf Seite 11
[3] Innerhalb von formalen Elementen können Satzzeichen eine andere Priorität haben. Siehe z. B. Unterabschnitt 2.3.1 auf Seite 15.

Tabelle 2.4.: Prioritäten von Operatoren in abnehmender Reihenfolge

besser umgehen können und Klammern dort überflüssig sind, werden viele Formeln auch in die Polnische Notation überführt.

2.4.3.1. Bausteine der aussagenlogischen Sprache

Zur Einteilung der aussagenlogischen Junktoren werden die folgenden Mengen definiert:

$$
\begin{array}{lll l}
\mathbb{N}_0 & := & & \text{Menge der } \textit{natürlichen Zahlen} \text{ einschließlich } 0 \\
\mathcal{C} & := & \{\top, \bot\} & \text{, Menge der } \textit{aussagenlogischen Konstanten} \\
\mathcal{U} & := & \{\neg\} & \text{, Menge der } \textit{unären aussagenlogischen Operatoren} \\
\mathcal{B} & := & \{\wedge, \vee, \rightarrow, \leftrightarrow, \leftarrow, \uparrow, \downarrow, \cdot, +\} & \text{, Menge der } \textit{binären aussagenlogischen Operatoren}
\end{array}
$$

Damit sind alle in der Tabelle 2.3 auf der vorherigen Seite verwendeten wesentlichen Konstanten und Operatoren[26] erfasst und es können die folgende Mengen definiert werden:

$$
\begin{array}{lll l}
\mathcal{V} & := & \{p_n \mid n \in \mathbb{N}_0\} & \text{, Menge der } \textit{atomaren Formeln} \\
\mathcal{J} & := & \mathcal{C} \cup \mathcal{U} \cup \mathcal{B} & \text{, Menge der } \textit{Junktoren} \text{ bzw. } \textit{Operatoren}\text{; auch } \textit{logische Signatur} \\
\mathcal{A} & := & \mathcal{V} \cup \mathcal{J} & \text{, } \textit{Alphabet der aussagenlogischen Sprache (für } \mathcal{J}\textit{)} \\
\mathcal{J}_x & \subseteq & \mathcal{J} & \text{, eine Teilmenge von } \mathcal{J} \text{ für eine Indexvariable } x \\
\mathcal{A}_x & := & \mathcal{V} \cup \mathcal{J}_x & \text{, Alphabet der aussagenlogischen Sprache } \textit{für } \mathcal{J}_x
\end{array}
\tag{2.4}
$$

[26] Jeweils nur die ersten der grau hinterlegten Zeilen sowie '·'.

Für Elemente aus $\mathcal{V}$ werden hier normalerweise die großen lateinischen Buchstaben A, B, C, usw. verwendet. Die Elemente aus $\mathcal{V}$ (atomare Formeln) werden auch *Satzbuchstaben* oder kurz *Atome* genannt.

2.4.3.2. Aussagenlogische Formeln

Neben dem Alphabet $\mathcal{A}$ bzw. . $\mathcal{A}_x$ werden noch Klammern als Gliederungszeichen verwendet. Damit können nun rekursiv für jede Teilmenge $\mathcal{J}_x$ von $\mathcal{J}$ zwei Mengen von Formeln definiert werden:

$\mathcal{F}_x$ sei die Menge der auf folgende Weise definierten *aussagenlogischen Formeln mit Klammerung*:

$$\begin{array}{rcrl} & & \mathcal{V} \subset \mathcal{F}_x & \\ & & \mathcal{J}_x \cap \mathcal{C} \subset \mathcal{F}_x & \\ A \in \mathcal{F}_x & \Rightarrow & (\circ A) \in \mathcal{F}_x & \text{, für } \circ \in \mathcal{U} \cap \mathcal{J}_x \\ A, B \in \mathcal{F}_x & \Rightarrow & (A \circ B) \in \mathcal{F}_x & \text{, für } \circ \in \mathcal{B} \cap \mathcal{J}_x \end{array}$$

Nur die auf diese Weise konstruierten Formeln sind Elemente von $\mathcal{F}_x$.
Für $\mathcal{J} = \mathcal{J}_x$ sei noch $\mathcal{F} := \mathcal{F}_x$.

$\mathcal{F}_x^p$ sei die Menge der auf folgende Weise definierten aussagenlogischen Formeln in *Polnischer Notation*:

$$\begin{array}{rcrl} & & \mathcal{V} \subset \mathcal{F}_x^p & \\ & & \mathcal{J}_x \cap \mathcal{C} \subset \mathcal{F}_x^p & \\ A \in \mathcal{F}_x^p & \Rightarrow & (\circ A) \in \mathcal{F}_x^p & \text{, für } \circ \in \mathcal{U} \cap \mathcal{J}_x \\ A, B \in \mathcal{F}_x^p & \Rightarrow & (A \circ B) \in \mathcal{F}_x^p & \text{, für } \circ \in \mathcal{B} \cap \mathcal{J}_x \end{array}$$

Nur die auf diese Weise konstruierten Formeln sind Elemente von $\mathcal{F}_x^p$.
Für $\mathcal{J} = \mathcal{J}_x$ sei noch $\mathcal{F}^p := \mathcal{F}_x^p$.

Wie man leicht sieht, gilt:

$$\mathcal{J}_x \subset \mathcal{J}_y \subset \mathcal{J} \Rightarrow \begin{cases} \mathcal{A}_x \subset \mathcal{A}_y \subseteq \mathcal{A} \\ \mathcal{F}_x \subset \mathcal{F}_y \subseteq \mathcal{F} \\ \mathcal{F}_x^p \subset \mathcal{F}_y^p \subseteq \mathcal{F}^p \end{cases}$$

Durch Anwendung der Klammerregeln von Paragraph 2.4.3.1 auf der vorherigen Seite lassen sich in der Regel noch viele Klammern der Formeln aus $\mathcal{F}_x$ einsparen. Die Formeln aus $\mathcal{F}_x^p$ sind frei von Klammern. Die Namen der Operatoren finden sich in der Tabelle 2.3 auf Seite 23. Für aussagenlogische Formeln, d. h. von Elementen aus $\mathcal{F}$ bzw. $\mathcal{F}^p$, werden hier normalerweise die kleinen griechischen Buchstaben α, β, γ, usw. verwendet. Sie können dabei auch atomare Formeln bezeichnen (siehe (2.4)).

2.4.4. Definition aussagenlogische Operatoren durch andere

Im folgenden gelte für zwei aussagenlogische Formeln α und β:

$\alpha = \beta \quad :\Leftrightarrow \quad$ α und β stimmen als Zeichenkette überein.

$\alpha \equiv \beta \quad :\Leftrightarrow \quad$ α und β können mit Hilfe erlaubter Substitutionen und geltender Axiome – siehe Unterabschnitt 2.4.5 auf Seite 27 – ineinander überführt werden.

Es werden verschiedene Teilmengen von $\mathcal{J}$ – logische Signaturen – eingeführt, die jeweils ausreichen alle anderen Elemente aus $\mathcal{J}$ zu definieren:

$$\begin{array}{lcll}
\mathcal{J}_{\text{bool}} & := & \{\neg, \wedge, \vee\} & \text{(Boolsche Signatur)} \\
\mathcal{J}_{\text{and}} & := & \{\neg, \wedge\} & \\
\mathcal{J}_{\text{or}} & := & \{\neg, \vee\} & \\
\mathcal{J}_{\text{imp}} & := & \{\neg, \rightarrow\} & \\
\mathcal{J}_{\text{rep}} & := & \{\neg, \leftarrow\} & \\
\mathcal{J}_{\text{nand}} & := & \{\uparrow\} & \\
\mathcal{J}_{\text{nor}} & := & \{\downarrow\} &
\end{array}$$

Im Folgenden stehen jeweils links die Formeln in üblicher Schreibweise vollständig geklammert und rechts in Polnischer Notation (ohne Klammern). Ferner seien α und β beliebige, nicht notwendig verschiedene Formeln aus der passenden Menge $\mathcal{F}_x$ bzw. der um die mit Hilfe der Definitionen erweiterten Formelmenge.

Ausgehend von den Operatoren aus der Boolschen Signatur $\mathcal{J}_{\text{bool}}$ werden die restlichen Operatoren aus $\mathcal{J}$ definiert. Die Definitionen sind in zwei Gruppen eingeteilt, und zwar die mit den Operatoren aus $\mathcal{J}_{\text{and}}$:

$$\begin{array}{rcl@{\qquad}rcll}
(\alpha \rightarrow \beta) & := & (\neg(\alpha \wedge (\neg\beta))) & \rightarrow\alpha\beta & := & \neg \wedge \alpha\neg\beta & (2.5) \\
(\alpha \leftarrow \beta) & := & (\neg(\beta \wedge (\neg\alpha))) & \leftarrow\beta\alpha & := & \neg \wedge \beta\neg\alpha & (2.6) \\
(\alpha \leftrightarrow \beta) & := & ((\alpha \rightarrow \beta) \wedge (\alpha \leftarrow \beta)) & \leftrightarrow\alpha\beta & := & \wedge \rightarrow\alpha\beta \leftarrow\alpha\beta & \\
\bot & := & (p_0 \wedge (\neg p_0)) & \bot & := & \wedge p_0 \neg p_0 & \\
(\alpha \cdot \beta) & := & (\alpha \wedge \beta) & \cdot\alpha\beta & := & \wedge\alpha\beta & \\
(\alpha \uparrow \beta) & := & (\neg(\alpha \wedge \beta)) & \uparrow\alpha\beta & := & \neg \wedge \alpha\beta & (2.7)
\end{array}$$

und die mit den Operatoren aus $\mathcal{J}_{\text{or}}$:

$$\begin{array}{rcl@{\qquad}rcll}
(\alpha \downarrow \beta) & := & (\neg(\alpha \vee \beta)) & \downarrow\alpha\beta & := & \neg \vee \alpha\beta & (2.8) \\
(\alpha + \beta) & := & ((\alpha \vee \beta) \wedge (\neg(\alpha \wedge \beta))) & +\alpha\beta & := & \wedge \vee \alpha\beta\neg \wedge \alpha\beta & \\
\top & := & (p_0 \vee (\neg p_0)) & \top & := & \vee p_0 \neg p_0 &
\end{array}$$

Ist '$\vee$' oder '$\wedge$' nicht vorgegeben, d. h. wird von den Elementen aus $\mathcal{J}_{\text{and}}$ bzw. $\mathcal{J}_{\text{or}}$ statt von denen aus $\mathcal{J}_{\text{bool}}$ ausgegangen, so muss man den fehlenden Operator mittels der passenden der beiden folgenden Definitionen einführen:

$$\begin{array}{rcl@{\qquad}rcl}
(\alpha \vee \beta) & := & (\neg((\neg\alpha) \wedge (\neg\beta))) & \vee\alpha\beta & := & \neg \wedge \neg\alpha\neg\beta \\
(\alpha \wedge \beta) & := & (\neg((\neg\alpha) \vee (\neg\beta))) & \wedge\alpha\beta & := & \neg \vee \neg\alpha\neg\beta
\end{array}$$

Nun sind wieder alle Operatoren definiert.

Entsprechend wird bei Vorgabe von $\mathcal{J}_{\text{imp}}$ bzw. $\mathcal{J}_{\text{rep}}$ die passende der beiden folgenden Definitionen ausgewählt:

$$\begin{array}{rcl@{\qquad}rcl}
(\alpha \vee \beta) & := & ((\neg\alpha) \rightarrow \beta) & \vee\alpha\beta & := & \rightarrow\neg\alpha\beta \\
(\alpha \wedge \beta) & := & (\neg((\neg\beta) \leftarrow \alpha)) & \wedge\alpha\beta & := & \neg\leftarrow\neg\beta\alpha
\end{array}$$

woraufhin dann (2.5) bzw. (2.6) als Gleichung nachzuweisen ist. Da aus (2.6) durch Vertauschung der Variablen unmittelbar

$$(\alpha \leftarrow \beta) \;\equiv\; (\beta \rightarrow \alpha) \qquad\qquad \leftarrow\alpha\beta \;\equiv\; \rightarrow\beta\alpha$$

folgt, vermindert sich der Aufwand dazu erheblich.

Bei Vorgabe von $\mathcal{J}_{\text{nand}}$ bzw. $\mathcal{J}_{\text{nor}}$ schließlich werden die passenden Definition aus

$$\begin{aligned} (\neg\alpha) &:= (\alpha \downarrow \alpha) & \neg\alpha &:= \downarrow\alpha\alpha \\ (\neg\alpha) &:= (\alpha \uparrow \alpha) & \neg\alpha &:= \uparrow\alpha\alpha \end{aligned}$$

und, da '$\neg$' jetzt definiert ist, aus

$$\begin{aligned} (\alpha \vee \beta) &:= (\neg(\alpha \downarrow \beta)) & \vee\alpha\beta &:= \neg\downarrow\alpha\beta \\ (\alpha \wedge \beta) &:= (\neg(\alpha \uparrow \beta)) & \wedge\alpha\beta &:= \neg\uparrow\alpha\beta \end{aligned} \tag{2.9}$$

ausgewählt und es ist (2.7) bzw. (2.8) als Gleichung nachzuweisen.

Abschließend ist noch nachzuweisen, dass mit Hilfe der jeweils passenden der Definitionen (2.5) bis (2.9), ausgehend vom jeweils passenden $\mathcal{F}_x$, genau die gesamte Formelmenge $\mathcal{F}$ erzeugt werden kann.

2.4.5. Aussagenlogisches Axiomensystem

Ausgehend von der logischen Signatur $\mathcal{J}_{\text{and}} = \{\neg, \wedge\}$ und der Definition 2.5 auf der vorherigen Seite von '$\rightarrow$' werden die folgenden vier logischen Axiome definiert:

$$\begin{aligned} &(\alpha \rightarrow \beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \gamma) && \rightarrow\rightarrow\alpha\rightarrow\beta\gamma\rightarrow\rightarrow\alpha\beta\rightarrow\alpha\gamma \\ &\alpha \rightarrow \beta \rightarrow \alpha \wedge \beta && \rightarrow\alpha\rightarrow\beta\wedge\alpha\beta \\ &\alpha \wedge \beta \rightarrow \alpha\,; \quad \alpha \wedge \beta \rightarrow \beta && \rightarrow\wedge\alpha\beta\alpha\,; \quad \rightarrow\wedge\alpha\beta\beta \\ &(\alpha \rightarrow \neg\beta) \rightarrow (\beta \rightarrow \neg\alpha) && \rightarrow\rightarrow\alpha\neg\beta\rightarrow\beta\neg\alpha \end{aligned}$$

> > > Aussagenlogik weiter bearbeiten. < < <

2.5. Prädikatenlogik

> > > Prädikatenlogik bearbeiten. < < <

2.6. Mengenlehre

> > > Mengenlehre bearbeiten. < < <

3. Design

Diese Projekt soll Open Source sein. Daher gilt für die Dokumente die *GNU Free Documentation License (FDL)* und für die Software die *GNU Affero General Public License (APGL)*. Die *GNU General Public License (GPL)* reicht für die Software nicht, da das Programm auch mittels eines Servers betrieben werden kann und soll. Damit das Projekt gegebenenfalls durch verschiedene Entwickler gleichzeitig bearbeitet werden kann und wegen des Konfigurationsmanagements wurde es als ein GitHub Projekt erstellt (siehe [20]).

Wenn die Lizenzen nicht mitgeliefert wurden, können sie unter `http://www.gnu.org/licenses/` gefunden werden.

3.1. Anforderungen

Die Anforderungen ergeben sich zunächst aus den Zielen in Abschnitt 1.3 auf Seite 6. Die beiden Ziele 1 *Daten* und 15 *Lizenz* sind für die Entwicklung von ASBA von sekundärer Bedeutung und werden daher in diesen Abschnitt nicht übernommen. Die anderen Ziele werden noch verfeinert.

> > > Ziele aus Abschnitt SZieleïn Anforderungen umwandeln. < < <

1. *Form*: Die Daten liegt in formaler, geprüfter Form vor. (siehe Ziel 2 auf Seite 6)
2. *Eingaben*: Die Eingabe von Daten erfolgt in einer formalen Syntax unter Verwendung der üblichen mathematischen Schreibweise. Folgende Daten können eingegeben werden:
   a) Axiome
   b) Sätze
   c) Beweise
   d) Fachbegriffe
   e) Fachgebiete
   f) Ausgabeschemata

   Dabei sind alle Begriffe nur innerhalb eines Fachgebietes und seiner untergeordneten Fachgebiete gültig, solange sie nicht umdefiniert werden. Das oberste Fachgebiet ist die ganze Mathematik. (siehe Ziel 3 auf Seite 6)
3. *Prüfung*: Vorhandene Beweise können automatisch geprüft werden. (siehe Ziel 4 auf Seite 6)
4. *Ausgaben*: Die Ausgabe kann in einer eindeutigen, formalen Syntax gemäß vorhandener Ausgabeschemata erfolgen. (siehe Ziel 5 auf Seite 6)
5. *Auswertungen*: Zusätzlich zur Ausgabe der Daten sind verschiedene Auswertungen möglich. Insbesondere kann zu jedem Beweis angegeben werden, wie viele Beweisschritte und welche Axiome und Sätze[1)] er verwendet. (siehe Ziel 6 auf Seite 6)

[1)] Sätze, die quasi als Axiome verwendet werden.

6. *Anpassbarkeit*: Fachbegriffe und die Darstellung bei der Ausgabe können mit Hilfe von – gegebenenfalls unbenannten – untergeordneten Fachgebieten angepasst werden. (siehe Ziel 7 auf Seite 6)
7. *Individualität*: Axiome und Sätze können für jeden Beweis individuell vorausgesetzt werden. Dabei sind fachgebietsspezifische Fachbegriffe erlaubt. (siehe Ziel 8 auf Seite 6)
8. *Internet*: Die Daten können auf mehrere Dateien verteilt sein. Ein Teil davon – oder sogar alle – können im Internet liegen. (siehe Ziel 9 auf Seite 6)
9. *Kommunikation*: Die Kommunikation mit ASBA kann mit den Fachbegriffen der einzelnen Fachgebiete erfolgen. (siehe Ziel 10 auf Seite 6)
10. *Zugriff*: Der Zugriff auf ASBA kann lokal und über das Internet erfolgen. (siehe Ziel 11 auf Seite 6)
11. *Unabhängigkeit*: ASBA kann offline und online arbeiten. (siehe Ziel 12 auf Seite 6)
12. *Rekursion*: Es kann rekursiv über alle verwendeten Dateien – auch solchen, die im Internet liegen – ausgewertet werden. (siehe Ziel 13 auf Seite 6)
13. *Bedienbarkeit*: ASBA ist einfach zu bedienen. (siehe Ziel 14 auf Seite 6)

3.2. Axiome

> > > Axiome auswählen und definieren. < < <

3.3. Beweise

> > > Schlussregeln auswählen und Beweise definieren. < < <

3.4. Datenstruktur

> > > Datenstruktur abstrakt und in XML definieren. < < <

3.5. Bausteine

> > > Bausteine? definieren. < < <

A. Anhang

A.1. Werkzeuge

Da dies ein Open Source Projekt sein soll, müssen alle Werkzeuge, die zum Ablauf der Software erforderlich sind, ebenfalls Open Source sein. Für die reine Entwicklung sollte das auch gelten, muss es aber nicht.

Werkzeuge zur Übersetzung der Quelldateien

1. Ein Übersetzer für LaTeXQuellcode (*.tex). – Verwendet wird *MiKTeX*.
2. Ein Übersetzer für C++ Quellcode (*.c, *.cpp, *.h, *.hpp). – Verwendet wird *Visual Studio Community 2017*.

Nicht unbedingt nötig, aber sinnvoll:

3. Ein Dokumentationssystem für in C++ Quellcode und darin enthaltene Doxygen Kommentare (*.c, *.cpp, *.h, *.hpp). – Verwendet wird *Doxygen* mit Konfigurationsdatei „Doxyfile“.
4. Ein Konfigurationsmanagementsystem zur Verwaltung der Quelldateien. – Verwendet wird *GitHub*.

Werkzeuge für die Entwicklung

5. *GitHub* als Online Konfigurationsmanagementsystem zur Zusammenarbeit verschiedener Entwickler. → `https://github.com/` – Lizenz siehe [7]
6. GitHub benötigt *Git* als Konfigurationsmanagementsystem. → `https://git-scm.com/` – Lizenz siehe [7]
7. *MiKTeX* für Dokumentation und Ausgaben in LaTeX. → `https://miktex.org/` – Lizenz siehe [11]
8. angedacht: *Visual Studio Community 2017*[1)] (*VS*) als Entwicklungsumgebung für C++. → `https://www.visualstudio.com/downloads/` – Lizenz siehe [10]
9. angedacht: In *Visual Studio Community 2015* integrierte Datenbank für Axiome, Sätze, Beweise, Fachbegriffe und Fachgebiete. – Lizenz siehe [10]
10. angedacht: *RapidXml* für Ein- und Ausgabe in XML. → `http://rapidxml.sourceforge.net/index.htm` – Lizenz siehe wahlweise [3] oder [13] [2)]
11. angedacht: *Doxygen* als Dokumentationssystem für C++. → `http://www.stack.nl/~dimitri/doxygen/` – Lizenz siehe [7]
12. angedacht: Doxygen benötigt *Ghostscript* als Interpreter für Postscript und PDF. → `http://ghostscript.com/` – Lizenz siehe [5]

---

1) Visual Studio Community ist zwar nicht Open Source, darf aber zur Entwicklung von Open Source Software unentgeltlich verwendet werden.

2) RapidXml stellt eine C++ Header-Datei zur Verfügung. Wenn diese im Quellcode eines Programms enthalten ist, gilt das ganze Programm als Open Source. Wenn diese Header-Datei nur in einer Bibliothek innerhalb eines Projekts verwendet wird, so gilt nur diese Bibliothek als Open Source.

13. angedacht: Doxygen benötigt *Graphviz* mit *Dot* zur Erzeugung und Visualisierung von Graphen. → http://www.graphviz.org/Home.php – Lizenz siehe [4]

Werkzeuge zur Bearbeitung der Quelldateien

14. *TEXstudio* als Editor für LATEX. → http://www.texstudio.org/ – Lizenz siehe [7]
TEXstudio benötigt einen Interpreter für Perl:

15. *Strawberry Perl* als Interpreter für Perl. → http://strawberryperl.com/ – Lizenz: Various OSI-compatible Open Source licenses, or given to the public domain

16. *Notepad*++ als Text-Editor. → https://notepad-plus-plus.org/ – Lizenz siehe [6]

17. *WinMerge* zum Vergleich von Dateien und Verzeichnissen. → http://winmerge.org/ – Lizenz siehe [6]

A.2. Offene Aufgaben

1. TODOs bearbeiten
2. Eingabeprogramm erstellen (liest XML)
3. Prüfprogramm erstellen
4. Ausgabeprogramm erstellen (schreibt XML)
5. Formelausgabe erstellen (erzeugt LATEX aus XML)
6. Axiome sammeln und eingeben
7. Sätze sammeln und eingeben
8. Beweise sammeln und eingeben
9. Fachbegriffe und Symbole sammeln und eingeben
10. Fachgebiete sammeln und eingeben
11. Ausgabeschemata sammeln und eingeben

B. Verzeichnisse

Tabellenverzeichnis

Abbildungsverzeichnis

Literaturverzeichnis

[1] Wolfgang Rautenberg, *Einführung in die Mathematische Logik*: Ein Lehrbuch, 3. Auflage, Vieweg+Teubner 2008

[2] *Apache License*, Version 2.0 →[1] http://www.apache.org/licenses/LICENSE-2.0 02.01.2004[2]

[3] *Boost Software License* 1.0 → http://www.boost.org/users/license.html 17.08.2003

[4] *Eclipse Public License* Version 1.0 → http://www.eclipse.org/org/documents/epl-v10.php 09.03.2017

[5] *GNU Affero General Public License* → http://www.gnu.org/licenses/agpl 19.11.2007

[6] *GNU General Public License* → http://www.gnu.org/licenses/old-licenses/gpl-1.0 02.1989

[7] *GNU General Public License*, Version 2 → http://www.gnu.org/licenses/old-licenses/gpl-2.0 06.1991

[8] *GNU Lesser General Public License*, Version 2.1 → http://www.gnu.org/licenses/old-licenses/lgpl-2.1 02.1999

[9] Lizenz für *Clover* → https://www.atlassian.com/software/clover 2017

[10] Lizenz für *Microsoft Visual Studio Express 2015* → https://www.visualstudio.com/de/license-terms/mt171551/ 2017

[11] Lizenz für *MikTeX* → https://miktex.org/kb/copying 14.01.2014

[12] Lizenz für *SAX* → http://www.saxproject.org/copying.html 05.05.2000

[13] *MIT License* → https://opensource.org/licenses/MIT/ 09.03.2017

[14] *Oracle Binary Code License Agreement* → http://java.com/license 02.04.2013

[15] *OSI Certified Open Source Software* → https://opensource.org/pressreleases/certified-open-source.php 16.06.1999

[16] *W3C Document License* → http://www.w3.org/Consortium/Legal/2015/doc-license 01.02.2015

[17] *W3C Software Notice and License* → http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html 13.05.2015

[18] *Hilbert II – Introduction* → http://www.qedeq.org/ 20.01.2014

[19] *Formal Correct Mathematical Knowledge*: GitHub Repository vom Projekt Hilbert II → https://github.com/m-31/qedeq/ 04.08.2016

[20] *ASBA – Axiome, Sätze, Beweise und Auswertungen*. Projekt zur maschinellen Überprüfung von mathematischen Beweisen und deren Ausgabe in lesbarer Form: GitHub Repository vom Projekt ASBA – in Bearbeitung → https://github.com/Dr-Winfried/ASBA

[1] Der Pfeil (→) verweist stets auf einen Link zu einer Seite im Internet.

[2] Das Datum hinter dem Link gibt – je nachdem welches bekannt ist – das Datum der letzten Änderung, den Stand der Seite oder das Datum, an dem die Seite angeschaut wurde an. Sind mehrere Daten vorhanden, wird das erste vorhandene in der angegebenen Reihenfolge genommen. – Dies gilt für alle hier aufgelisteten Seiten im Internet.

[21] Meyling, Michael: *Anfangsgründe der mathematischen Logik*
→ http://www.qedeq.org/current/doc/math/qedeq_logic_v1_de.pdf 24. Mai 2013 (in Bearbeitung)

[22] Meyling, Michael: *Formale Prädikatenlogik*
→ http://www.qedeq.org/current/doc/math/qedeq_formal_logic_v1_de.pdf 24. Mai 2013 (in Bearbeitung)

[23] Meyling, Michael: *Axiomatische Mengenlehre*
→ http://www.qedeq.org/current/doc/math/qedeq_set_theory_v1_de.pdf 24. Mai 2013 (in Bearbeitung)

[24] Meyling, Michael: *Elements of Mathematical Logic*
→ http://www.qedeq.org/current/doc/math/qedeq_logic_v1_en.pdf 24. Mai 2013 (in Bearbeitung)

[25] Meyling, Michael: *Formal Predicate Calculus*
→ http://www.qedeq.org/current/doc/math/qedeq_formal_logic_v1_en.pdf 24. Mai 2013 (in Bearbeitung)

[26] Meyling, Michael: *Axiomatic Set Theory*
→ http://www.qedeq.org/current/doc/math/qedeq_set_theory_v1_en.pdf 24. Mai 2013 (in Bearbeitung)

[27] Wikipedia: *Aussagenlogik* Kapitel 2.2 *Mögliche Junktoren*
→ https://de.wikipedia.org/wiki/Junktor#M.C3.B6gliche_Junktoren 20.01.2016

[28] Wikipedia: *Aussagenlogik* Kapitel 4 *Formaler Zugang*
→ https://de.wikipedia.org/wiki/Aussagenlogik#Formaler_Zugang 13.02.2017

[29] Wikipedia: *Identität (Logik)* Kapitel 2.3 *Identität in der Informatik* → https://de.wikipedia.org/wiki/Identit%C3%A4t_(Logik)#Identit.C3.A4t_in_der_Informatik 18.05.2017

[30] Wikipedia: *Kalkül* → https://de.wikipedia.org/wiki/Kalk%C3%BCl 26.02.2017

[31] Wikipedia: *Mengenlehre* → https://de.wikipedia.org/wiki/Mengenlehre 03.03.2017

[32] Wikipedia: *Prädikatenlogik erster Stufe*
→ https://de.wikipedia.org/wiki/Pr%C3%A4dikatenlogik_erster_Stufe 24.02.2017

[33] Wikipedia: *Schlussregel* → https://de.wikipedia.org/wiki/Schlussregel 01.05.2017

[34] Wikipedia: *Natürliches Schließen*
→ https://de.wikipedia.org/wiki/Systeme_nat%C3%BCrlichen_Schlie%C3%9Fens 01.05.2017

Index

Symbolverzeichnis

Glossar

(**AR**) Anfangsregel. 34, 35, 37

(**FS**) formaler Satz. 34, 35, 38

(**MR**) Monotonieregel. 34, 35, 38

(**SR**) Schnittregel (Modus ponens). 34, 35, 39

(**TR**) Abtrennungsregel. 34, 35, 37

$\mathcal{A}$ Das Alphabet der aussagenlogischen Sprache. 24, 35

$\mathcal{A}_x$ Eine Teilmenge des Alphabets $\mathcal{A}$ der aussagenlogischen Sprache. 24, 35

$\mathcal{B}$ Die Menge der aussagenlogischen, binären Operatoren. 23, 35

$\mathcal{C}$ Die Menge der aussagenlogischen Konstanten. 23, 35

$\mathcal{F}_x$ Eine Teilmenge der Menge $\mathcal{F}$ der aussagenlogischen Formeln mit Klammerung. 24, 35

$\mathcal{F}_x^p$ Eine Teilmenge der Menge $\mathcal{F}$ der aussagenlogischen Formeln in polnischer Notation. 24, 35

$\mathcal{J}$ Die Menge der aussagenlogischen Operatoren. 24, 35

$\mathcal{J}_x$ Eine Teilmenge der Menge $\mathcal{J}$ der aussagenlogischen Operatoren . 24, 35

$\mathcal{M}$ Die Menge der metasprachliche Operatoren und der mit Gleichheit verwandten Symbole. 13, 35

$\mathcal{U}$ Die Menge der aussagenlogischen unären Operatoren. 23, 35

$\mathcal{V}$ Die Menge der aussagenlogischen atomaren Formeln. 24, 35

$:=$ Ein metasprachlicher Operator: ... definitionsgemäß gleich ... 13, 23, 35

$\vdash$ Ableitungsrelation: ... ableitbar ... (siehe *ableitbar*). 15, 23, 34, 35, 37

$=$ Ein metasprachlicher Operator: ... gleich (ist dasselbe wie, ist identisch zu) ... 13, 14, 23, 35

$\equiv$ Ein (Meta-)Operator: ... äquivalent (ist das gleiche wie, ist so wie) zu ... 13, 23, 35

$\bot$ Eine Aussagenlogische Konstante: Falsch. 22, 35

$\lhd$ Ein Beispielsymbol für eine Relation mit Umkehrrelation $\rhd$. 14, 35

$\unlhd$ Ein Beispielsymbol für eine Relation mit Gleichheit und Umkehrrelation $\unrhd$. 14, 35

$\top$ Eine Aussagenlogische Konstante: Wahr. 22, 35

$:\Leftrightarrow$ Ein metasprachlicher Operator: ... definitionsgemäß gleich (definitionsgemäß genau dann, wenn) ... 13, 23, 35

$\Leftrightarrow$ Ein metasprachlicher Operator: ... genau dann wenn ... 12, 13, 23, 35

$\Rightarrow$ Ein metasprachlicher Operator: ... dann auch ... 12, 13, 23, 35

$||$ Ein metasprachlicher Operator: ... oder ... 12, 23, 34, 35

$\Leftarrow$ Ein metasprachlicher Operator: ... sofern ... 12, 13, 23, 35

&& Ein metasprachlicher Operator: ... und ... Die Priorität ist höher als die von '|'. 12, 14, 15, 23, 34, 35, 37

$\neq$ Ein metasprachlicher Operator: ... ungleich (nicht dasselbe wie, nicht identisch zu) ... 13, 14, 23, 35

$\not\equiv$ Ein (Meta-)Operator: ... nicht äquivalent (ist nicht das gleiche wie, ist nicht so wie) ... 13, 23, 35

$\sim$ Ein Beispielsymbol für eine Relation. 14, 35

$\simeq$ Ein Beispielsymbol für eine Relation mit Gleichheit. 14, 35

$\rhd$ Ein Beispielsymbol für eine Relation mit Umkehrrelation $\lhd$. 14, 35

$\unrhd$ Ein Beispielsymbol für eine Relation mit Gleichheit und Umkehrrelation $\unlhd$. 14, 35

$\longleftarrow$ Substitution: ... substituiert durch ... (siehe die Definition in Unterabschnitt 2.3.1 auf Seite 15). 23, 35

| Ein metasprachlicher Operator: ... und ... Die Priorität ist niedriger als die von '&&'. 13, 23, 34, 35, 37

(($\wedge$B)) Beseitigung von '($\wedge$B)'. 35

(($\wedge$E)) Beseitigung von '($\wedge$E)'. 35

((=B)) Beseitigung von '='. 17, 35

((=E)) Einführung von '='. 35

(($\rightarrow$B)) Beseitigung von '($\rightarrow$B)'. 35

(($\rightarrow$E)) Beseitigung von '($\rightarrow$E)'. 35

(($\neg$1)) Einführung/Beseitigung von '$\neg$' Teil 1. 35

(($\neg$2)) Einführung/Beseitigung von '$\neg$' Teil 1. 35

(($\neg$3)) Beweistechnik „Indirekter Beweis“. 35

(($\neg$4)) Reductio ad absurdum (indirekter Beweis). 35

ableitbar Wenn sich eine Formel β aus einer Formel α mittels zulässiger Transaktionen ableiten lässt, heißt β ableitbar aus α. Sprechweise: „α ableitbar β“. Eine oder beide Formeln α bzw. β dürfen dabei durch Formelmengen ersetzt werden. (*siehe auch* $\vdash$ und Ableitungsrelation) – Synonym: beweisbar. 15, 34, 36

Ableitungsrelation Die Relation '$\vdash$'. 23, 34, 37

Abtrennungsregel Eine *Schlussregel* - siehe (TR). 17, 34, 36

allgemeingültige Schlussregel Eine Schlussregel die aus den Basisregeln und den schon bekannten allgemeingültigen Schlussregeln abgeleitet werden kann. 18, 19, 34, 39

Anfangsregel Eine *Schlussregel* um beginnen zu können - siehe (AR). 16, 34, 36

ASBA Programmsystem, das **A**xiome, **S**ätze, **B**eweise und **A**uswertungen behandeln kann. 4–6, 8, 9, 11, 12, 14, 23, 27, 28, 31, 34

atomare Formel Eine Formel, die sich nicht weiter zerlegen lässt. 24, 34, 36

Ausgabeschema Ein Schema, mit dem bestimmte mathematische Objekte ausgegeben werden sollen. 6, 8, 9, 27, 30, 34

Aussage Eine Aussage in natürlicher Sprache oder als Formel, die einen Wahrheitswert liefert. 34, 38, 39

Axiom Eine Formel, die unbewiesen als wahr angesehen wird. 4–9, 23, 27–30, 34

Basisregel Eine *Schlussregel*, die nicht mehr auf andere zurückgeführt wird. Obwohl das auch auf die Identitätsregeln zutrifft, werden diese hier aber nicht dazu gezählt. 14, 15, 17, 19, 31, 34, 37, 38

Beweis Eine zulässige Ableitung von Folgerungen aus gegebenen Voraussetzungen. 4–9, 23, 27–30, 34

beweisbar Synonym zu *ableitbar*. 15, 34, 37

Beweisschritt Eine Vorschrift, wie aus vorgegebenen Aussagen eine weitere folgt. 11, 34

Boolsche Signatur Die logische Signatur $\{\neg, \wedge, \vee\}$. 25, 34

Fachbegriff Ein Name für einen mathematischen Begriff. 4–6, 9, 27–30, 34

Fachgebiet Ein Teil der Mathematik mit einer zugehörigen Basis aus Axiomen, Sätzen und spezifischen Fachbegriffen und Darstellungen. 4–6, 9, 27–30, 34

Folgerung Die Folgerungen einer Schlussregel sind die Aussagen über ihrem Querstrich. . 16, 18, 34

formaler Satz Formale Darstellung eines mathematischen Satzes - siehe (FS). 14, 34, 36

formales Element Ein mathematisches Element in formaler Schreibweise. Bis auf wenige Aussagen kommen darin *metasprachliche Ausdrücke* nicht mehr vor. 13–16, 34, 39

Identitätsregel Eigentlich eine Basisregel zur Identität. Da die Identitätsregeln nur zur Rechtfertigung der Substitution verwendet werden, werden sie hier nicht zu den Basisregeln gezählt. 17, 34, 38

interessierende Eigenschaft Solche Eigenschaften von Ausdrücken, die im aktuellen Zusammenhang von Interesse sind. 12, 13, 34

Kontraposition Die allgemeingültige Aussage $(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)$. 21, 31, 34

logische Signatur Eine in *Metasprache* verfasste Aussage, die auch zusammengesetzt sein kann. 24, 25, 34, 38

Metasprache Eine Sprache, in der Aussagen über Elemente einer anderen Sprache getroffen werden können. 11, 34, 38

metasprachliche Aussage Eine in *Metasprache* verfasste Aussage, die auch zusammengesetzt sein kann. 11, 34

metasprachlicher Ausdruck Eine in normaler Sprache verfasste Aussage, die auch zusammengesetzt sein kann. 11, 12, 34, 38, 39

metasprachlicher Operator Ein Operator, dessen Operanden *metasprachliche Ausdrücke* sind. 12, 23, 34, 36, 37

Monotonieregel Eine *Schlussregel* - siehe (MR). 15, 16, 34, 36

Prädikat Ein Element der *Prädikatenlogik* (*siehe* Abschnitt 2.5 auf Seite 27). Z. B. kann man eine „$Gruppe$“ als ein zweistelliges Prädikat „$Gruppe(G, +)$“ definieren, in dem G eine Menge und '$+$' eine Operation, d. h. eine (zweistellige) Funktion „$+ : G \times G \to G$“ ist, so dass die Gruppenaxiome erfüllt sind. 9, 34

Prädikatenlogik siehe Abschnitt 2.5 auf Seite 27. 34, 38

Satz Eine mathematische Aussage, dass eine bestimmte Folgerung aus gegebenen Voraussetzungen abgeleitet werden kann. 4–6, 8, 9, 23, 27–30, 34

Schlussregel Eine Regel für eine (zulässige) Umwandlung von Formeln. 15, 18, 34, 37–39

Schnittregel Eine allgemeingültige Schlussregel - siehe (SR). 17, 19, 31, 34, 36

Substitution Die Ersetzung von einem, mehreren oder allen formalen Elementen (α) in einem anderen formalen Element (Γ) durch ein drittes formales Element (β), formal: $\Gamma(\alpha \leftarrow\!\!\prec \beta)$. Wenn alle α in Γ durch β ersetzt, ist die Substitution *vollständig*. (siehe die Definition (2.3) in Unterabschnitt 2.3.2). 15, 16, 34, 37, 38

vergleichbar Zwei *metasprachliche Ausdrücke* bzw. *formale Elemente* heißen – auf eine bestimmte Art – vergleichbar, wenn sie auf diese Art (z. B. als Zeichenketten oder als vergleichbare Ergebnisse von Formeln) verglichen werden können. Die Art muss implizit bekannt oder explizit angegeben sein. Meistens genügt es zu wissen, was für metasprachliche Ausdrücke bzw. formale Elemente es sind. Sie müssen dann nur von derselben Art sein. 12, 13, 16, 34

Voraussetzung Die Voraussetzungen einer Schlussregel sind die Aussagen über ihrem Querstrich. . 16, 34

Wahrheitswert Wahrheitswerte sind die Werte „wahr" und „falsch", oft auch als „true" und „false" oder einfach '1' und '0' bezeichnet. 12, 13, 34

zulässige Transformation Eine zulässige Umformung oder Erzeugung einer Formel aus einer vorgegebenen Menge von Formeln, d. h. die Anwendung einer allgemeingültigen Schlussregel. 14, 15, 34