

der Definition der Syntax gesagt *wenn F und G Formeln sind* (vgl. mit der Induktionsannahme!), *dann sind auch $\neg F$, $(F \wedge G)$ usw. Formeln* (dies wird durch den Induktionsschritt abgedeckt).

Mit der strukturellen Induktion zeigt man eine Behauptung also tatsächlich für alle Formeln der Aussagenlogik.

Hat man etwas Übung, so geht man bei obigem knapper vor. Zur Illustration wollen wir dies hier einmal machen. Wir wollen $\text{grad}(F) \leq L(F)$ für alle $F \in \mathcal{L}_{AL}$ beweisen.

Beweis. Wir zeigen dies mittels struktureller Induktion. Der Induktionsanfang ist klar, denn für jedes Aussagensymbol A gilt $\text{grad}(A) = 0 \leq 1 = L(A)$. Sei nun angenommen, dass F und G zwei Formeln sind mit $\text{grad}(F) \leq L(F)$ und $\text{grad}(G) \leq L(G)$. Im Induktionsschritt ergibt sich nun $\text{grad}(\neg F) = \text{grad}(F) + 1 \leq L(F) + 1 = L(\neg F)$ und $\text{grad}((F \circ G)) = \text{grad}(F) + \text{grad}(G) + 1 \leq L(F) + L(G) + 1 \leq L(F) + L(G) + 3 = L((F \circ G))$ für jeden zweistelligen Junktor \circ . \square

Im obigen Beweis wird sehr schnell vorgegangen. Insb. wird nicht einmal mehr an den einzelnen Stellen begründet, warum die (Un-)Gleichungen gelten. Da nur sehr wenig benutzt werden kann (nämlich nur die Definition von grad , von L oder einfache Rechenregeln), ist dies in Ordnung. Macht man aber seine ersten Beweise dieser Art, sollte man immer sauber begründen, warum einzelne Schritte gelten.

Fragen

1. Beweisen Sie mittels struktureller Induktion, dass für alle Formeln F der Aussagenlogik gilt:

$$1 + \text{tiefe}(F) \leq L(F)$$

2. Beweisen Sie mittels struktureller Induktion, dass für alle Formeln F der Aussagenlogik gilt:

$$\text{tiefe}(F) \leq \text{grad}(F)$$

2.4 Noch einmal kurz und knapp

Wir fassen den Stoff zur Syntax der Semantik sowie zu den darauf basierenden Themen zur strukturellen Rekursion und zur strukturellen Induktion noch einmal zusammen und wiederholen die einzelnen Begriffe. Bei Unklarheiten zu den einzelnen Begriffen, lohnt sich stets ein Blick in den Haupttext oben.

Syntax der Aussagenlogik

Wir haben in diesem Abschnitt die Syntax der Aussagenlogik definiert, d.h. wir haben definiert, was wir alles als Formeln der Aussagenlogik bezeichnen wol-

len. Wichtige Begriffe waren hier: Alphabet, Aussagensymbol, Junktor, atomare Formel, komplexe Formel, wohlgeformte Formel, Formel, Teilformel, Hauptoperator, Negation, Konjunktion, Disjunktion, Implikation und Biimplikation. Als Symbole haben wir $\neg, \vee, \wedge, \Rightarrow, \Leftrightarrow$ für die Junktoren sowie AS_{AL} und \mathcal{L}_{AL} für die Menge der Aussagensymbole bzw. die Menge der wohlgeformten Formeln der Aussagenlogik kennengelernt.

Um die Menge \mathcal{L}_{AL} zu definieren, haben wir gesagt, dass alle Aussagensymbole (atomare) Formeln sind und dass, wenn F und G bereits Formeln sind, dann auch $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \Rightarrow G)$ und $(F \Leftrightarrow G)$ (komplexe) Formeln sind.

Eine wohlgeformte Formel der Aussagenlogik ist dann z.B. die Zeichenkette $((A \wedge B) \Rightarrow \neg C)$ nicht aber $)A \vee B($ (und zunächst auch z.B. nicht $A \wedge B$, da wir noch keine Klammerersparnisregeln kennengelernt haben).

Strukturelle Rekursion

Aufbauend auf der Definition der wohlgeformten Formeln der Aussagenlogik haben wir dann mit der strukturellen Rekursion eine Möglichkeit kennengelernt, eine Funktion f zu definieren, die jeder Formel $F \in \mathcal{L}_{AL}$ einen Wert $f(F)$ zuweist.

Um dies auszuführen, um also eine totale Funktion $f : \mathcal{L}_{AL} \rightarrow D$ zu definieren (wobei D eine beliebige Menge ist), haben wir gesehen, dass man zunächst in der Rekursionsbasis $f(A)$ für jedes $A \in AS_{AL}$ festlegt. Anschließend legt man dann im Rekursionsschritt für die Negation eine Funktion $f_{\neg} : D \rightarrow D$ und für jeden zweistelligen Junktor $\circ \in \{\vee, \wedge, \Rightarrow, \Leftrightarrow\}$ eine Funktion $f_{\circ} : D \times D \rightarrow D$ fest. Mit diesen kann dann für eine beliebige Formel F der Wert $f(F)$ rekursiv ermittelt werden. Ist bspw. $F = (G \wedge H)$, so ist $f(F) = f_{\wedge}(f(G), f(H))$ und $f(G)$ und $f(H)$ müssen rekursiv weiter bestimmt werden.

Als Beispiele für mittels struktureller Rekursion definierte Funktionen haben wir den Grad und die Tiefe einer Formel kennengelernt. Ferner haben wir in einem Exkurs Strukturbäume eingeführt.

Strukturelle Induktion

Zuletzt haben wir gesehen, wie man, ebenfalls aufbauend auf der Definition der wohlgeformten Formeln der Aussagenlogik, mittels struktureller Induktion die Gültigkeit einer Behauptung für alle Formeln der Aussagenlogik zeigen kann.

Dazu muss man zunächst im Induktionsanfang die Behauptung für alle Aussagensymbole (also alle atomaren Formeln) zeigen. In der Induktionsannahme nimmt man dann an, dass die Behauptung für zwei Formeln F und G gilt und zeigt dann im Induktionsschritt, dass die Behauptung (unter dieser Annahme) auch für $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \Rightarrow G)$ und $(F \Leftrightarrow G)$ gilt.

Die strukturelle Induktion ist damit eine (Beweis-)Technik, mit der man Behauptungen für alle Formeln der Aussagenlogik zeigen kann. Exemplarisch haben wir gesehen, dass der Grad einer aussagenlogischen Formeln höchstens so groß werden kann wie die Länge der Formel. Weitere Beispiele sind in den Fragen im Abschnitt zur strukturellen Induktion zu finden.