Logica

Collegedictaat bij IN2 013 en IN2 310

J.F.M. Tonino

juli 2002

Voorwoord

Een voorwoord dient met een verantwoording te beginnen. Welnu, het voorliggende dictaat *Logica* is gebaseerd op het boek:

> S.C. van Westrhenen, R. Sommerhalder en J.F.M. Tonino, *Logica, een inleiding met toepassingen in de informatica*, Academic Service, Schoonhoven, 1993.

Op een aantal punten wijkt het dictaat af van het boek:

- Vele kleine, en minder kleine, fouten zijn verbeterd.
- Een gedeelte van het materiaal is vervallen. Dit betreft voornamelijk het deel over axiomatische semantiek (Hoare-logica), de theorie over programmacorrectheid. Verder is de theorie rond de stelling van Herbrand komen te vervallen.
- De boomstelling (11.3.4) voor de predicatenlogica is zodanig gegeneraliseerd dat zij ook voor oneindige bomen geldt. Het uitgebreide bewijs voor de compactheidsstelling (13.2.2) kon hierdoor aanzienlijk worden vereenvoudigd.
- De paragrafen 8.1, 9.1, en 12.1 zijn herzien door een herschikking van de tekst en door toevoeging van diagrammen.

Het dictaat bevat tentamenstof behorende bij de colleges:

- **IN2013** (oude curriculum)

  Het gehele dictaat, met uitzondering van de gedeelten die in een klein lettertype zijn gezet, behoort tot de tentamenstof.

- **IN2310** (nieuwe curriculum, met ingang van collegejaar 2002/2003)

  Alleen de hoofdstukken 1 tot en met 14 behoren tot de tentamenstof met uitzondering van de gedeelten die in een klein lettertype zijn gezet. De inhoud van de hoofdstukken 15 tot en met 17 zal alleen worden geschetst, maar niet worden getoetst.

De gedeelten in een kleiner lettertype die niet tot de tentamenstof behoren, zijn meer geavanceerd. Laat dit de lezer er niet van weerhouden om de betreffende teksten te lezen! Deze brengen een zekere academische verdieping aan.

Suggesties en foutmeldingen blijven uiteraard van harte welkom.

J.F.M. Tonino Delft, juli 2002

Inhoudsopgave

Hoofdstuk 1

Wat is Logica?

Het woord *logica* wordt in uiteenlopende betekenissen gebruikt. Daarom bespreken we eerst in paragraaf 1.1 wat wij in dit boek onder logica zullen verstaan. Vervolgens zal in paragraaf 1.2 een aantal voorbeelden van logische redeneringen worden gepresenteerd en zal op informele wijze worden onderzocht wanneer een redenering logisch geldig is. Tenslotte geven we in paragraaf 1.3 een beknopt overzicht van de geschiedenis van de logica.

1.1 Het onderwerp van logica

Het woord *logica* is afgeleid van het Griekse woord 'logikos', *de taal, het spreken* of *de menselijke rede betreffend*, dat weer afstamt van 'logos', *woord, rede* of *begrip*. Volgens Van Dale's Groot Woordenboek der Nederlandse Taal (12^{e} druk, 1992) luidt de eerste betekenis van logica:

> *wetenschap die zich met de wetten van het denken bezighoudt.*

De logica als wetenschappelijke discipline wordt tegenwoordig verdeeld in de *wijsgerige* en de *formele* logica.

In de wijsgerige logica houdt men zich bezig met de filosofische grondslagen en de begrenzingen van het menselijk logisch denken. Voorbeelden van fundamentele vragen op dit onderzoeksgebied zijn: 'Wat is een logische redenering?', 'Wanneer is een redenering logisch correct?', 'Wat is de reikwijdte van het menselijk denken?', 'Zijn er verschillende logica's mogelijk?', 'Waarin onderscheidt zich een logische redenering in een normatieve wetenschap, bijvoorbeeld de rechtswetenschap, van een logische redenering in een empirische wetenschap zoals bijvoorbeeld de natuurkunde?' en 'Wat is de empirische betekenis van een conclusie van een redenering?'.

In de formele logica, ook wel *mathematische* of *symbolische* logica genoemd, bestudeert men vooral de structuur van gegeven logica's of logische systemen. Hierbij wordt bij voorkeur gebruik gemaakt van mathematische methoden.

Men definieert bijvoorbeeld zeer nauwkeurig wat *beweringen* zijn en wanneer een bewering *waar*, *onwaar*, *algemeen geldig* of *onvervulbaar* is. Ook zal men de *redeneerregels*, op grond waarvan men uit een aantal premissen bepaalde conclusies mag trekken, exact willen beschrijven (*premisse*: reeds afgeleide bewering, letterlijk: voorafgaande stelling; *conclusie*: bewering geconcludeerd uit de premissen).

Uit het bovenstaande blijkt, dat de formele logica zich vooral bezig houdt met de *technische* aspecten van het (menselijke) logisch redeneren. De uitwerking van de eventuele filosofische implicaties van resultaten in de formele logica behoort dan weer tot de wijsgerige logica. In dit boek zullen we ons uitsluitend bezighouden met formele logica. Anders gezegd, wij zullen onder logica verstaan: *formele logica.*

1.2 Logische redeneringen

In deze paragraaf zullen we voorbeelden geven van logische redeneringen. Ook zullen we de vraag proberen te beantwoorden wat een redenering correct maakt. Beschouw om te beginnen de volgende zinnen:

(1.1) Als het regent, dan wordt de stoep nat.

Het regent.

Derhalve wordt de stoep nat.

Niemand zal moeite hebben om in deze zinnen een correcte redenering te herkennen. Immers, zo redeneren we, *als* we de eerste twee zinnen van (1.1) voor waar aannemen, *dan* kunnen we er niet onderuit om de derde zin te accepteren en dus voor waar aan te nemen. Blijkbaar speelt het begrip *waarheid* een rol bij het beoordelen of een redenering correct is. Om deze rol wat nader te bestuderen, bekijken we een ander stel zinnen dat wellicht een redenering vormt:

(1.2) Als het regent, dan wordt de stoep nat.

De stoep wordt nat.

Derhalve regent het.

Kunnen we nu (1.2) als een correcte redenering beschouwen? Laten we om deze vraag te beantwoorden, dezelfde procedure toepassen als we bij (1.1) hebben gedaan. We moeten ons dus de vraag stellen of we, indien we de eerste twee zinnen van (1.2) voor waar aannemen, gedwongen zijn om ook de derde zin voor waar aan te nemen. Welnu, dat zijn we niet! Immers, het kan heel goed mogelijk zijn dat onze buurman de stoep aan het schrobben is, zodat deze inderdaad nat wordt zonder dat het regent. Precies om deze reden vormen de zinnen (1.2) geen correcte redenering.

Om onze bevindingen precies te kunnen formuleren zullen we wat terminologie invoeren. In redenering (1.1) noemt men de eerste twee uitspraken de *premissen* van de redenering en de derde de *conclusie*. Een correcte redenering zullen we voortaan een *logisch geldige* redenering noemen.

We zijn nu in staat om ons verworven inzicht omtrent het logisch geldig zijn van een redenering als volgt te formuleren:

> *Een redenering is logisch geldig indien men gedwongen is om de conclusie voor waar aan te nemen, wanneer men de premissen voor waar aanneemt.* ($*$)

Op dit punt aangekomen vraagt de lezer zich wellicht af of dit betekent dat een redenering alleen logisch geldig kan zijn indien de premissen waar zijn. De volgende redenering laat zien dat dit niet het geval is:

(1.3) Als mensen vleugels hebben, dan kunnen dieren spreken.

Mensen hebben vleugels.

Derhalve kunnen dieren spreken.

Volgens het zojuist door ons geformuleerde criterium ($*$) is dit een logisch geldige redenering. Immers, we zijn gedwongen om de conclusie van (1.3) te accepteren, *indien* we de premissen ervan accepteren. Er wordt niet van ons verwacht *dat* we de premissen accepteren!

Vergelijken we redenering (1.3) met redenering (1.1), dan valt het op dat deze dezelfde *vorm* bezitten. Gebruikmakend van het symbool $\therefore$ dat *derhalve* betekent, kunnen we deze vorm als volgt duidelijk maken:

(1.4) Als p, dan q.

p.

$\therefore\ q$.

Het maakt niet uit wat we voor p en q invullen: zolang we voor beide voorkomens van p dezelfde bewering invullen en dat ook voor q doen, blijft de redenering logisch geldig. De belangrijke les die we hieruit leren, is dat de logische geldigheid van een redenering een eigenschap is van de vorm en niet van de inhoud ervan. Dit maakt het mogelijk om regels te formuleren op grond waarvan men, op logisch geldige wijze, conclusies kan afleiden uit gegeven premissen. Deze regels worden *afleidingsregels* genoemd.

De afleidingsregel die in (1.4) is gebruikt, noemt men *modus ponens*. Volgens deze regel mag men uit premissen van de vorm *Als p, dan q* en p de conclusie q afleiden. Schematisch kan men deze regel als volgt weergeven:

$$\frac{\text{Als } p, \text{ dan } q \qquad p}{q} \quad \textit{modus ponens}$$

Let op het verschil tussen een redenering en een afleidingsregel: een redenering is een reeks beweringen eindigende met een conclusie, terwijl een afleidingsregel bepaalt op grond van welke beweringen een conclusie mag worden getrokken.

Door afleidingsregels toe te passen kunnen *afleidingen* worden geconstrueerd. Een afleiding kan worden beschouwd als een redenering waarin iedere redeneerstap wordt gerechtvaardigd door de toepassing van een afleidingsregel. Afleidingsregels zijn dus het cement van een afleiding. Er zijn in de loop van de tijd verschillende systemen bedacht om afleidingen te construeren. In dit boek worden enkele hiervan besproken.

Redeneringen van het type (1.4) behoren tot het domein van de *propositielogica* die we uitgebreid in deel I zullen bestuderen. De volgende redenering behoort tot het gebied van de zogenaamde *predicatenlogica* (zie deel II):

(1.5) Alle mensen zijn sterfelijk.

Socrates is een mens.

$\therefore$ Socrates is sterfelijk.

Deze redenering voldoet aan onze test ($*$) voor logische geldigheid. De vorm van deze redenering is echter duidelijk anders dan die van bijvoorbeeld (1.1). Om wat meer greep te krijgen op de vorm van (1.5), geven we een redenering die er een variant van is.

(1.6) Alle mieren zijn onsterfelijk.

Socrates is een mier.

$\therefore$ Socrates is onsterfelijk.

Merk op dat de redenering logisch geldig is, terwijl de premissen onwaar zijn! De redeneringen (1.5) en (1.6) hebben beide de volgende vorm:

(1.7) Alle x die A zijn, zijn ook B.

s is A.

$\therefore$ s is B.

Substitueren we voor A overal 'mens', voor B 'sterfelijk' en voor s 'Socrates', dan hebben we onze oorspronkelijke redenering (1.5) weer terug. Evident blijft (1.7) logisch geldig ongeacht wat we voor A, B en s invullen. Ook hier wordt bevestigd dat de logische geldigheid van een redenering een eigenschap is van de vorm van die redenering.

1.3 Beknopte geschiedenis

Het onderzoek van het menselijk redeneren heeft een eeuwenlange traditie, die aanvankelijk vooral steunde op de logische geschriften van Aristoteles zoals die zijn verzameld in zijn 'Organon' (organon: werktuig).

Gedurende de vroege Middeleeuwen werden deze geschriften in de Arabische wereld uitvoerig bestudeerd en van commentaar voorzien. In het oosten (Bagdad) gebeurde dit onder andere door Al-Farabi (†950) en Avicenna (†1037) en in het westen (Cordoba) door Averroes (†1198). In de late Middeleeuwen werd dit, mede dankzij de latijnse vertaling van de geschriften van Aristoteles, voortgezet in de middeleeuwse christelijke scholastiek. De weerslag hiervan is onder andere te vinden in de geschriften van William van Ockham ('Summa Logicae', ca. 1326) en Walter Burleigh (†1349).

Het vaak herdrukte werk 'Logique ou l'Art de Penser' (1662), geschreven door A. Arnauld en P. Nicole heeft, vooral door de hierin gebruikte vierdeling: 'Réflexions sur les Idées', 'Sur les Jugements', 'Du Raisonnement' en 'De la Méthode', een grote invloed gehad.

De opbloei van de experimentele natuurwetenschap na de renaissance heeft in hoge mate de studie van natuurwetenschappelijke methoden gestimuleerd. Hierdoor ontstond de inductieve logica of de methodologie van de empirische wetenschappen als een nieuwe tak van de logica. Het eerste begin hiervan vindt men in de geschriften van Baco van Verulam (†1626), onder andere in zijn 'Novum Organum'; een verdere systematische ontwikkeling vindt men onder andere bij John Stuart Mill ('A System of Logic, Ratiocinative and Inductive', 1843), Wilhelm Wundt ('Logik', 1880 – 1883) en Christian Sigwart ('Logik', 1873 – 1887, heruitgave 1923).

Leibnitz' ideeën omtrent een 'Ars Combinatoria', een ontwerp van een universele symbolentaal voor het beschrijven van het menselijk redeneren, evenals de 'Wissenschaftslehre' (1837) van Bernard Bolzano vormen een aanzet tot de ontwikkeling van de mathematische logica.

In de negentiende eeuw werd door wiskundigen als Gottlob Frege, George Boole en Guiseppe Peano een begin gemaakt met de studie van de logica met behulp van wiskundige methoden. Vanaf dat moment kan men dus spreken van mathematische logica.

Na de ontwikkeling van een geschikte wiskundige notatie werd de mathematische logica al spoedig gebruikt voor de fundering van de wiskunde. Een eerste poging hiertoe was Frege's boek 'Grundgesetze der Arithmetik begriffschriftlich abgeleitet' (1891 en 1903). Deze faalde echter jammerlijk omdat Bertrand Russell aantoonde dat in Frege's systeem een contradictie (de beroemde paradox van Russell) afleidbaar was.

Een aantal jaren later gaven Bertrand Russell en Alfred North Whitehead in hun 'Principia Mathematica' (1910 – 1913) aan op welke wijze de wiskunde met behulp van de mathematische logica gefundeerd kan worden zonder dat dit leidt tot contradicties. Door de gecompliceerdheid ervan heeft de 'Principia Mathematica' echter niet iedereen kunnen overtuigen.

Latere funderingen van de wiskunde met behulp van de mathematische logica (of als onderdeel van de mathematische logica) vindt men in de school van de Duitse wiskundige David Hilbert en bij Paul Bernays, John von Neumann, Jacques Herbrand, Kurt Gödel, Alonzo Church, Stephen C. Kleene, Willard V.O. Quine, Alfred Tarski en en de Nederlandse logicus Evert W. Beth.

Mijlpalen in de ontwikkeling van de mathematische logica zijn de ontwikkeling van Tarski's semantiek voor formele talen ('Der Wahrheitsbegriff in den formalisierten Sprachen', 1933 – 1936), het werk over de vervulbaarheid van formules door Thoralf Skolem ('Logisch-kombinatorische Untersuchungen über die Erfüllbarkeit oder Beweisbarkeit mathematischer

Sätze nebst einem Theoreme über dichte Mengen', 1920), de onderzoekingen over bewijstheorie van Herbrand (in zijn proefschrift 'Recherches sur la théorie de la démonstration', 1930), de volledigheidsstelling van Gödel die de gelijkwaardigheid van waarheid en bewijsbaarheid uitdrukt ('Die Vollständigkeit der Axiome des logischen Funktionenkalküls', 1930), de onvolledigheidsstelling van de rekenkunde eveneens van Gödel ('Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I', 1931), het bewijs van de onbeslisbaarheid van de predicatenlogica door Church (1936), en het bewijs van de consistentie van de rekenkunde door Gerhard Gentzen (1936).

Van de resultaten na de Tweede Wereldoorlog vermelden we nog de non-standaardanalyse van Abraham Robinson, waarin een logische fundering wordt gegeven voor het gebruik van infinitesimale grootheden in de analyse, en het bewijs van de onafhankelijkheid van het keuze-axioma en de continuümhypothese door Paul Cohen.

Op dit moment is de formele logica een bloeiende internationaal beoefende wetenschap met eigen tijdschriften en congressen. Het vakgebied bevat een groot aantal deelgebieden zoals bewijstheorie, modeltheorie, recursietheorie, algoritmentheorie, axiomatische verzamelingenleer en typentheorie.

Na de Tweede Wereldoorlog heeft zich, onder invloed van de ontwikkelingen in de informatica en door de vele toepassingen van informatica in andere vakgebieden, zelfs een nieuwe tak van de logica ontwikkeld, namelijk de *toegepaste logica.* In de toegepaste logica onderzoekt men bijvoorbeeld hoe formele methoden uit de logica kunnen worden ingezet voor de grondslagen van de informatica of bij de ontwikkeling van bewijsbaar correcte computerprogramma's.

Tegenwoordig worden formele methoden die ontleend zijn aan de logica, niet alleen gebruikt in de wiskunde, de informatica, en de 'artificial intelligence', maar bijvoorbeeld ook in de biologie, de taalkunde en de rechtswetenschap.

Deel I

Propositielogica

Hoofdstuk 2

Syntaxis van de Propositielogica

De *propositielogica* houdt zich bezig met de analyse van *proposities* of *beweringen*, en hun logische betrekkingen. Kenmerkend voor deze analyse is het onderscheid tussen *enkelvoudige* en *samengestelde beweringen*, waarbij samengestelde beweringen zijn opgebouwd uit enkelvoudige beweringen en *voegwoorden* of *connectieven* (§2.1). Deze laatste zijn bepalend voor de genoemde logische betrekkingen tussen beweringen.

In de propositielogica worden beweringen voorgesteld als symbolische uitdrukkingen in een *formele taal*. Deze kunstmatige taal noemt men de *taal van de propositielogica*. Bij de beschrijving van zowel natuurlijke als formele talen hanteert men het onderscheid tussen *syntaxis* en *semantiek*. Onder de syntaxis van een taal verstaat men een beschrijving van de grammaticale aspecten ervan. De semantiek, daarentegen, is een beschrijving van de betekenis van de uitdrukkingen in die taal. In §2.2 wordt de syntaxis van de propositielogica behandeld. De semantiek komt aan de beurt in hoofdstuk 3.

De overige paragrafen uit dit hoofdstuk zijn gewijd aan onderwerpen die gerelateerd zijn aan de syntaxis van de propositielogica. In §2.3 komt het begrip *structurele inductie* aan de orde. Dit is een bewijstechniek die verwant is aan volledige inductie over de natuurlijke getallen. Vervolgens behandelen we in §2.4 zogenaamde *recursieve definities*. We besluiten het hoofdstuk met notationele kwesties betreffende de taal van de propositielogica (§2.5).

2.1 Beweringen en connectieven

Zoals gezegd is propositielogica de logica van beweringen en hun logische betrekkingen. Een *bewering* is een abstractie van wat uitgedrukt wordt door een constaterende zin. Voorbeelden van constaterende zinnen zijn:

(2.1) E.W. Dijkstra is een bekend Nederlands informaticus.

(2.2) Twee plus twee is gelijk aan vijf.

(2.3) Socrates is sterfelijk.

(2.4) Socrates is sterfelijk en twee plus twee is gelijk aan vijf.

(2.5) Twee plus twee is niet gelijk aan vijf.

(2.6) Als twee plus twee gelijk is aan vijf, dan is één gelijk aan nul.

(2.7) Mozart was schilder of Frans Brüggen is dirigent.

Al deze zinnen drukken een bepaalde stand van zaken ofwel een constatering uit; vandaar de naamgeving. Let op dat we een onderscheid maken tussen de zin en de bewering die door de zin wordt uitgedrukt. *Constaterende zin* is een grammaticaal concept, terwijl *bewering* een logisch concept is. Het is mogelijk dat verschillende constaterende zinnen dezelfde bewering uitdrukken. Bijvoorbeeld:

(2.8) Socrate est mortel.

en zin (2.3) drukken dezelfde bewering uit. In de praktijk hebben we weinig last van dit onderscheid tussen bewering en constaterende zin, omdat we in dit boek zelden of nooit belang stellen in grammaticale concepten.

Een bewering is *waar* of *onwaar*. Zo zijn bijvoorbeeld de beweringen uitgedrukt door de zinnen (2.1), (2.3), (2.5), (2.6) en (2.7) waar, terwijl de beweringen uitgedrukt door (2.2) en (2.4) onwaar zijn. Het is echter ook gebruikelijk om te spreken over de waarheid of onwaarheid van constaterende zinnen. Een zin is dan waar indien de uitgedrukte bewering overeenkomt met de werkelijke stand van zaken, en anders onwaar. Het zal blijken dat *waarheid* de enige voor de propositielogica relevante eigenschap van beweringen of zinnen is. Andere eigenschappen van beweringen of zinnen, zoals de lengte van een zin, of bijvoorbeeld de diepzinnigheid van een bewering, doen er niet toe.

Hoewel een bewering slechts waar of onwaar kan zijn, is het niet altijd mogelijk uit te maken of een gegeven bewering waar of onwaar is. Beroemde voorbeelden van beweringen waarvan de waarheidswaarde nog steeds niet bekend is, zijn:

(2.9) Het heelal is eindig.

(2.10) *Het vermoeden van Goldbach* (1690 – 1764):

Ieder even natuurlijk getal groter dan vier is de som van twee priemgetallen.

(2.11) *Het laatste theorema van Fermat*[1] (1601 – 1665):

---

[1] Inmiddels heeft de Engelsman Andrew Wiles dit vermoeden in 1994 bewezen.

> De vergelijking $x^n + y^n = z^n$ met n, x, y en z positieve gehele getallen, heeft voor $n > 2$ geen oplossing.

De grammaticale opbouw van de gegeven zinnen is niet erg ingewikkeld. Zo zijn (2.1), (2.2) en (2.3) enkelvoudige hoofdzinnen die geen voegwoorden bevatten. De zinnen (2.4), (2.5), (2.6) en (2.7), zijn zinnen die uit enkelvoudige hoofdzinnen en voegwoorden zijn samengesteld. De hier gebruikte voegwoorden zijn *en*, *niet*, en *of*, en de 'voegconstructie' *als* ... *dan*, die we in het vervolg ook als voegwoord zullen beschouwen. Zin (2.4) is bijvoorbeeld samengesteld uit de enkelvoudige hoofdzinnen 'Socrates is sterfelijk' en 'Twee plus twee is gelijk aan vijf', en het voegwoord 'en'. In hoofdstuk 8 zullen we zinnen als 'Socrates is sterfelijk' verder ontleden. Voorlopig zullen we zinnen alleen ontleden in bestanddelen die enkelvoudige constaterende zinnen zijn.

In de volgende paragraaf zullen we een *formele taal* definiëren, waarin we enkelvoudige en samengestelde beweringen kunnen aanduiden. We spreken van een 'formele taal', aangezien het een kunstmatige taal betreft die een mathematische, dus formele, structuur bezit. In die formele taal kunnen we een beperkt aantal voegwoorden aanduiden, te weten *niet*, *en*, *of*, *als* ... *dan* en *dan en slechts dan als*. Het laatstgenoemde voegwoord treft men vrijwel uitsluitend in wiskundige taal aan.

De genoemde voegwoorden krijgen een exact omschreven betekenis toegewezen die ontleend is aan het gebruik van die voegwoorden in wiskundige taal. De betekenissen die op deze wijze worden verkregen, wijken soms iets af van de betekenissen die de voegwoorden in het Nederlands kunnen bezitten. Omgekeerd kunnen sommige voegwoorden uit het Nederlands niet precies worden weergegeven door die uit de formele taal. De reden hiervoor is dat deze voegwoorden in het normale taalgebruik een te vage betekenis hebben voor nauwkeurig gebruik in de logica.

Aan de hand van een aantal voorbeelden zullen we laten zien dat voegwoorden zoals 'en', 'of', 'niet', 'als ... dan', 'maar', 'omdat' en 'daar' in het normale taalgebruik soms een vage betekenis hebben of zelfs meerdere betekenissen bezitten. Vergelijk om te beginnen het gebruik van het voegwoord 'maar' in de volgende twee zinnen:

(2.12) Cees is een uiterst linkse marxist, maar hij heeft gevoel voor humor.

(2.13) Vijf is een natuurlijk getal, maar vijf is ook een geheel getal.

In zin (2.12) wordt door het gebruik van het voegwoord 'maar' een contrast uitgedrukt, alsof we verbaasd zouden zijn dat Cees ook gevoel voor humor heeft. In zin (2.13) is dit contrast geheel afwezig. Het is ook onduidelijk of we met (2.13) iets anders bedoelen dan met:

(2.14) Vijf is een natuurlijk getal, en vijf is ook een geheel getal.

Uit onderstaande voorbeelden blijkt dat ook voegwoorden als ‘en’, ‘of’ en ‘als . . . dan’ in het dagelijkse taalgebruik anders geïnterpreteerd kunnen worden, dan de schrijver of spreker wellicht bedoelde.

(2.15) Als ik gisteren uit het raam van de 13^{e} verdieping was gesprongen, dan was ik gewond geraakt.

(2.16) Als Joost van den Vondel gisteren uit het raam van de 13^{e} verdieping was gesprongen, dan was hij in een vogel veranderd.

(2.17) Als Amsterdam de hoofdstad van Nederland is, dan is zeven een priemgetal.

(2.18) Marie is getrouwd en Marie heeft een baby gekregen.

(2.19) Marie heeft een baby gekregen en Marie is getrouwd.

(2.20) Ruud komt met de auto of Ruud komt te voet.

(2.21) Men kan geld schenken aan het Rode Kruis of aan de Hartstichting.

De beweringen (2.15) en (2.16) hebben dezelfde structuur en de enkelvoudige beweringen waaruit deze beweringen bestaan, zijn alle onjuist. Immers, evenmin als Joost van den Vondel ben ik gisteren uit het raam gesprongen; ook ben ik niet gewond en is Vondel niet in een vogel veranderd. Echter bewering (2.15) zal in het algemeen als waar ervaren worden, terwijl de meeste Nederlandse taalgebruikers bewering (2.16) als onwaar of als onzinnig zullen classificeren. Iets soortgelijks geldt voor bewering (2.17). De beide enkelvoudige beweringen zijn waar, maar omdat elk verband ertussen ontbreekt, wordt de samengestelde ‘als . . . dan’-bewering als onwaar of zelfs als onzinnig ervaren. Deze voorbeelden tonen aan, dat in het dagelijkse taalgebruik het waar of onwaar zijn van de samengestelde bewering

(2.22) Als p dan q.

niet alleen afhankelijk is van de waarheidswaarden van p en q, maar ook van het verband tussen beide beweringen. Filosofisch is het zeer relevant dit verband tussen p en q te bestuderen en nader te definiëren. Voor het gebruik in de wiskunde en voor veel toepassingen in de informatica is het echter voldoende om de waarheidswaarde van bewering (2.22) slechts te laten afhangen van de waarheidswaarden van p en q en niet van een ‘verband’ tussen p en q, hoe dan ook gedefinieerd. Men spreekt van de *materiële implicatie*, in tegenstelling tot de *strikte implicatie* waarbij de waarheidswaarde van de bewering afhankelijk is van de waarheidswaarden van p en q, en van de vraag of er een bepaald verband tussen p en q bestaat. Als we de materiële implicatie hanteren, hetgeen we overigens in de rest van dit boek zullen doen, dan moeten we de zinnen (2.15), (2.16) en (2.17) als waar beoordelen. Dit is in overeenstemming met de wiskundige praktijk (vergelijk bijvoorbeeld de zinnen (2.16) en (2.6)). Gebruiken we echter de strikte implicatie dan is alleen zin (2.15) waar.

In de voorbeelden (2.18) en (2.19) suggereert het gebruik van het voegwoord 'en' een tijdsvolgorde. Dit heeft tot gevolg dat deze beweringen, ofschoon ze beide uit dezelfde enkelvoudige beweringen zijn opgebouwd, als enigszins verschillend worden ervaren. In de wiskunde speelt de tijd geen of nauwelijks een rol. In een uitspraak als

(2.23) Het getal e is irrationaal en 5 is een positief geheel getal.

ontbreekt aan het voegwoord 'en' elk tijdsaspect. In de logica zal het voegwoord *en* dan ook geen tijdsaspect bezitten.

De zinnen (2.20) en (2.21) laten zien hoe het voegwoord 'of' in het Nederlands op verschillende manieren kan worden gebruikt. Volgens bewering (2.20) komt Ruud met de auto of te voet, maar niet tegelijkertijd met de auto èn te voet. Bewering (2.21) stelt dat men geld kan schenken aan het Rode Kruis of aan de Hartstichting, maar ook aan beide instellingen. Men zegt dat het voegwoord 'of' in voorbeeld (2.20) *exclusief* wordt gebruikt en in voorbeeld (2.21) *inclusief*. Het voegwoord 'of' wordt in het normale taalgebruik dus afwisselend inclusief en exclusief gebruikt. In de logica zullen we ons echter bedienen van het inclusieve voegwoord *of*.

Resumerend: in de formele taal van de propositielogica zullen we beschikken over de voegwoorden *niet*, *en*, *of*, *als* ... *dan* en *dan en slechts dan als*. Daarbij is gekozen voor een tijdsonafhankelijk *en*, een inclusief *of* en een materiële implicatie *als* ... *dan*. Deze keuze is gebaseerd op de wijze waarop de voegwoorden in de wiskunde (en ook in veel andere wetenschappen) worden gebruikt. In plaats van over 'voegwoorden' zullen we voortaan spreken over *connectieven*. *Niet* wordt dan een *éénplaatsig* connectief genoemd, omdat het uit één bewering een nieuwe bewering vormt. De overige connectieven noemt men *tweeplaatsig*, omdat deze uit twee beweringen een nieuwe bewering kunnen vormen.

2.2 De taal van de propositielogica

In deze paragraaf zullen we een formele taal invoeren: de taal van de propositielogica. De 'zinnen' uit deze taal zullen proposities (beweringen) zijn. Een formele taal lijkt in een aantal opzichten op een natuurlijke taal. Zij bezit een *alfabet* waarin de symbolen die in de taal worden gebruikt, zijn vastgelegd, en *grammaticale regels* aan de hand waarvan kan worden vastgesteld of een reeks symbolen een welgevormde uitdrukking van die taal voorstelt. De beschrijving van het alfabet en de grammaticale regels van de taal van de propositielogica vormen samen de *syntaxis* van de propositielogica.

Nu we een taal gaan definiëren, moeten we oppassen dat we niet in de war raken. Immers, het boek dat u nu aan het lezen bent, is óók in een taal geschreven: Nederlands aangevuld met wiskundige taal. De taal van de propositielogica zullen we in dit dialect van het Nederlands beschrijven. De taal van de propositielogica wordt de *objecttaal* genoemd, aangezien zij het 'object' van het beschrijven is. Het met wiskundige taal aangevulde Nederlands waarin die beschrijving plaatsvindt, noemt men de *metataal*.

In het vervolg zullen we de taal van de propositielogica aanduiden als $\mathcal{P}$ ($\mathcal{P}$ropositielogica). In $\mathcal{P}$ worden connectieven aangeduid met symbolen. Enkelvoudige beweringen worden weergegeven door zogenaamde *propositiesymbolen*, waarvoor we $p_0, p_1, \ldots$ zullen gebruiken. Samengestelde beweringen kunnen dan worden geschreven als een combinatie van connectieven, propositiesymbolen en *haakjes*. Deze haakjes hebben dezelfde functie als in wiskundige formules. Dienen ze er in de wiskunde voor om ervoor te zorgen dat de operaties op de juiste termen worden toegepast, in $\mathcal{P}$ zorgen zij ervoor dat het duidelijk is welke proposities door de connectieven met elkaar worden verbonden.

2.2.1 DEFINITIE **Alfabet van $\mathcal{P}$**
Het alfabet van de objecttaal $\mathcal{P}$ is onderverdeeld in drie categorieën die respectievelijk bestaan uit de volgende symbolen:

1. propositiesymbolen: $p_0, p_1, \ldots$.
2. connectieven: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$.
3. haakjes: (,) .

De connectieven $\neg$, $\wedge$, $\vee$, $\rightarrow$ en $\leftrightarrow$ worden traditioneel met de namen *negatie*, *conjunctie*, *disjunctie*, *implicatie* en *equivalentie* aangeduid (zie ook tabel 2.1). De negatie $\neg$ is een éénplaatsig connectief, terwijl de overige connectieven tweeplaatsig zijn.

Net zomin als de Nederlandse taal bestaat uit alle rijtjes van letters of woorden, bestaat de taal $\mathcal{P}$ uit alle rijtjes van tekens uit het alfabet, zoals gedefinieerd in definitie 2.2.1. Alleen rijtjes die aan bepaalde voorwaarden voldoen —zogenaamd *syntactisch welgevormd* zijn— zijn toegestaan. De syntaxis

connectief	benaming	'betekenis'
$\neg$	negatie	niet
$\wedge$	conjunctie	en
$\vee$	disjunctie	of
$\rightarrow$	implicatie	als ... dan
$\leftrightarrow$	equivalentie	dan en slechts dan als

Tabel 2.1: De connectieven van $\mathcal{P}$.

voorziet in een beschrijving van alle syntactisch welgevormde rijtjes. Om structuur aan te brengen in de syntaxis van de Nederlandse taal, wordt een groot aantal *syntactische categorieën* onderscheiden. Voorbeelden hiervan zijn: de lidwoorden, de zelfstandige naamwoorden, de werkwoorden, de naamwoordelijke zinsdelen en bijwoordelijke bepalingen. In de syntaxis van $\mathcal{P}$ onderscheiden we slechts één syntactische categorie, namelijk die van de *formules.* Een formule is een symboolrij over het alfabet van $\mathcal{P}$ die aan bepaalde voorwaarden voldoet. Aangezien een formule informeel gesproken een enkelvoudige of samengestelde propositie (bewering) aanduidt, zal de verzameling van alle formules worden genoteerd als *PROP* (*PROP*osities). We zullen bovendien de woorden 'propositie' en 'formule' door elkaar gebruiken. Merk op dat we een onderscheid maken tussen de taal $\mathcal{P}$ en de verzameling *PROP* van alle formules van $\mathcal{P}$. Met $\mathcal{P}$ associëren we alle noties die met de propositielogica te maken hebben, waaronder de syntaxis en de semantiek ervan.

2.2.2 DEFINITIE **Formule**
De verzameling PROP is de kleinste verzameling die voldoet aan de volgende eisen:

1. $p_i \in PROP$ *voor alle* $i \in \mathbb{N}$.
2. *Als* $A, B \in PROP$,
   dan $\neg A$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$, $(A \leftrightarrow B) \in PROP$.

De elementen van PROP worden formules *genoemd. Verder noemt men formules van de vorm* p_i atomen *en formules van de vorm* p_i *of* $\neg p_i$ literalen.

In bovenstaande definitie worden de letters A en B gebruikt. Deze letters behoren niet tot het alfabet van $\mathcal{P}$. Het zijn variabelen uit de metataal die als *waarde* een formule uit *PROP* hebben. Dit soort variabelen noemt men daarom *metavariabelen.* De uitdrukking '$(A \wedge B)$' duidt de formule aan die bestaat uit een '(', gevolgd door de formule waarvoor A staat, gevolgd door het connectief '$\wedge$', gevolgd door de formule waarvoor B staat, en tenslotte een ')'. Voor metavariabelen die proposities aanduiden, zullen we altijd hoofdletters gebruiken. Deze kunnen eventueel van een index zijn voorzien.

Een geheel andere metavariabele die we in het vervolg vaak zullen gebruiken, is het symbool $\star$.

2.2.3 DEFINITIE **De metavariabele $\star$**
Het symbool $\star$ is een metavariabele waarvan de waarde altijd een tweeplaatsig logisch connectief is; dit wil zeggen $\star \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.

Uit bovenstaande definitie volgt dat de uitdrukking $(A \star B)$ kan staan voor de formules $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$ of $(A \leftrightarrow B)$.

2.2.4 VOORBEELD Constructie van een formule.
De symboolrij $F = \neg(p_6 \rightarrow (p_7 \vee p_9))$ is een formule. Dit kan men als volgt inzien: $F_1 = p_6$, $F_2 = p_7$ en $F_3 = p_9$ zijn formules volgens het eerste lid van definitie 2.2.2, en $F_4 = (F_2 \vee F_3) = (p_7 \vee p_9)$, $F_5 = (F_1 \rightarrow F_4) = (p_6 \rightarrow (p_7 \vee p_9))$ en $F_6 = \neg F_5 = \neg(p_6 \rightarrow (p_7 \vee p_9))$ zijn formules volgens het tweede lid van definitie 2.2.2.

De reeks F_1, F_2, F_3, F_4, F_5, $F_6 = F$, ook wel een *constructiereeks* van F genoemd, geeft aan op welke wijze de formule F uit de propositiesymbolen p_6, p_7 en p_9 geconstrueerd kan worden. Merk trouwens op dat de gebruikte F_i $(i = 1, \ldots, 6)$ metavariabelen zijn.

De vraag 'Is F een formule?' kan ook als volgt beantwoord worden. $F = \neg F_5$ is een formule als F_5 een formule is. Op haar beurt is $F_5 = (F_1 \rightarrow F_4)$ een formule als F_1 en F_4 formules zijn. F_1 is het propositiesymbool p_6, dus een formule. Verder is $F_4 = (F_2 \vee F_3)$ een formule als F_2 en F_3 formules zijn. Maar F_2 en F_3 zijn respectievelijk de propositiesymbolen p_7 en p_9 en derhalve formules. Volgens definitie 2.2.2 is F dus een formule, zodat $F \in PROP$.

Uit het bovenstaande blijkt dat men voor iedere symboolrij F op effectieve wijze en in een eindig aantal stappen kan nagaan of $F \in PROP$. ■

2.2.5 DEFINITIE **Subformule en echte subformule**

1. *Een formule A is een* subformule *van de formule F als één van de drie onderstaande voorwaarden geldt.*
   (a) $A = F$.
   (b) $F = \neg C$ *en A is een subformule van C.*
   (c) $F = (C \star D)$ *en A is een subformule van C of D.*

2. *Een formule A is een* echte subformule *van de formule F als A een subformule is van F en $A \neq F$.*

2.2.6 VOORBEELD Subformules en echte subformules.
De formules F_1, F_2, F_3, F_4, F_5 en $F_6 = F$ uit voorbeeld 2.2.4 zijn alle subformules van F. Ook geldt dat $F_3 = p_9$ een subformule is van F_3, F_4 en F_5.

De vraag 'Is A een subformule van F?' kan op een effectieve wijze en in een eindig aantal stappen worden beantwoord. Immers, dit is het geval dan en slechts dan als A voorkomt in de constructiereeks van F.

De formules F_1, F_2, F_3, F_4 en F_5 zijn echte subformules van F. Deze komen immers alle in de constructiereeks van F voor, en zijn ongelijk aan F. ■

De definities 2.2.2 en 2.2.5(1) hebben gemeen dat zij op het eerste gezicht circulair lijken. In de eerstgenoemde definitie wordt de verzameling *PROP* gebruikt om *PROP* zelf te definiëren. De tweede definitie gebruikt het begrip subformule in de definitie van dat begrip. Men spreekt van *recursieve definities*. In paragraaf 2.4 wordt deze techniek uitgebreid besproken. Men komt recursieve definities ook vaak in de informatica tegen, namelijk in de vorm van procedures (of functies) die zichzelf aanroepen. Dit worden *recursieve procedures* (of *functies*) genoemd.

We besluiten deze paragraaf met een voorbeeld waarin we laten zien hoe (eenvoudige) Nederlandse zinnen kunnen worden vertaald in de propositielogica.

2.2.7 VOORBEELD Vertalingen van beweringen.
Om beweringen te vertalen naar de propositielogica omkaderen we eerst het hoofdvoegwoord in een zin. Vervolgens vervangen we het omkaderde voegwoord door het ermee corresponderende logische connectief, en omkaderen we de hoofdvoegwoorden in de deelzinnen die ontstaan zijn. Deze stap herhalen we totdat er geen deelzinnen meer zijn waarin een voegwoord kan worden omkaderd. Tenslotte vervangen we de enkelvoudige deelzinnen door geschikte propositiesymbolen.

1. Toos slaapt $\boxed{\text{en}}$ Coby werkt,

   Toos slaapt $\wedge$ Coby werkt,

   $s \wedge w$.

   Hierbij staat s voor 'Toos slaapt', en w voor 'Coby werkt'. In de volgende voorbeelden zullen we dit soort informatie niet expliciet vermelden.

2. $\boxed{\text{Als}}$ het regent, $\boxed{\text{dan}}$ fiets ik niet,

   Het regent $\to$ Ik fiets $\boxed{\text{niet}}$, Het regent $\rightarrow$ $\neg$ ik fiets,

$r \rightarrow \neg f$.

3. Cees drinkt $\boxed{\text{dan en slechts dan}}$ koffie $\boxed{\text{als}}$ hij er een koekje of een glaasje cognac bij krijgt,

   Cees drinkt koffie $\leftrightarrow$ (hij krijgt er een koekje bij $\boxed{\text{of}}$ hij krijgt er een glaasje cognac bij),

   Cees drinkt koffie $\leftrightarrow$ (hij krijgt er een koekje bij $\vee$ hij krijgt er een glaasje cognac bij),

   $c \leftrightarrow (k \vee c)$. ∎

2.3 Structurele inductie

Vaak is het nodig om te bewijzen dat alle formules uit *PROP* een bepaalde eigenschap bezitten. Hiertoe gebruikt men zogenaamde *structurele inductie*. Dit betekent dat men eerst bewijst dat propositiesymbolen de eigenschap hebben, en vervolgens bewijst dat een samengestelde formule de eigenschap heeft als de samenstellende subformules die eigenschap bezitten. In deze paragraaf wordt deze bewijsmethode nader uitgelegd.

Eigenschappen van natuurlijke getallen kunnen vaak worden bewezen met volledige inductie. Nemen we bijvoorbeeld de eigenschap

$$E(n): \quad \sum_{j=0}^{n} j = \frac{n(n+1)}{2},$$

dan behoeven we om te bewijzen dat E geldt voor alle natuurlijke getallen, slechts aan te tonen dat:

1. E voor het getal 0 geldt, ofwel dat $E(0)$ waar is. Dit noemt men de *inductiebasis*.

2. Als E voor het natuurlijke getal n geldt, E dan ook geldt voor het getal $n+1$. Anders geformuleerd, als $E(n)$ waar is, moet ook $E(n+1)$ waar zijn. Dit noemt men de *inductiestap*.

De aanname '$E(n)$ is waar' in clausule 2 noemt men de *inductieveronderstelling* of *inductiehypothese*. Soms is het handiger om als inductieveronderstelling de variant '$E(k)$ geldt voor $0 \leq k \leq n$' te kiezen. Als men de inductiebasis en de inductiestap bewezen heeft, dan volgt uit het beginsel van volledige inductie dat E geldt voor alle natuurlijke getallen.

Voor de verzameling $PROP$ geldt een analoog beginsel, namelijk het beginsel van *inductie naar de opbouw van de formules in* $PROP$. Ter onderscheiding van volledige inductie, noemt men deze vorm van inductie ook wel *structurele inductie over* $PROP$. De inductie vindt immers plaats over de structuur (de opbouw) van de formules.

2.3.1 Definitie **Structurele inductie over** $PROP$.
Zij P *een eigenschap van formules uit* $PROP$. *Onder een* bewijs door middel van structurele inductie over $PROP$ *dat* P *geldt voor alle* $F \in PROP$, *verstaat men een bewijs van de volgende clausules:*

1. $P(p_i)$ *geldt voor alle propositiesymbolen* $p_i \in PROP$ $(i \in \mathbb{N})$;
2. *Uit de aanname dat* $P(A)$ *geldt, volgt dat* $P(\neg A)$ *geldt* $(A \in PROP)$;
3. *Uit de aanname dat* $P(A)$ *en* $P(B)$ *gelden, volgt dat* $P((A \star B))$ *geldt* $(A, B \in PROP)$.

Clausule 1 noemt men de inductiebasis *De clausules 2 en 3 vormen samen de* inductiestap. *De aanname in clausule 2 dat* $P(A)$ *geldt, en die in clausule 3 dat* $P(A)$ *en* $P(B)$ *gelden, worden de* inductiehypothesen *genoemd.*

In stelling 2.3.5 wordt bewezen dat inductie over $PROP$ een geldige bewijsmethode is. Dat wil zeggen dat de geldigheid van de drie clausules in definitie 2.3.1 inderdaad garandeert dat de betreffende eigenschap geldt voor alle formules in $PROP$.

2.3.2 Voorbeeld Structurele inductie.
Zij $F \in PROP$. Laten we zeggen dat F de eigenschap P bezit, indien het aantal haakjes in F *even* is. We zullen met behulp van structurele inductie laten zien dat alle formules in $PROP$ deze eigenschap bezitten. In het onderstaande bewijs duiden we met $\#(F)$ het aantal haakjes in F aan.

1. Zij $F = p_i$ voor zekere $i \in \mathbb{N}$, ofwel F is een propositiesymbool. Dan geldt zeker $P(p_i)$, aangezien $\#(p_i) = 0$.
2. Zij $F = \neg A$ voor zekere $A \in PROP$, en neem aan dat $P(A)$ geldt, ofwel dat $\#(A)$ even is (inductiehypothese). Hieruit volgt dat $P(\neg A)$, aangezien $\#(\neg A) = \#(A)$.
3. Zij $F = (A \star B)$ voor zekere $A, B \in PROP$, en neem aan dat zowel $P(A)$ als $P(B)$ geldt, ofwel dat $\#(A)$ en $\#(B)$ beide even zijn (inductiehypothese). Maar dan is $\#((A \star B)) = \#(A) + \#(B) + 2$ ook even, en geldt derhalve $P((A \star B))$.

Hiermee is P met structurele inductie bewezen. ■

In het bewijs van de volgende stelling maken we gebruik van het begrip *complexiteit* van een formule.

2.3.3 DEFINITIE **Complexiteit van een formule**
De complexiteit *van een formule* $F \in PROP$*, notatie* $comp(F)$*, is gedefinieerd als het aantal connectieven dat in* F *voorkomt.*

2.3.4 VOORBEELD Complexiteit van formules.

1. $comp(p_5) = 0$;
2. $comp(((p_7 \vee p_3) \leftrightarrow p_1)) = 2$;
3. $comp(\neg(p_1 \vee (p_4 \vee p_2))) = 3$. ■

2.3.5 STELLING *Zij* P *een eigenschap van formules uit* $PROP$*. Als* P *met structurele inductie over* $PROP$ *bewezen kan worden, dan geldt* P *voor alle formules uit* $PROP$*.*

BEWIJS Neem aan dat de clausules 1, 2 en 3 van definitie 2.3.1 bewezen zijn. Met behulp van volledige inductie naar $comp(F)$ zullen we dan bewijzen dat P geldt voor alle formules uit $PROP$.

1. Zij $comp(F) = 0$. Hieruit volgt dat F een propositiesymbool is. Dit betekent dat $F = p_i$ voor zekere $i \in \mathbb{N}$. Uit clausule 1 volgt nu direct dat $P(F)$ geldt.
2. Zij $comp(F) = n+1$. De inductiehypothese luidt: De eigenschap P geldt voor alle formules C waarvoor $0 \leq comp(C) \leq n$. Voor F doen zich de volgende twee mogelijkheden voor:
   (a) $F = \neg A$. Dan geldt dat $comp(A) = n$. Uit de inductiehypothese volgt $P(A)$. Maar dan geeft toepassing van clausule 2 dat $P(\neg A)$, ofwel $P(F)$.
   (b) $F = (A \star B)$. Dan geldt dat $comp(A)$, $comp(B) \leq n$. Uit de inductiehypothese volgt $P(A)$ en $P(B)$. Toepassing van clausule 3 levert dan $P((A \star B))$, ofwel $P(F)$.

   Hiermee is $P(F)$ aangetoond en de inductiestap voltooid.

Uit het beginsel van volledige inductie volgt nu dat alle formules $F \in PROP$ de eigenschap P bezitten. ■

Tot slot van deze paragraaf geven we nog een voorbeeld van een toepassing van structurele inductie over $PROP$.

2.3.6 VOORBEELD Structurele inductie.
Zeg dat $F \in PROP$ de eigenschap P bezit, indien F hoogstens $2 \cdot comp(F) + 1$ subformules heeft.

Let op het woord 'hoogstens'. Om eigenschap P te bezitten kan een formule F precies $2 \cdot comp(F) + 1$ subformules hebben, maar het mogen er ook minder zijn. Beschouw bijvoorbeeld $\neg p_0$, $(p_1 \wedge p_2)$ en $(p_1 \vee p_1)$. Al deze formules hebben complexiteit 1. De eerste en de derde formule bezitten ieder slechts 2 subformules, terwijl de tweede er 3 bezit.

Het bewijs dat $P(F)$ geldt voor alle $F \in PROP$ verloopt nu aldus.

1. Zij $F = p_i$ voor zekere $i \in \mathbb{N}$, ofwel F is een propositiesymbool. In dit geval geldt zeker $P(F)$ want het aantal subformules van F is gelijk aan $1 = 2 \cdot comp(F) + 1$.
2. Zij $F = \neg A$, waarbij $A \in PROP$. Veronderstel dat $P(A)$. Dit betekent dat A hoogstens $2 \cdot comp(A) + 1$ subformules bezit. Aangezien $comp(\neg A) = comp(A) + 1$ en een subformule van $\neg A$ ofwel een subformule van A, ofwel $\neg A$ zelf moet zijn, heeft F hoogstens $(2 \cdot comp(A) + 1) + 1 < 2(comp(A) + 1) + 1 = 2 \cdot comp(F) + 1$ subformules. Derhalve geldt $P(F)$.
3. Zij $F = (A \star B)$, waarbij $A, B \in PROP$. Veronderstel dat $P(A)$ en $P(B)$. Dit betekent dat A en B respectievelijk hoogstens $2 \cdot comp(A) + 1$ en $2 \cdot comp(B) + 1$ subformules bezitten. Enerzijds geldt dat $comp((A \star B)) = comp(A) + comp(B) + 1$. Anderzijds geldt voor iedere subformule C van $(A \star B)$:
   (a) C is een subformule van A, of
   (b) C is een subformule van B, of
   (c) $C = (A \star B)$.

   De formule F bezit dus hoogstens $(2 \cdot comp(A) + 1) + (2 \cdot comp(B) + 1) + 1 = 2 \cdot comp(F) + 1$ subformules. Derhalve geldt $P(F)$.

We hebben dus met behulp van structurele inductie laten zien dat $P(F)$ geldt voor alle $F \in PROP$. ■

2.4 Recursieve definities

Functies over de natuurlijke getallen kunnen soms gedefinieerd worden door middel van recurrente betrekkingen. In een recurrente betrekking wordt een functie gedefinieerd in termen van zichzelf. Een standaardvoorbeeld is de definitie van de machtsverheffing $f(n) = a^n$ $(a \in \mathbb{N}^+)$. De recurrente betrekking voor f luidt:

$$f(n+1) \quad = \quad a.f(n) \qquad (n \in \mathbb{N})$$

terwijl $f(0) = 1$ een geschikte randvoorwaarde is. We merken op dat in de bovenstaande betrekking het symbool f zowel links als rechts van het symbool $=$ voorkomt: f wordt dus in termen van zichzelf gedefinieerd. Met behulp van volledige inductie is eenvoudig te bewijzen dat de f die aan deze recurrente betrekking met de gegeven randvoorwaarde voldoet, een functie $f : \mathbb{N} \rightarrow \mathbb{N}$ is, waarvoor $f(n) = a^n$ voor alle $n \in \mathbb{N}$.

Een ander bekend voorbeeld is de rij van Fibonnacci:

$$1, 1, 2, 3, 5, 8, 13, 21, 34, \ldots$$

Vanaf de derde term geldt dat iedere volgende term gelijk is aan de som van de twee direct eraan voorafgaande termen. Noteren we het n^{e} getal van Fibonnacci als $F(n)$, dan kunnen we de rij als een functie op de natuurlijke getallen beschouwen. Deze functie F wordt door de volgende recurrente betrekking met randvoorwaarden gedefinieerd:

$$\begin{aligned} F(0) &= 1, \\ F(1) &= 1, \\ F(n+2) &= F(n) + F(n+1) \qquad (n \in \mathbb{N}). \end{aligned}$$

Het is wederom eenvoudig om met volledige inductie te bewijzen dat dit recurrente stelsel een unieke oplossing F heeft met $F(0) = 1$, $F(1) = 1$, $F(2) = 2$, $F(3) = 3$, $F(4) = 5$, $F(5) = 8, \ldots$.

De volgende stelling geeft aan op welke manier men met behulp van bepaalde stelsels recurrente betrekkingen plus randvoorwaarden functies over *PROP* op een unieke en éénduidige manier kan definiëren.

2.4.1 STELLING **Recursieve definities van functies over *PROP***

Zij gegeven een niet-lege verzameling W en een vijftal functies:

$$\begin{aligned} h_\neg &: W \rightarrow W, \qquad \textit{en} \\ h_\star &: W \times W \rightarrow W. \end{aligned}$$

Dan heeft het stelsel recurrente betrekkingen

$$\begin{aligned} f(\neg A) &= h_\neg(f(A)), \\ f((A \star B)) &= h_\star(f(A), f(B)) \qquad (A, B \in \mathit{PROP}), \end{aligned}$$

met als randvoorwaarde dat $f(p_i)$ *voor ieder propositiesymbool* p_i $(i \in \mathbb{N})$ *een éénduidig bepaald element van* W *is, precies één oplossing* $f : PROP \to W$ *die aan alle* $F \in PROP$ *een unieke functiewaarde* $f(F)$ *toekent.*

Het bewijs van deze stelling, dat tamelijk gecompliceerd is, zullen we hier achterwege laten. In concrete gevallen kan men over het algemeen eenvoudig inzien dat een gegeven recurrent stelsel een unieke oplossing f bezit en kan men voor willekeurige $F \in PROP$ gemakkelijk de functiewaarde $f(F)$ bepalen.

2.4.2 VOORBEELD Lengte van een formule.
Zij gegeven het volgende stelsel recurrente betrekkingen met randvoorwaarden:

$$\begin{aligned} f(p_i) &= 1, \\ f(\neg A) &= f(A) + 1, \\ f((A \star B)) &= f(A) + f(B) + 3. \end{aligned}$$

Volgens stelling 2.4.1 heeft dit recurrente stelsel een unieke oplossing f. Iedere functiewaarde van f kan met behulp van de recurrente betrekkingen en de randvoorwaarden worden berekend. Zij bijvoorbeeld $F = (p_1 \wedge (p_2 \to p_3))$, dan kunnen we $f(F)$ als volgt berekenen:

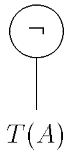
$$\begin{aligned} f((p_1 \wedge (p_2 \to p_3))) &= f(p_1) + f((p_2 \to p_3)) + 3, \\ &= f(p_1) + (f(p_2) + f(p_3) + 3) + 3, \\ &= 1 + 1 + 1 + 3 + 3, \\ &= 9. \end{aligned}$$

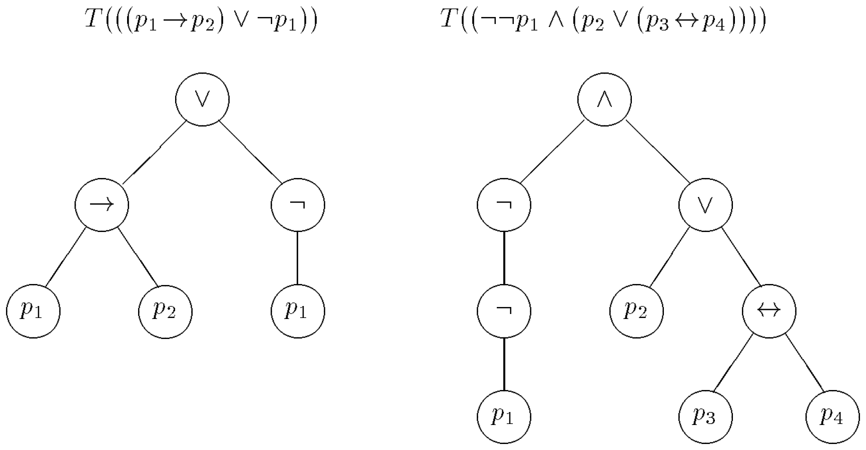
De functie f voegt aan een formule $F \in PROP$ het aantal symbolen dat in F voorkomt, dus de *lengte* ervan, toe. ∎

2.4.3 VOORBEELD Ontledingsboom.
De functie T, die recursief wordt gedefinieerd, voegt aan iedere formule $F \in PROP$ haar zogenaamde *ontledingsboom* $T(F)$ toe.

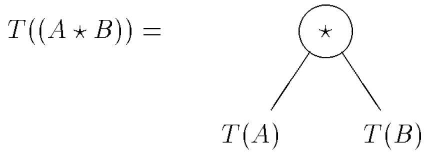
$T(p_i) =$

$T(\neg A) =$





Figuur 2.1: Twee ontledingsbomen.



Volgens stelling 2.4.1 is $T(F)$ voor alle $F \in PROP$ op eenduidige wijze gedefinieerd. In figuur 2.1 zijn twee ontledingsbomen weergegeven. ∎

2.5 Notationele kwesties

Voor de menselijke lezer zouden veel haakjes in formules beter weggelaten of door andere soorten haakjes, zoals vierkante haakjes of accoladen, vervangen kunnen worden. Ook de notatie van de propositiesymbolen draagt niet bij tot het leesgemak. Teneinde de leesbaarheid van formules te vergroten, voeren we de volgende notatieconventies in.

1. De propositiesymbolen $p_0, p_1, p_2, \ldots$ worden vervangen door $p, q, r, \ldots$. De symbolen $p, q, r, \ldots$ zijn *metavariabelen* die propositiesymbolen aanduiden.

2. Analoog aan de prioriteitsregels voor de operaties in de rekenkunde ('Meneer Van Dale Wacht Op Antwoord'), zijn er ook precedentieregels voor

prioriteit	connectief
1	$\neg$
2	$\wedge, \vee$
3	$\rightarrow, \leftrightarrow$

Tabel 2.2: De prioriteiten van de connectieven.

de connectieven van $\mathcal{P}$. Deze zijn weergegeven in tabel 2.2. Hierbij geldt dat de prioriteit van een connectief afneemt bij het oplopen van de rangnummers. Merk op dat $\wedge$ en $\vee$, respectievelijk $\rightarrow$ en $\leftrightarrow$ gelijke prioriteiten bezitten. Door toepassing van de gegeven prioriteitsregels kunnen sommige haakjes worden weggelaten. Zo is bijvoorbeeld $\neg p \wedge q \rightarrow r \vee s$ gelijkwaardig aan $((\neg p \wedge q) \rightarrow (r \vee s))$. Immers, de tabel schrijft voor dat de $\wedge$ en de $\vee$ voorafgaan aan de $\rightarrow$. De lezing van formule $p \rightarrow q \leftrightarrow r$ is echter niet eenduidig. Hier moeten haakjes worden geplaatst: $\rightarrow$ en $\leftrightarrow$ hebben een gelijke prioriteit. De formules $p \rightarrow (q \leftrightarrow r)$ en $(p \rightarrow q) \leftrightarrow r$ zijn niet alleen als tekenrij verschillend, ze zijn ook niet onder gelijke omstandigheden waar. De haakjes zijn dus echt noodzakelijk!

3. Niet alleen de haakjes (en) mogen worden gebruikt, ook de rechte haakjes [en], en de accoladen { en } mogen paarsgewijs worden toegepast. Zo mag bijvoorbeeld $((p \rightarrow q) \rightarrow r) \rightarrow s$ worden genoteerd als $[(p \rightarrow q) \rightarrow r] \rightarrow s$.

In het vervolg van dit boek zullen we deze notatieconventies zoveel mogelijk toepassen. We wijzen er met klem op dat bovenstaande notatieconventies zijn ingevoerd voor het gemak van de lezer. Het begrip formule blijft derhalve zoals vastgelegd in definitie 2.2.2. De notaties voor formules die voortkomen door toepassing van deze conventies, dienen te worden beschouwd als *meta-notaties*, dus als metatalige aanduidingen van formules uit de objecttaal.

In de notatie van formules zoals tot nu toe gebruikt, worden de tweeplaatsige connectieven tussen de operanden gezet en de éénplaatsige direct voor de betreffende operand. De haakjes worden gebruikt om aan te geven welke operanden bij een bepaald connectief behoren. Bijvoorbeeld in $p \wedge (q \vee p)$ worden de haakjes om $q \vee p$ gebruikt om aan te geven dat deze subformule een operand is van het connectief $\wedge$.

Met behulp van de zogenaamde *prefix-* of *postfixnotatie* kan het gebruik van haakjes vermeden worden. Bij de prefixnotatie wordt een tweeplaatsig connectief vóór zijn beide operanden en bij de postfixnotatie achter zijn beide operanden gezet. De éénplaatsige operator staat in prefixnotatie vóór en bij postfixnotatie achter zijn operand. Men noemt de prefixnotatie ook wel de *Poolse notatie* en de postfixnotatie de *omgekeerde Poolse notatie*.

2.5.1 VOORBEELD Voorbeelden van prefixnotatie zijn:

1. $\rightarrow p\,q$ in plaats van $p \rightarrow q$;
2. $\vee \wedge p\,q\,r$ in plaats van $(p \wedge q) \vee r$;
3. $\leftrightarrow \wedge \neg p \neg\neg q \rightarrow p\,q$ in plaats van $(\neg p \wedge \neg\neg q) \leftrightarrow (p \rightarrow q)$. ■

2.5.2 VOORBEELD Voorbeelden van postfixnotatie zijn:

1. $p\,q \rightarrow$ in plaats van $p \rightarrow q$;
2. $p\,q \vee r \wedge$ in plaats van $(p \vee q) \wedge r$;
3. $p \neg q \neg\neg \vee p\,q \rightarrow \leftrightarrow$ in plaats van $(\neg p \vee \neg\neg q) \leftrightarrow (p \rightarrow q)$. ■

In de volgende definitie wordt een recursieve definitie gegeven van de prefix- en de postfixnotatie van formules $F \in PROP$.

2.5.3 DEFINITIE **Prefix- en postfixnotatie**
Zij PRE *en* POST *respectievelijk de verzameling van prefixnotaties en de verzameling van postfixnotaties voor de formules uit PROP.*

De functie $\pi : PROP \rightarrow \mathrm{PRE}$ *waarvan de functiewaarde* $\pi(F)$ *gelijk is aan de prefixnotatie van* F*, kan als volgt recursief worden gedefinieerd:*

$$\begin{aligned} \pi(p_i) &= p_i, \\ \pi(\neg A) &= \neg\,\pi(A), \\ \pi((A \star B)) &= \star\,\pi(A)\,\pi(B). \end{aligned}$$

De functie $\Pi : PROP \rightarrow \mathrm{POST}$ *waarvan de functiewaarde* $\Pi(F)$ *gelijk is aan de postfixnotatie van* F*, kan eveneens recursief worden gedefinieerd:*

$$\begin{aligned} \Pi(p_i) &= p_i, \\ \Pi(\neg A) &= \Pi(A)\,\neg, \\ \Pi((A \star B)) &= \Pi(A)\,\Pi(B)\,\star. \end{aligned}$$

De functies π en Π zijn gedefinieerd door middel van recursie over *PROP*. Volgens stelling 2.4.1 zijn beide functies dus goed gedefinieerd.

2.5.4 VOORBEELD Prefix- en postfixnotatie.
Met behulp van de definitie van de functies π en Π kan men stapsgewijs de prefix- en postfixnotatie van een gegeven formule F berekenen. Als voorbeeld nemen we de formule $F = (p \rightarrow q) \wedge (r \vee (s \rightarrow t))$.

1. Prefixnotatie.

$$\begin{aligned}\pi(F) &= \wedge\, \pi(p{\rightarrow}q)\, \pi(r \vee (s{\rightarrow}t)),\\ &= \wedge{\rightarrow}p\, q\, \pi(r \vee (s{\rightarrow}t)),\\ &= \wedge{\rightarrow}p\, q \vee r\, \pi(s{\rightarrow}t),\\ &= \wedge{\rightarrow}p\, q \vee r{\rightarrow}s\, t.\end{aligned}$$

2. Postfixnotatie.

$$\begin{aligned}\Pi(F) &= \Pi(p{\rightarrow}q)\, \Pi(r \vee (s{\rightarrow}t))\wedge,\\ &= p\, q{\rightarrow}\, \Pi(r \vee (s{\rightarrow}t))\wedge,\\ &= p\, q{\rightarrow}r\, \Pi(s{\rightarrow}t) \vee \wedge,\\ &= p\, q{\rightarrow}r\, s\, t{\rightarrow} \vee \wedge.\end{aligned}$$ ■

2.6 Opgaven

1. Vertaal de volgende zinnen in de *propositielogica*:
   (a) *Als Jan droomt, dan slaapt hij.*
   (b) *Slapen is een noodzakelijke voorwaarde voor dromen.*
   (Beschouw hierbij *slapen* en *dromen* als proposities.)
   (c) *Niet drinken is een voldoende voorwaarde om niet dronken te worden.*
   (d) *Als ik gedronken heb en toch autorijd, dan loop ik kans op een forse boete.*
   (e) *Als ik gedronken heb en toch autorijd, dan loop ik kans op een forse boete, behalve als het promillage alcohol in mijn bloed onder een bepaalde waarde ligt.*
   (f) *Als ik autorijd en ik ben dronken of onder invloed van XTC, dan maak ik de weg zeer onveilig.*

2. Geef een recursieve definitie van de functie *comp*.

3. Definieer de *nestingsdiepte* van een formule als het maximum aantal keren dat paren haakjes om een propositiesymbool staan. Zo is bijvoorbeeld de nestingsdiepte van $((p_0 \vee p_1){\rightarrow}(p_2 \wedge p_3))$ gelijk aan 2, en die van $((p_0 \vee p_1){\rightarrow}(p_2 \wedge (p_3 \vee p_4)))$ gelijk aan 3. Geef een recursieve definitie van de functie *nestingsdiepte*.

4. Bewijs met structurele inductie dat de nestingsdiepte van een formule (zie opgave 3) altijd kleiner dan of gelijk is aan de complexiteit van de formule.

5. Probeer volgens het schema van stelling 2.4.1 een functie te definiëren die voor een formule F de verzameling echte subformules van F oplevert. Waarom lukt dit niet? Pas stelling 2.4.1 zodanig aan dat dit wel mogelijk wordt.

Hoofdstuk 3

Semantiek van de Propositielogica

In dit hoofdstuk wordt de semantiek (betekenistheorie) van de propositielogica behandeld. In de paragrafen 3.1 en 3.2 worden de noties *valuatie*, *model* en *logisch gevolg* ingevoerd. Deze begrippen worden in paragraaf 3.3 toegepast om redeneringen, gegeven in het Nederlands, op hun geldigheid te toetsen.

3.1 Valuaties

De 'objecten' die door de formules van onze objecttaal $\mathcal{P}$ aangeduid worden, zijn beweringen die waar of onwaar kunnen zijn. Door dit uitgangspunt beperken we ons tot een bepaald soort logica, namelijk de *klassieke tweewaardige* propositielogica. 'Klassiek' omdat we aannemen dat iedere bewering een waarheidswaarde heeft en 'tweewaardig' omdat we uitgaan van twee waarheidswaarden, namelijk *waar* en *onwaar*. Deze waarden worden ook wel aangeduid met respectievelijk 1 en 0, en worden *Booleaanse* waarden genoemd. De verzameling $\{0, 1\}$ wordt genoteerd als $\mathbb{B}$.

De vraag is hoe we de waarheidswaarde van een formule $F \in PROP$ kunnen bepalen. Ter vergelijking kijken we naar de analyse: ook hier hebben we te maken met formules. Beschouw bijvoorbeeld de formule $x^2 + 3xy + 5y^3$. Van deze formule kunnen we alleen de waarde bepalen als we de waarden van x en y weten. Zijn deze waarden bekend dan kunnen we de bijbehorende waarde van de formule uitrekenen, aangezien we kunnen vermenigvuldigen en optellen.

Keren we terug naar de propositielogica, dan treffen we daar in feite dezelfde situatie aan. Beschouw bijvoorbeeld de formule

$$F = p \vee (q \rightarrow r). \tag{3.1}$$

We kunnen ook in dit geval alleen iets over de waarde van F zeggen, indien we de waarden van p, q en r kennen. Een verschil met de analyse is dat we hier te maken hebben met waarheidswaarden. Ook de operaties zijn anders. Er is

A	B	$\neg A$	$A \wedge B$	$A \vee B$	$A \to B$	$A \leftrightarrow B$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

Tabel 3.1: Waarheidstafels van de logische connectieven.

nu sprake van connectieven. We moeten dus ook weten hoe deze connectieven werken op waarheidswaarden.

Als we de waarheidswaarde van een willekeurige formule $F \in PROP$ willen berekenen, dienen we de waarheidswaarden van alle propositiesymbolen die in F voorkomen, te kennen. Omdat we onze aanpak zo algemeen mogelijk willen houden, introduceren we een functie die waarheidswaarden toekent aan *alle* propositiesymbolen. Zo'n functie wordt een *valuatie* of *interpretatie* genoemd. Een valuatie zal ons in staat stellen om aan iedere formule, ongeacht welke propositiesymbolen daarin voorkomen, een waarheidswaarde toe te kennen.

3.1.1 DEFINITIE **Valuatie**
Een valuatie *is een functie v die aan ieder propositiesymbool p van $\mathcal{P}$ een waarheidswaarde $v(p) \in \mathbb{B}$ toekent.*

Stel nu dat een bepaalde valuatie v gegeven is, waarvoor geldt dat:

$$\begin{aligned} v(p) &= 0, \\ v(q) &= 1, \\ v(r) &= 0. \end{aligned} \tag{3.2}$$

Hoe kunnen we dan de waarheidswaarde van formule 3.1 uitrekenen? We weten immers nog niet hoe de logische connectieven op waarheidswaarden werken. De zogenaamde *waarheidstafels* geven het antwoord op deze vraag. Waarheidstafels kan men vergelijken met de tafels voor het vermenigvuldigen. De waarheidstafels van de logische connectieven kan men aantreffen in tabel 3.1. Aan het einde van deze paragraaf volgt een toelichting op deze tafels.

Nu zijn we in staat om de waarheidswaarde van formule (3.1) met betrekking tot een valuatie die aan (3.2) voldoet, te evalueren. Immers, in rij 3 van de tafel voor de implicatie $\to$ (kolom 6 in tabel 3.1) kunnen we aflezen dat de waarheidswaarde van $(q \to r)$ gelijk moet zijn aan 0. Combineren we dit resultaat met de rest van de formule met gebruikmaking van de tafel voor de disjunctie $\vee$ (kolom 5, rij 1), dan vinden we dat de waarheidswaarde van de gehele formule gelijk is aan 0. We moeten wel bedenken dat dit resultaat afhankelijk is van de gekozen valuatie. Een andere valuatie geeft in het algemeen een andere uitkomst!

Het berekenen van de waarheidswaarde van een gegeven formule met betrekking tot een gegeven valuatie v kan recursief worden gedefinieerd als een functie $v^+ : PROP \to \mathbb{B}$. De naamgeving v^+ duidt op het feit dat deze functie kan worden beschouwd als een *uitbreiding* van de functie v. Hierop komen we terug na definitie 3.1.3. Ten behoeve van de recursieve definitie (zie stelling 2.4.1) geven we eerst een getaltheoretische beschrijving van de logische connectieven.

3.1.2 DEFINITIE **Waarheidsfuncties connectieven**

$$\begin{array}{rcl} f_\neg(x) & = & (1-x), \\ f_\wedge(x,y) & = & \mathsf{min}\,(x,y), \\ f_\vee(x,y) & = & \mathsf{max}\,(x,y), \\ f_\to(x,y) & = & \mathsf{max}\,(1-x,y), \\ f_\leftrightarrow(x,y) & = & \begin{cases} 1 & \textit{als } x=y, \\ 0 & \textit{anders.} \end{cases} \end{array}$$

In de uitdrukking $1-x$ *wordt de waarde van* x *als een natuurlijk getal beschouwd, verder geven de functies* min *en* max *respectievelijk het minimum en het maximum van hun argumentwaarden die daarbij eveneens als natuurlijke getallen worden beschouwd.*

Het is eenvoudig in te zien dat de functies f overeenstemmen met de waarheidstafels uit tabel (3.1). De volgende definitie geeft de recursieve beschrijving van de waarheidsfunctie v^+.

3.1.3 DEFINITIE **Waarheidswaarde van formules**

Zij v *een gegeven valuatie. Aan iedere formule* $F \in PROP$ *wordt door de functie* $v^+ : PROP \to \mathbb{B}$ *een* waarheidswaarde $v^+(F) \in \mathbb{B}$ *toegekend in overeenstemming met de volgende recurrente betrekkingen:*

$$\begin{array}{rcl} v^+(p_i) & = & v(p_i), \\ v^+(\neg A) & = & f_\neg(v^+(A)), \\ v^+(A \star B) & = & f_\star(v^+(A), v^+(B)). \end{array}$$

Uit stelling 2.4.1 volgt dat $v^+(F)$ voor alle $F \in PROP$ eenduidig is gedefinieerd. Uit de definitie blijkt dat voor gegeven v en F de waarde $v^+(F)$ volledig wordt bepaald door de waarheidswaarden die v toekent aan de propositiesymbolen p_i $(i \in \mathbb{N})$. Verder stemmen de functies v^+ en v overeen op propositiesymbolen. Hieruit volgt dat de functie v^+ opgevat kan worden als een *uitbreiding* van v. Omdat v^+ vastligt zodra v is gegeven, zullen we in het vervolg het superscript $+$ in v^+ weglaten.

3.1.4 VOORBEELD Berekening van $v(F)$.

1. Zij F de formule (3.1), en v een valuatie die voldoet aan (3.2), dan geldt:

$$\begin{aligned} v(F) &= v(p \vee (q \rightarrow r)), \\ &= f_{\vee}(v(p), v(q \rightarrow r)), \\ &= f_{\vee}(0, f_{\rightarrow}(v(q), v(r))), \\ &= f_{\vee}(0, f_{\rightarrow}(1, 0)), \\ &= f_{\vee}(0, 0), \\ &= 0. \end{aligned}$$

2. Zij $F = [(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow [p \rightarrow r]$, en laat v als volgt voor de atomen gedefinieerd zijn: $v(p) = 0$, $v(q) = 1$ en $v(r) = 0$. De waarde $v(F)$ wordt als volgt berekend.

$$\begin{aligned} v(F) &= v([(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow [p \rightarrow r]), \\ &= f_{\rightarrow}(v((p \rightarrow q) \wedge (q \rightarrow r)), v(p \rightarrow r)), \\ &= f_{\rightarrow}(f_{\wedge}(v(p \rightarrow q), v(q \rightarrow r)), v(p \rightarrow r)), \\ &= f_{\rightarrow}(f_{\wedge}(f_{\rightarrow}(v(p), v(q)), f_{\rightarrow}(v(q), v(r))), f_{\rightarrow}(v(p), v(r))), \\ &= f_{\rightarrow}(f_{\wedge}(f_{\rightarrow}(0, 1), f_{\rightarrow}(1, 0)), f_{\rightarrow}(0, 0)), \\ &= f_{\rightarrow}(f_{\wedge}(1, 0), 1), \\ &= f_{\rightarrow}(0, 1), \\ &= 1. \qquad \blacksquare \end{aligned}$$

Uit deze berekeningen —die voor andere formules uit *PROP* volkomen analoog verlopen— blijkt dat de waarheidswaarde van een formule $F \in PROP$ slechts afhankelijk is van de waarheidswaarden van de in F voorkomende propositiesymbolen.

3.1.5 STELLING **Valuatiestelling**
Zij $F \in PROP$. Als v en w valuaties zijn zodanig dat $v(p) = w(p)$ voor ieder propositiesymbool p dat in F voorkomt, dan geldt dat $v(F) = w(F)$.

BEWIJS Door middel van eenvoudige structurele inductie over *PROP*. ■

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Tabel 3.2: De waarheidstafel van $\oplus$.

Uit de waarheidstafels in tabel 3.1 kan men afleiden dat $A \wedge B$ en $B \wedge A$ steeds dezelfde waarheidswaarde hebben. Dus $v(A \wedge B) = v(B \wedge A)$ voor alle $A, B \in PROP$. Zo geldt ook $v(A \vee B) = v(B \vee A)$.

Verder volgt uit de waarheidstafel voor de disjunctie $\vee$, dat dit voegwoord de betekenis heeft van een *inclusief of.* Immers, $v(A \vee B) = 1$ dan en slechts dan als $v(A) = 1$ en $v(B) = 0$, of als $v(A) = 0$ en $v(B) = 1$, òf als $v(A) = v(B) = 1$.

De *exclusieve of,* ook wel de *xor* genaamd en genoteerd als $\oplus$, kan gedefinieerd worden zodat $v(A \oplus B) = 1$ dan en slechts dan als $v(A) = 1$ en $v(B) = 0$, òf als $v(A) = 0$ en $v(B) = 1$. In de overige gevallen geldt $v(A \oplus B) = 0$. Zie tabel 3.2.

Bij de formele interpretatie van de implicatie $\rightarrow$ is het gebruik ervan in de wiskunde gevolgd. Dat dit niet altijd in overeenstemming is met het gebruik van 'als . . . dan' in de omgangstaal, bleek al in paragraaf 2.1. Beschouw bijvoorbeeld de reeds eerder genoemde bewering

(3.3) Als Amsterdam de hoofdstad van Nederland is, dan is zeven een priemgetal.

In de objecttaal $\mathcal{P}$ kan men deze bewering aanduiden door $p \rightarrow q$, waarin p en q respectievelijk de beweringen 'Amsterdam is de hoofdstad van Nederland' en 'Zeven is een priemgetal' aanduiden. Ten aanzien van de situatie in de wereld geldt nu dat $v(p) = v(q) = 1$. Dit betekent dat $v(p \rightarrow q) = 1$. Formeel kwalificeren we bewering (3.3) dus als waar, terwijl we deze in de omgangstaal echter als onzinnig zouden bestempelen.

Tot slot laten we aan de hand van enkele eenvoudige wiskundige uitspraken zien, dat de materiële implicatie het gebruik van het voegwoord 'als . . . dan' in de wiskunde op een juiste manier weergeeft. Beschouw de volgende wiskundige uitspraak over natuurlijke getallen.

(3.4) Als n deelbaar is door 4, dan is n deelbaar door 2.

Merk allereerst op dat deze uitspraak waar is voor alle natuurlijke getallen n. Laat p en q respectievelijk de volgende beweringen aanduiden: 'n is deelbaar door 4' en 'n is deelbaar door 2'.

Voor $n = 4$ geldt dat zowel p als q waar zijn. Nemen we $n = 6$, dan is p onwaar, maar q waar. Indien we bijvoorbeeld $n = 13$ nemen, dan zijn zowel p als q onwaar. In al deze gevallen is uitspraak (3.4) waar, zoals we reeds geconstateerd hebben. Dit is in overeenstemming met de waarheidstafel voor de implicatie: $v(A \rightarrow B) = 1$ indien $v(A) = 0$ of $v(B) = 1$.

De volgende uitspraak over natuurlijke getallen is niet waar:

(3.5) Als n deelbaar is door 2, dan is n deelbaar door 4.

Immers, 6 is deelbaar door 2, maar niet door 4. Ook dit is in overeenstemming met de waarheidstafel: $v(A \rightarrow B) = 0$ indien $v(A) = 1$ en $v(B) = 0$.

Een bekende wiskundige redeneervorm is de *redenering uit het ongerijmde*. Deze redeneervorm is gebaseerd op het feit dat uit de waarheidstafel voor de implicatie volgt dat $v(A) = 0$ indien $v(A \rightarrow B) = 1$ en $v(B) = 0$.

Stel men wil de bewering $\neg p$ bewijzen, en men weet dat $p \rightarrow q$ en $\neg q$ waar zijn. Het bewijs van $\neg p$ uit het ongerijmde verloopt dan aldus. Stel dat p *waar* is. In combinatie met het gegeven dat $p \rightarrow q$ waar is, volgt hieruit dat q waar is. Het andere gegeven stelt echter dat $\neg q$ waar is, in tegenspraak met het zojuist geconstateerde. De aanname dat p waar is, moet dus onjuist zijn, zodat $\neg p$ het geval moet zijn.

De waarheidstafel voor de implicatie voorspelt inderdaad dat deze redenering correct is. Er is gegeven dat $v(p \rightarrow q) = v(\neg q) = 1$. Hieruit volgt dat $v(q) = 0$. Uit de waarheidstafel volgt dat $v(p) = 0$ indien $v(p \rightarrow q) = 1$ en $v(q) = 0$. Maar dit betekent dat $v(\neg p) = 1$. Schematisch is dit de volgende redeneervorm:

(3.6) $p \rightarrow q$

$\neg q$

$\therefore\ \neg p$

3.2 Model en logisch gevolg

In deze paragraaf zullen we een aantal belangrijke begrippen definiëren. Daarbij staat de notie valuatie die in de vorige paragraaf is geïntroduceerd, centraal.

3.2.1 Definitie **Model, tautologie en logisch gevolg**

1. *Een* model *van een formule $F \in PROP$ is een valuatie v zodanig dat $v(F) = 1$. Men noemt F* vervulbaar *als F een model heeft. Als v een model is voor F, zegt men ook dat F* vervuld *wordt door v.*

2. *Een* model *van een verzameling* $\Gamma \subseteq PROP$ *is een valuatie* v *zodanig dat* $v(F) = 1$ *voor alle* $F \in \Gamma$. *Per definitie is iedere valuatie een model van de lege verzameling. Men noemt* Γ vervulbaar *als* Γ *een model heeft. Als* v *een model is voor* Γ, *dan zegt men ook dat* Γ vervuld *wordt door* v.

3. *Een formule* F *wordt een* tautologie *genoemd, notatie* $\models F$, *als iedere valuatie* v *een model van* F *is. Men noemt* F *ook* algemeen geldig.

4. *Een formule* F *is een* logisch gevolg *van een verzameling* $\Gamma \subseteq PROP$, *notatie* $\Gamma \models F$, *als ieder model van* Γ *tevens een model van* F *is.*

Vaak noteert men $\{A_1, \ldots, A_n\} \models F$ als $A_1, \ldots, A_n \models F$. Hierbij is de volgorde van de formules $A_1, \ldots, A_n$ irrelevant, zoals ook uit de definitie van de notie logisch gevolg blijkt. $\emptyset \models F$ is equivalent met $\models F$.

De definitie van het begrip logisch gevolg stemt overeen met de in §1.2 ontwikkelde intuïtie betreffende de logische geldigheid van een redenering. Hebben we een redenering van de vorm $\Gamma, \therefore\; F$, waarin Γ de verzameling premissen voorstelt en F de conclusie, dan is deze redenering logisch geldig, dan en slechts dan als F een logisch gevolg is van Γ. Dus als ieder model van de premissen tevens een model van de conclusie is, ofwel als de waarheid van de premissen met betrekking tot een willekeurige valuatie de waarheid van de conclusie forceert.

Ingeval de redenering niet logisch geldig is, dan moet er een valuatie bestaan die een model is voor de premissen, maar die de conclusie onwaar maakt. Zo'n valuatie wordt een *tegenvoorbeeld* genoemd.

3.2.2 DEFINITIE **Logisch geldige redenering en tegenvoorbeeld**
Zij $\Gamma \subset PROP$ *en* $F \in PROP$.

1. *De redenering* $\Gamma, \therefore F$ *is* logisch geldig, *dan en slechts dan als* $\Gamma \models F$.

2. *Een* tegenvoorbeeld *voor de redenering* $\Gamma, \therefore F$ *is een valuatie* v *zodanig dat* v *een model is voor* Γ, *maar niet voor* F.

3.2.3 VOORBEELD Logisch gevolg en vervulbaarheid.

1. $A, \neg A \models B$.
   Bewijs: Er is geen enkele valuatie v die model is van zowel A als $\neg A$, dus alle valuaties die model zijn van zowel A als $\neg A$, zijn een model van B.

2. $\Gamma = \{p \rightarrow q, \neg p, q\}$ is een vervulbare verzameling.
   Bewijs: Een valuatie v waarvoor $v(p) = 0$ en $v(q) = 1$, vervult Γ. ■

p	q	r	$p\to q$	$q\to r$	$(p\to q)\to(q\to r)$	$p\wedge r$	F
0	0	0	1	1	1	0	0
0	0	1	1	1	1	0	0
0	1	0	1	0	0	0	1
0	1	1	1	1	1	0	0
1	0	0	0	1	1	0	0
1	0	1	0	1	1	1	1
1	1	0	1	0	0	0	1
1	1	1	1	1	1	1	1

Tabel 3.3: Waarheidstafel voor $F = [(p\to q)\to(q\to r)]\to(p\wedge r)$.

Met behulp van waarheidstabellen is eenvoudig na te gaan of een formule een tautologie is. Zij $F \in \mathit{PROP}$ een formule die n propositiesymbolen $q_1,\ldots,q_n$ bevat. We hebben gezien dat de waarheidswaarde van F slechts afhankelijk is van de waarheidswaarden van $q_1,\ldots,q_n$ (zie voorbeeld 3.1.5). Totaal zijn er 2^n verschillende rijtjes $v(q_1),\ldots,v(q_n)$ bestaande uit nullen en enen. Hieruit volgt dat F een tautologie is als $v(F) = 1$ voor al deze rijtjes $v(q_1),\ldots,v(q_n)$. We kunnen al deze rijtjes $v(q_1),\ldots,v(q_n)$ op systematische wijze weergeven in een waarheidstafel voor F. We zullen dit aan een voorbeeld illustreren.

3.2.4 VOORBEELD Waarheidstafel van een formule.
Tabel 3.3 is een waarheidstafel voor de formule $F = [(p\to q)\to(q\to r)]\to(p\wedge r)$. Omdat F alleen de propositiesymbolen p, q en r bevat, hebben we in de waarheidstafel te maken met $2^3 = 8$ rijen. De tabel bevat naast de drie kolommen voor de waarheidswaarden van p, q en r en een kolom voor de waarheidswaarden van F, vier hulpkolommen voor de waarheidswaarden van de subformules $p\to q$, $q\to r$, $(p\to q)\to(q\to r)$ en $p\wedge r$. Uit deze waarheidstafel van F, volgt dat F geen tautologie is. Immers, de kolom van F bestaat niet alleen uit enen. Er zijn dus valuaties mogelijk waarvoor $v(F) = 0$. Zo'n valuatie hebben we een *tegenvoorbeeld* genoemd. Een tegenvoorbeeld is af te lezen uit een rij waarvoor $v(F) = 0$. Zo is een valuatie waarvoor $v(p) = v(q) = v(r) = 0$ (zoals in rij 1), een tegenvoorbeeld voor F. ■

Op dezelfde wijze waarop we waarheidstafels kunnen gebruiken om na te gaan of een formule een tautologie is, kunnen we nagaan of een formule het logisch gevolg is van een (eindige) verzameling formules. Zij $F \in \mathit{PROP}$ en $\Gamma \subseteq \mathit{PROP}$ en laat $q_1,\ldots,q_n$ de enige propositiesymbolen zijn die in F of in de formules in Γ voorkomen. In dit geval zijn er weer 2^n rijtjes $v(q_1),\ldots,v(q_n)$ van

p	q	r	$p{\rightarrow}q$	$q{\rightarrow}r$	$p{\rightarrow}r$
0	0	0	1	1	1
0	0	1	1	1	1
0	1	0	1	0	1
0	1	1	1	1	1
1	0	0	0	1	0
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	1	1

Tabel 3.4: Waarheidstafel voor de formules $p{\rightarrow}q$, $q{\rightarrow}r$ en $p{\rightarrow}r$.

p	q	$p{\rightarrow}q$	$\neg p$	$\neg q$
0	0	1	1	1
0	1	1	1	0
1	0	0	0	1
1	1	1	0	0

Tabel 3.5: Waarheidstafel voor de formules $p{\rightarrow}q$, $\neg p$ en $\neg q$.

nullen en enen. Hieruit volgt dat $\Gamma \models F$, indien voor alle rijtjes $v(q_1), \ldots, v(q_n)$ geldt dat als $v(A) = 1$ voor alle $A \in \Gamma$, dan ook $v(F) = 1$. In termen van waarheidstafels: $\Gamma \models F$ als voor alle rijen waarvoor geldt dat de formules uit Γ een '1' hebben, ook F een '1' heeft.

3.2.5 VOORBEELD Waarheidstafels en logisch gevolg.
De waarheidstafel in tabel 3.4 laat zien dat $p{\rightarrow}q, q{\rightarrow}r \models p{\rightarrow}r$. Duidelijk is dat in de rijen 1, 2, 4 en 8 waarvoor de beide formules $p{\rightarrow}q$ en $q{\rightarrow}r$ een '1' hebben, ook de formule $p{\rightarrow}r$ een '1' bezit. ■

3.2.6 VOORBEELD Tegenvoorbeeld.
Als we met behulp van een waarheidstafel proberen om na te gaan of

(3.7) $p{\rightarrow}q$
$\neg p$
$\therefore\ \neg q$

een logische geldig redenering is, krijgen we de waarheidstafel als in tabel 3.5. Uit de tafel blijkt dat de redenering niet logisch geldig is. Immers de tweede

A	B	$\neg B$	$A \rightarrow B$	$A \vee \neg B$	$\neg A$
0	0	1	1	1	1
0	1	0	1	0	1
1	0	1	0	1	0
1	1	0	1	1	0

Tabel 3.6: Waarheidstafel voor de formules $A \rightarrow B$, $A \vee \neg B$ en $\neg A$.

rij geeft een situatie waarvoor de premissen waar zijn, maar de conclusie niet. Zoals we uit de genoemde rij kunnen aflezen, wordt zo'n situatie gerealiseerd door een valuatie die voldoet aan $v(p) = 0$ en $v(q) = 1$. Een dergelijke valuatie is dus een *tegenvoorbeeld* voor de redenering. ■

De methode van de waarheidstafels is ook te gebruiken om na te gaan of formules die *metavariabelen* bevatten, tautologieën zijn. Zo kan men ook redeneringen waarin de formules metavariabelen bevatten, op hun logische geldigheid testen. Dit laatste wordt geïllustreerd in het volgende voorbeeld.

3.2.7 VOORBEELD Tegenvoorbeeld voor een redenering met metavariabelen. Beschouw de volgende redenering:

(3.8) $A \rightarrow B$

$A \vee \neg B$

$\therefore \neg A$

Tabel 3.6 laat zien dat deze redenering niet logisch geldig is voor willekeurige $A, B \in PROP$. Immers, in rij 4 zijn de twee premissen waar, maar is de conclusie onwaar. Een valuatie v en twee proposities A en B zodanig dat $v(A) = v(B) = 1$ vormen een tegenvoorbeeld voor de redenering. Dergelijke v, A en B zijn eenvoudig te vinden: neem willekeurige propositiesymbolen p en q, kies een valuatie met $v(p) = v(q) = 1$, en stel $A = p$ en $B = q$.

Aangezien de tabel ook rijen bevat waarin het niet het geval is dat alle premissen waar zijn en de conclusie onwaar is —in deze tabel de rijen 1, 2 en 3—, zijn er substituties van A en B te geven zodanig dat de redenering logisch geldig is. Als men bijvoorbeeld neemt $A = p \wedge \neg p$ en $B = p \vee \neg p$ voor een willekeurig propositiesymbool p, dan geldt voor alle valuaties v dat $v(A) = 0$ en $v(B) = 1$. Dit betekent dat van tabel 3.6 alleen rij 2 overblijft. Voor *deze* keuze van A en B is de redenering logisch geldig.

We zien dat er keuzen voor A en B zijn, die de redenering logisch geldig maken, en die dat niet doen. Als iedere keuze voor A en B de redenering logisch geldig maakt, dan is er sprake van een geldig redeneerschema. ■

3.3 Redeneringen

In deze paragraaf laten we aan de hand van een aantal voorbeelden zien, hoe de correctheid van een redenering in de Nederlandse taal gecontroleerd kan worden met behulp van de tot nu toe ontwikkelde logische concepten.

Daartoe worden eerst de premissen en de conclusie van de redenering zo goed mogelijk omgezet in een reeks formules $P_1, \ldots, P_n, C \in \mathit{PROP}$. Hoe dit kan worden gedaan, is uiteengezet in hoofdstuk 2, voorbeeld 2.2.7. De redenering in de Nederlandse taal is correct dan en slechts dan als $P_1, \ldots, P_n \models C$.

3.3.1 VOORBEELD Contrôle van Nederlandstalige redeneringen.

1. De redenering luidt:

   (a) Karel is ijverig $\boxed{\text{of}}$ intelligent.

   (b) $\boxed{\text{Als}}$ Karel slaagt, $\boxed{\text{dan}}$ is hij intelligent.

   (c) Karel slaagt $\boxed{\text{niet}}$ $\boxed{\text{en}}$ hij is toch ijverig.

   (d) Dus Karel is $\boxed{\text{niet}}$ intelligent.

   Na vertaling in de propositielogica verkrijgt men de volgende premissen en conclusie:

   (a) $P_1 = k \vee i$

   (b) $P_2 = s \rightarrow i$

   (c) $P_3 = \neg s \wedge k$

   (d) $C \ = \neg i$

   Met een waarheidstafel is eenvoudig na te gaan dat de redenering niet logisch geldig is. Een valuatie met $v(k) = v(i) = 1$ en $v(s) = 0$ is een tegenvoorbeeld. Deze is immers een model voor de verzameling premissen $\{P_1, P_2, P_3\}$, maar niet voor de conclusie C.

2. Beschouw de onderstaande redenering:

   (a) $\boxed{\text{Als}}$ Jan komt, $\boxed{\text{dan}}$ regent het $\boxed{\text{niet}}$.

   (b) Het regent.

   (c) Dus Jan komt $\boxed{\text{niet}}$.

   Na vertaling luiden de premissen en de conclusie:

   (a) $P_1 = k \rightarrow \neg r$

(b) $P_2 = r$

(c) $C \;= \neg k$

Met behulp van een waarheidstafel is eenvoudig na te gaan dat de redenering logisch geldig is. Het is ook direct in te zien. Immers uit $v(k \rightarrow \neg r) = 1$ en $v(r) = 1$ volgt $v(\neg k) = 1$.

3. De volgende redenering lijkt veel op de vorige, maar is onjuist.

   (a) $\boxed{\text{Als}}$ het $\boxed{\text{niet}}$ regent, $\boxed{\text{dan}}$ komt Jan.

   (b) Het regent.

   (c) Dus Jan komt $\boxed{\text{niet}}$.

   De premissen en de conclusie luiden nu:

   (a) $P_1 = \neg r \rightarrow k$

   (b) $P_2 = r$

   (c) $C \;= \neg k$

   Het is eenvoudig te verifiëren dat een valuatie die voldoet aan $v(r) = v(k) = 1$ een tegenvoorbeeld voor de redenering is. ■

3.4 Opgaven

1. Zij $A, B, C \in PROP$ en zij gegeven dat $v(A) = v(C) = 1$ en $v(B) = 0$. Bereken dan de waarheidswaarden van de volgende proposities met betrekking tot v.

   (a) $\neg A \rightarrow (A \rightarrow B)$.

   (b) $\neg A \rightarrow (B \vee C)$.

   (c) $(A \wedge C) \rightarrow B$.

   (d) $(A \rightarrow B) \vee (B \rightarrow A)$.

   (e) $((A \wedge B) \vee (A \rightarrow (A \rightarrow (C \rightarrow A)))) \rightarrow C$.

   (f) $B \rightarrow (A \rightarrow (A \rightarrow (\neg B \rightarrow (C \rightarrow B))))$.

2. Zij $A, B \in PROP$. Maak een waarheidstafel voor elk van de volgende proposities. Ga vervolgens na of de betreffende propositie een *tautologie* is. Geef een tegenvoorbeeld (inclusief substituties voor A en B) als dit niet zo is.

   (a) $\neg A \rightarrow (A \rightarrow B)$.

   (b) $B \to (A \to B)$.
   (c) $(\neg A \wedge \neg B) \leftrightarrow \neg(A \vee B)$.
   (d) $\neg(A \to B) \leftrightarrow (\neg A \to \neg B)$.

3. Ga met behulp van een waarheidstafel na of de volgende redenering *logisch geldig* is voor iedere keuze van $A, B, C \in PROP$:

$$A \to B, \neg C \to \neg B, \neg A \to A, \ \therefore \ C.$$

4. Vertaal de volgende redenering in de *propositielogica* en ga vervolgens na of deze *logisch geldig* is.

   Als Thea slaapt en Arie gaat gillen, dan wordt Peter wakker.
   Ofwel Thea slaapt niet, ofwel Arie gaat gillen.
   Derhalve wordt Peter wakker als Thea slaapt.

5. Bewijs de valuatiestelling (3.1.5).

Hoofdstuk 4

Stellingen over de Propositielogica

In dit hoofdstuk wordt een aantal uiteenlopende eigenschappen van de propositielogica behandeld. In §4.1 wordt het begrip *meta-stelling* geïntroduceerd en worden een aantal meta-stellingen bewezen. Vervolgens wordt in §4.2 een algebraïsche kijk op de propositielogica gegeven. Tenslotte wordt in §4.3 de notie *normaalvorm* gepresenteerd.

4.1 Metataal en meta-stellingen

In paragraaf 2.2 hebben we het onderscheid tussen objecttaal en metataal geïntroduceerd. Een formule als $((p \wedge q) \rightarrow p)$ behoort zoals we hebben gezien tot de objecttaal $\mathcal{P}$. Maar tot welke taal behoort de volgende uitdrukking?

$$\models ((p \wedge q) \rightarrow p). \tag{4.1}$$

Deze uitdrukking kan niet in $\mathcal{P}$ thuishoren aangezien het symbool $\models$ niet tot het alfabet van $\mathcal{P}$ behoort. Het antwoord op de vraag kan dus alleen nog maar luiden dat uitdrukking (4.1) tot de metataal behoort. Dit vooronderstelt echter dat $\models$ een symbool is van de metataal en dat bovendien alle formules van de objecttaal tevens tot de metataal behoren. In feite is onze metataal dus een (forse) uitbreiding van de objecttaal.

Wat (4.1) uitdrukt, is dat de objecttalige formule $((p \wedge q) \rightarrow p)$ een tautologie is. In de metataal kunnen we dus eigenschappen van formules van de objecttaal vastleggen. Uitdrukkingen uit de metataal die niet tot de objecttaal behoren, noemt men *metabeweringen*. Als een metabewering, zoals (4.1), bovendien waar is, noemt men deze een *meta-stelling*.

Een ander aspect van de metataal dat we tegenkwamen, is het feit dat er in de metataal variabelen, de zogenaamde *metavariabelen*, worden gebruikt. We hebben niet alleen de letters $A, B, \ldots$ gebruikt als metavariabelen voor formules uit $PROP$, maar we hebben ook Γ gebruikt als metavariabele voor deelverzamelingen van $PROP$. We hebben zelfs $\star$ gebruikt als metavariabele

metatalig connectief	afkorting van
$\sim$	niet
&	en
$\bigvee$	of
$\Rightarrow$	als ... dan
$\Leftrightarrow$	dan en slechts dan als

Tabel 4.1: De connectieven van de metataal.

voor de tweeplaatsige connectieven van $\mathcal{P}$. Kortom, we kunnen in de metataal verschillende typen metavariabelen gebruiken.

We willen de metataal vanwege het praktische gemak nog verder uitbreiden. Beschouw bijvoorbeeld de meta-stelling:

$$\text{Als} \quad \models A \quad \text{en} \quad \models B, \quad \text{dan} \quad \models A \wedge B. \tag{4.2}$$

Het is vervelend om vaak *Als ... dan* en *en* te moeten schrijven. Daarom willen we voor deze metatalige voegwoorden metatalige connectieven introduceren. Deze zijn opgenomen in tabel 4.1. Merk op dat er voor ieder connectief van de objecttaal een ermee corresponderend connectief uit de metataal bestaat. Om geen verwarring te zaaien zijn hiervoor andere symbolen gebruikt. Men dient zich wel te realiseren dat de metatalige connectieven *afkortingen* zijn van de voegwoorden zoals deze in de *wiskunde* worden gebruikt! Voor deze metatalige connectieven zullen we dezelfde precedentieregels gebruiken als voor de objecttalige.

Met gebruik van de metatalige connectieven kan meta-stelling (4.2) nu als volgt worden weergegeven:

$$\models A \quad \& \quad \models B \quad \Rightarrow \quad \models A \wedge B.$$

Hoe deze en andere meta-stellingen kunnen worden bewezen, laten we zien in het volgende voorbeeld. Let er in deze bewijzen op dat er vaak een beroep moet worden gedaan op de begrippen uit definitie 3.2.1.

4.1.1 VOORBEELD Meta-stellingen.

1. $\models A \quad \& \quad \models B \quad \Rightarrow \quad \models A \wedge B$.
   Bewijs: Uit $\models A$ en $\models B$ volgt dat voor alle valuaties v geldt $v(A) = v(B) = 1$. Hieruit volgt dat voor alle v geldt $v(A \wedge B) = 1$, ofwel $\models A \wedge B$.

2. $A \models B \quad \Rightarrow \quad \models A \rightarrow B$ (*deductiestelling*).
   Bewijs: Uit $A \models B$ volgt dat voor iedere valuatie v geldt: als $v(A) = 1$

dan $v(B) = 1$. Dit betekent dat er geen valuatie v bestaat zodanig dat $v(A) = 1$ en $v(B) = 0$. Derhalve moet voor alle valuaties v gelden dat $v(A \to B) = 1$, zodat $\models A \to B$. ■

Dat niet alle metabeweringen ook meta-stellingen zijn volgt uit:

4.1.2 VOORBEELD Tegenvoorbeeld.
De metabewering:

$$\models A \vee B \quad \Rightarrow \quad \models A \quad \bigvee \quad \models B$$

is geen meta-stelling. We kunnen namelijk bepaalde $A, B \in PROP$ kiezen zodanig dat wel geldt $\models A \vee B$, maar niet $\models A$ en ook niet $\models B$. Stel namelijk $A = p$ en $B = \neg p$ voor een atoom p. Dan geldt zeker $\models p \vee \neg p$, maar er geldt niet $\models p$ en ook niet $\models \neg p$. Immers, er bestaan valuaties v waarvoor $v(p) = 1$ (zodat $v(\neg p) = 0$), maar ook valuaties v waarvoor $v(p) = 0$ (zodat $v(\neg p) = 1$). De keuze $A = p$ en $B = \neg p$ wordt een *tegenvoorbeeld* voor de metabewering genoemd. ■

In de volgende stelling zijn een aantal 'nuttige' tautologieën bij elkaar gezet.

4.1.3 STELLING **Tautologieën**
Zij $A, B \in PROP$. De volgende formules zijn tautologieën:

1. $A \vee \neg A$ (tertium non datur *of* wet van de uitgesloten derde).
2. $(A \to B) \leftrightarrow (\neg B \to \neg A)$ (contrapositie).
3. $\neg(A \wedge B) \leftrightarrow (\neg A \vee \neg B)$ (wet van De Morgan).
4. $\neg(A \vee B) \leftrightarrow (\neg A \wedge \neg B)$ (wet van De Morgan).
5. $(A \to B) \leftrightarrow (\neg A \vee B)$.
6. $\neg(A \to B) \leftrightarrow (A \wedge \neg B)$.
7. $(A \leftrightarrow B) \leftrightarrow [(A \to B) \wedge (B \to A)]$.
8. $\neg(A \leftrightarrow B) \leftrightarrow [\neg(A \to B) \vee \neg(B \to A)]$.

BEWIJS Al deze tautologieën kunnen met behulp van waarheidstafels worden bewezen. Men kan ze echter ook bewijzen door een beroep te doen op de definitie van het begrip tautologie in termen van valuaties. Bij wijze van voorbeeld zullen we tautologie 6 op beide manieren bewijzen.

A	B	$\neg B$	$(A\to B)$	$\neg(A\to B)$	$A\wedge\neg B$	F
0	0	1	1	0	0	1
0	1	0	1	0	0	1
1	0	1	0	1	1	1
1	1	0	1	0	0	1

Tabel 4.2: $F = \neg(A\to B)\leftrightarrow(A\wedge\neg B)$ is een tautologie.

De waarheidstafel van tautologie 6 vindt men in tabel 4.2. Uit deze tafel blijkt dat de kolom van de betreffende formule alleen enen bevat. Dit betekent dat zij een tautologie is. Het tweede bewijs bezit het karakter van een berekening en ziet er als volgt uit:

$$\begin{array}{lcl} v(\neg(A\to B)) = 1 & \Leftrightarrow & v(A\to B) = 0, \\ & \Leftrightarrow & v(A) = 1 \quad \& \quad v(B) = 0, \\ & \Leftrightarrow & v(A) = 1 \quad \& \quad v(\neg B) = 1, \\ & \Leftrightarrow & v(A\wedge\neg B) = 1. \end{array}$$

Aangezien dit voor alle valuaties v opgaat, is $\neg(A\to B)\leftrightarrow(A\wedge\neg B)$ een tautologie. ■

In de volgende definitie introduceren we de relatie $\approx$ in de metataal. Deze relatie geldt tussen twee formules van $\mathcal{P}$, indien deze uitwisselbaar zijn. Dergelijke formules worden equivalent genoemd.

4.1.4 DEFINITIE **Equivalentie van formules**

Twee formules $A, B \in PROP$ noemt men equivalent, *notatie $A \approx B$, als geldt $\models A\leftrightarrow B$.*

Dus de equivalentie van twee formules A en B kan worden aangetoond door te bewijzen dat $A\leftrightarrow B$ een tautologie is. De relatie $\approx$ blijkt een speciaal soort relatie te zijn: een *equivalentierelatie.*

4.1.5 DEFINITIE **Equivalentierelatie**

Een relatie $R \subseteq D \times D$ is een equivalentierelatie *dan en slechts dan als:*

1. R reflexief *is, ofwel voor alle $a \in D$ geldt aRa;*
2. R symmetrisch *is, ofwel voor alle $a, b \in D$ geldt: als aRb, dan bRa; en*
3. R transitief *is, ofwel voor alle $a, b, c \in D$ geldt: als aRb en bRc, dan aRc.*

4.1.6 VOORBEELD Equivalentierelatie.
Definieer de relatie $R \subseteq \mathbb{N} \times \mathbb{N}$ als volgt: xRy geldt dan en slechts dan als x en y na deling door 7 dezelfde rest hebben. Er geldt dan bijvoorbeeld $7R35$, $3R24$, en $18R25$. Ga zelf na dat deze relatie R een equivalentierelatie is.

4.1.7 STELLING *De relatie $\approx$ is een equivalentierelatie.*

BEWIJS Dit volgt onmiddellijk uit:

1. $\approx$ is reflexief, want voor elke $A \in PROP$ geldt $\models A \leftrightarrow A$, zodat $A \approx A$.
2. $\approx$ is symmetrisch, want voor alle $A, B \in PROP$ geldt

$$\begin{array}{rcll} A \approx B & \Leftrightarrow & \models A \leftrightarrow B & \Leftrightarrow \\ \models B \leftrightarrow A & \Leftrightarrow & B \approx A. & \end{array}$$

3. $\approx$ is transitief, want voor alle $A, B, C \in PROP$ geldt

$$\models A \leftrightarrow B \quad \& \quad \models B \leftrightarrow C \quad \Rightarrow \quad \models A \leftrightarrow C.$$

Waarmede de stelling is bewezen. ∎

4.1.8 VOORBEELD Equivalente formules.
De resultaten van stelling 4.1.3 kunnen ook in termen van de relatie $\approx$ worden geformuleerd. Dit levert bijvoorbeeld voor nummer 3 en 5:

1. $(A \wedge B) \approx \neg(\neg A \vee \neg B)$,
2. $(A \rightarrow B) \approx (\neg A \vee B)$. ∎

Voor het leveren van een bewijs is het soms nodig subformules van een formule te vervangen door andere formules. Dit vervangen noemt men naar analogie met de wiskunde *substitueren*. In feite hebben we tot nog toe stilzwijgend verondersteld hoe formules te substitueren voor metavariabelen. We zullen nu een geschikte notatie invoeren.

De substitutie van de formule C voor alle voorkomens van het propositiesymbool p in de formule A wordt genoteerd als $A[p{:=}C]$. Dat is dus de formule die men verkrijgt door alle voorkomens van p in A te vervangen door C. De substitutie-operatie voegt aan de formule A de formule $A[p{:=}C]$ toe en kan dus beschouwd worden als een functie $S^p_C : PROP \rightarrow PROP$ met als functiewaarde $S^p_C(A) = A[p{:=}C]$. We kunnen $[p{:=}C]$ ook opgevatten als een *postfixoperator*, een operator die ná zijn operand wordt geschreven.

4.1.9 DEFINITIE **Substitutie**
Zij $A, B, C \in PROP$*, en* p *en* q *propositiesymbolen. De* substitutie-*operatie* $[p{:=}C]$ *is als volgt recursief gedefinieerd:*

$$\begin{aligned} q[p{:=}C] &= \begin{cases} C & \text{als } q = p, \\ q & \text{als } q \neq p, \end{cases} \\ (\neg A)[p{:=}C] &= \neg\,(A[p{:=}C]), \\ (A \star B)[p{:=}C] &= (A[p{:=}C]) \,\star\, (B[p{:=}C]). \end{aligned}$$

Merk op dat we in de definitie haakjes hebben moeten gebruiken om aan te geven wat het argument is voor de operator $[p{:=}C]$. Merk ook op dat we eigenlijk moeten aantonen dat $F[p{:=}C] \in PROP$ als gegeven is dat $F, C \in PROP$ en p een propositiesymbool is. We laten dit echter aan de lezer over.

4.1.10 STELLING **Substitutiestelling**
Voor alle formules $F, C, D \in PROP$ *en propositiesymbolen* p *geldt:*

$$C \approx D \quad \Rightarrow \quad F[p{:=}C] \approx F[p{:=}D].$$

BEWIJS Het bewijs wordt geleverd met behulp van structurele inductie over $PROP$ met betrekking tot de formule F. Hiertoe onderscheiden we de volgende gevallen:

1. $F = q$ waarbij q een propositiesymbool is.
   (a) Als $q = p$, dan $F[p{:=}C] = C$ en $F[p{:=}D] = D$. Aangezien gegeven is dat $C \approx D$, volgt dat $F[p{:=}C] \approx F[p{:=}D]$.
   (b) Als $q \neq p$, dan $F[p{:=}C] = F[p{:=}D] = q$, zodat geldt $F[p{:=}C] \approx F[p{:=}D]$.
2. $F = \neg A$ voor zekere $A \in PROP$. De inductieveronderstelling luidt dat $A[p{:=}C] \approx A[p{:=}D]$. Dit betekent dat voor alle valuaties v geldt: $v(A[p{:=}C]) = v(A[p{:=}D])$. Hieruit volgt:

$$\begin{aligned} v(F[p{:=}C]) &= v(\neg(A[p{:=}C])), \\ &= f_\neg(v(A[p{:=}C])), \\ &= f_\neg(v(A[p{:=}D])), \\ &= v(\neg(A[p{:=}D])), \\ &= v(F[p{:=}D]), \end{aligned}$$

   zodat $F[p{:=}C] \approx F[p{:=}D]$.

3. $F = (A \star B)$ voor zekere $A, B \in PROP$. De inductieveronderstelling luidt dat $A[p{:=}C] \approx A[p{:=}D]$ en $B[p{:=}C] \approx B[p{:=}D]$. Dit betekent dat voor alle valuaties v het volgende geldt: $v(A[p{:=}C]) = v(A[p{:=}D])$ en $v(B[p{:=}C]) = v(B[p{:=}D])$. Hieruit volgt:

$$\begin{array}{rcl} v(F[p{:=}C]) & = & v((A[p{:=}C]) \star (B[p{:=}C])), \\ & = & f_\star(v(A[p{:=}C]), v(B[p{:=}C])), \\ & = & f_\star(v(A[p{:=}D]), v(B[p{:=}D])), \\ & = & v((A[p{:=}D]) \star (B[p{:=}D])), \\ & = & v(F[p{:=}D]), \end{array}$$

zodat $F[p{:=}C] \approx F[p{:=}D]$.

Hiermede is de stelling bewezen. ■

4.1.11 VOORBEELD Substitutie.
Zij $F = p{\rightarrow}q$, $C = q{\rightarrow}p$ en $D = \neg q \vee p$, zodat volgens stelling 4.1.3(5) geldt dat $C \approx D$. Uit de substitutiestelling volgt dat $F[p{:=}C] = (q{\rightarrow}p){\rightarrow}q \approx (\neg q \vee p){\rightarrow}q = F[p{:=}D]$. ■

4.1.12 VOORBEELD Toepassing substitutie.
Zij $C = p{\rightarrow}q$ en $D = \neg p \vee q$, zodat volgens stelling 4.1.3(5) geldt dat $C \approx D$. Zij vervolgens $F = [(p{\rightarrow}q) \wedge p]{\rightarrow}[(p{\rightarrow}q) \wedge (p{\rightarrow}q)]$, en laat G verkregen zijn door in F het eerste en het derde voorkomen van C door D te vervangen. G is dus de formule $[(\neg p \vee q) \wedge p]{\rightarrow}[(p{\rightarrow}q) \wedge (\neg p \vee q)]$.

Door de substitutiestelling op een slimme wijze toe te passen, kan men laten zien dat $F \approx G$. Zij namelijk $M = [(r \wedge p){\rightarrow}[(p{\rightarrow}q) \wedge r]$, en zij r een propositiesymbool dat verschillend is van p en q, dan volgt door toepassing van definitie 4.1.9 dat $F = M[r{:=}C]$ en $G = M[r{:=}D]$. De substitutiestelling is nu direct toepasbaar. ■

Uit het bovenstaande voorbeeld blijkt hoe de substitutiestelling in de praktijk wordt gebruikt. Als men in een formule F een aantal voorkomens van een subformule C vervangt door een formule D waarvoor $D \approx C$, dan geldt voor de resulterende formule G dat $G \approx F$. Dit gebruik van de substitutiestelling wordt, naast het toepassen van een substitutie-operatie $[p{:=}C]$, ook *substitutie* genoemd.

4.2 Algebraïsche eigenschappen

Uit de wiskunde weten wij dat de operaties som, product en unaire min, bepaalde eigenschappen bezitten. Zo geldt bijvoorbeeld dat $--a = a$, $a{+}b = b{+}a$

en $a(b+c) = ab+ac$. Men noemt dergelijke eigenschappen *algebraïsch*. Ook de logische connectieven voldoen aan een aantal algebraïsche eigenschappen. In de volgende stelling staan de meest bekende bij elkaar.

4.2.1 STELLING *Voor alle $A, B, C \in PROP$ geldt:*

1. *Commutativiteit.*

   $(A \wedge B) \approx (B \wedge A)$,
   $(A \vee B) \approx (B \vee A)$.

2. *Associativiteit.*

   $A \wedge (B \wedge C) \approx (A \wedge B) \wedge C$,
   $A \vee (B \vee C) \approx (A \vee B) \vee C$.

3. *Distributiviteit.*

   $A \wedge (B \vee C) \approx (A \wedge B) \vee (A \wedge C)$,
   $A \vee (B \wedge C) \approx (A \vee B) \wedge (A \vee C)$.

4. *Wetten van De Morgan.*

   $\neg(A \wedge B) \approx (\neg A \vee \neg B)$,
   $\neg(A \vee B) \approx (\neg A \wedge \neg B)$.

5. *Idempotentie.*

   $(A \wedge A) \approx A$,
   $(A \vee A) \approx A$.

6. *Dubbele negatie.*

   $\neg\neg A \approx A$.

7. *Absorptie.*

   $(A \vee \neg A) \vee B \approx (A \vee \neg A)$,
   $(A \wedge \neg A) \wedge B \approx (A \wedge \neg A)$,
   $(A \vee \neg A) \wedge B \approx B$,
   $(A \wedge \neg A) \vee B \approx B$.

BEWIJS Als voorbeeld bewijzen we de tweede wet van De Morgan. De overige bewijzen laten we aan de lezer over. Voor alle valuaties v geldt:

$$\begin{aligned} v(\neg(A \vee B)) = 1 \quad & \Leftrightarrow \quad v(A \vee B) = 0 \\ & \Leftrightarrow \quad v(A) = v(B) = 0 \\ & \Leftrightarrow \quad v(\neg A) = v(\neg B) = 1 \\ & \Leftrightarrow \quad v(\neg A \wedge \neg B) = 1. \end{aligned}$$

Hieruit volgt $\models \neg(A \vee B) \leftrightarrow (\neg A \wedge \neg B)$, zodat $\neg(A \vee B) \approx (\neg A \wedge \neg B)$. ■

4.3 Normaalvormen

In de wiskunde definieert men voor vergelijkingen of formules vaak een standaardvorm. Zo'n standaardvorm noemt men ook wel een *normaalvorm*. Een bekend voorbeeld hiervan is de vierkantsvergelijking, waarvan de normaalvorm luidt: $ax^2 + bx + c = 0$.

Ook in de logica kent men normaalvormen van formules, onder andere de zogenaamde *conjunctieve normaalvorm* (CNV) en de *disjunctieve normaalvorm* (DNV). Het zal blijken dat men aan een formule in de conjunctieve normaalvorm gemakkelijk kan zien of deze een tautologie is en aan een formule in de disjunctieve normaalvorm of deze onvervulbaar is.

4.3.1 DEFINITIE **Normaalvormen**
Zij $F \in PROP$, dan zegt men dat:

1. *F in* conjunctieve normaalvorm *(CNV) staat als F de volgende vorm bezit:*

   $$C_1 \wedge C_2 \wedge \ldots \wedge C_n,$$

   waarin iedere C_i $(1 \leq i \leq n)$ een disjunctie van literalen L_{ij} $(1 \leq j \leq m_i)$ is (zie definitie 2.2.2):

   $$C_i = L_{i1} \vee L_{i2} \vee \ldots \vee L_{im_i}.$$

   De formules C_i noemt men de conjunctieleden *van F.*

2. *F in* disjunctieve normaalvorm *(DNV) staat als F de volgende vorm bezit:*

   $$D_1 \vee D_2 \vee \ldots \vee D_n,$$

   waarin iedere D_i $(1 \leq i \leq n)$ een conjunctie van literalen L_{ij} $(1 \leq j \leq m_i)$ is:

   $$D_i = L_{i1} \wedge L_{i2} \wedge \ldots \wedge L_{im_i}.$$

   De formules D_i noemt men de disjunctieleden *van F.*

Merk op dat een formule F in CNV algemeen geldig is dan en slechts dan als ieder conjunctielid C_i een paar literalen p en $\neg p$ bevat (dit mag voor ieder conjunctielid een ander paar zijn). Analoog geldt: een formule F in DNV is onvervulbaar dan en slechts dan als ieder disjunctielid D_i een paar literalen p en $\neg p$ bevat.

4.3.2 VOORBEELD Formules in CNV.

1. $p \vee \neg p$,
2. $(p \vee \neg q \vee r) \wedge (p \vee \neg r)$.

De eerste formule is algemeen geldig, de tweede niet. ∎

4.3.3 VOORBEELD Formules in DNV.

1. $p \wedge \neg p$,
2. $(p \wedge q \wedge \neg r) \vee (p \wedge \neg r)$,
3. $(p \wedge \neg p \wedge q) \vee (q \wedge \neg q \wedge r)$.

De eerste en de derde formule zijn onvervulbaar. De tweede formule is vervulbaar. Zo is bijvoorbeeld een valuatie v waarvoor $v(p) = v(q) = 1$ en $v(r) = 0$ een model. ∎

4.3.4 STELLING *Voor iedere formule $F \in PROP$ bestaat er een formule F^c in conjunctieve normaalvorm en een formule F^d in disjunctieve normaalvorm zodanig dat $F \approx F^c$ en $F \approx F^d$.*

BEWIJS We zullen deze stelling bewijzen door te laten zien hoe een formule in conjunctieve normaalvorm kan worden omgezet. Deze methode kan eenvoudig worden aangepast om disjunctieve normaalvormen te verkrijgen.

CNV-algoritme

> Herschrijf iedere subformule van F onder gebruikmaking van de volgende equivalenties (van links naar rechts) totdat de formule in CNV staat:

$$\begin{array}{rcl}
A \rightarrow B & \approx & \neg A \vee B \\
\neg(A \rightarrow B) & \approx & A \wedge \neg B \\
\neg(A \wedge B) & \approx & \neg A \vee \neg B \\
\neg(A \vee B) & \approx & \neg A \wedge \neg B \\
A \leftrightarrow B & \approx & (A \rightarrow B) \wedge (B \rightarrow A) \\
\neg(A \leftrightarrow B) & \approx & \neg(A \rightarrow B) \vee \neg(B \rightarrow A) \\
\neg\neg A & \approx & A \\
A \vee (B \wedge C) & \approx & (A \vee B) \wedge (A \vee C) \\
(A \wedge B) \vee C & \approx & (A \vee C) \wedge (B \vee C)
\end{array}$$

Uit stelling 4.1.3 en de substitutiestelling (4.1.10) volgt dat iedere herschrijving met behulp van één van de bovenstaande equivalenties een formule oplevert die equivalent is met de oorspronkelijke. Verder is het niet moeilijk om in te zien dat dit algoritme altijd een CNV oplevert. Immers, als dat niet zo was, dan zou nog minstens éénmaal een herschrijving kunnen worden uitgevoerd. ∎

4.3.5 VOORBEELD Normaalvormen.

1. Een CNV van $p \rightarrow (q \rightarrow p)$ kan als volgt worden verkregen:

$$\begin{aligned} p \rightarrow (q \rightarrow p) &\approx p \rightarrow (\neg q \vee p), \\ &\approx \neg p \vee (\neg q \vee p), \\ &\approx p \vee \neg p \vee \neg q. \end{aligned}$$

   Merk op dat het resultaat ook in DNV staat.

2. Een DNV van $(p \vee \neg q) \rightarrow r$ kan als volgt worden verkregen:

$$\begin{aligned} (p \vee \neg q) \rightarrow r &\approx \neg(p \vee \neg q) \vee r, \\ &\approx (\neg p \wedge \neg\neg q) \vee r, \\ &\approx (\neg p \wedge q) \vee r. \end{aligned}$$

   Om deze formule in CNV te krijgen moet nog de volgende herschrijving worden uitgevoerd:

$$(\neg p \wedge q) \vee r \approx (\neg p \vee r) \wedge (q \vee r). \qquad \blacksquare$$

4.4 Opgaven

1. Bewijs dat de relatie R uit voorbeeld 4.1.6 inderdaad een equivalentierelatie is.

2. Bewijs de gevallen 3, 4(a) en 7 van stelling 4.2.1.

3. Zij $A, B \in PROP$. Ga voor elk van de volgende metabeweringen na of deze juist dan wel onjuist is. Geef in het eerste geval een bewijs; geef in het tweede geval voorbeelden van proposities A en B waarvoor de bewering onjuist is.

   (a) $\models A \wedge B \quad \Rightarrow \quad \models A \quad \& \quad \models B$.

   (b) $\models \neg A \quad \Rightarrow \quad \sim \models A$.

(c) $\sim \models A \quad \Rightarrow \quad \models \neg A$.

(d) $\models A \quad \bigvee \quad \models B \quad \Rightarrow \quad \models A \vee B$.

(e) $\models A \to B \quad \Rightarrow \quad [\models A \quad \Rightarrow \quad \models B\ ]$.

(f) $[\models A \quad \Rightarrow \quad \models B\ ] \quad \Rightarrow \quad \models A \to B$.

(g) $\models A \vee B \quad \& \quad \sim \models A \quad \Rightarrow \quad \models B$.

(h) $\models A \vee B \quad \& \quad \models \neg A \quad \Rightarrow \quad \models B$.

(i) $\models A \quad \Rightarrow \quad \models A \to A$.

(j) $\models A \vee \neg A \quad \Rightarrow \quad \models A$.

(k) $A \models (B \to A) \quad \Leftrightarrow \quad B \models (A \to A)$.

4. Geef het algoritme om formules in DNV te brengen.

5. Zij A, B, C, D propositiesymbolen. Breng de volgende formules in DNV en CNV.

   (a) $A \to (A \wedge B)$.

   (b) $(A \wedge B) \to C$.

   (c) $(\neg A \to B) \to (C \to \neg D)$.

   (d) $A \to (B \to (C \leftrightarrow \neg D))$.

6. Hoe kan men aan een formule in CNV zien of deze een tautologie is? En hoe kan men aan een formule in DNV zien of deze niet vervulbaar is?

Hoofdstuk 5

De Boommethode voor de Propositielogica

In dit hoofdstuk behandelen we een methode waarmee op een effectieve wijze kan worden nagegaan of een redenering logisch geldig is. Deze methode staat bekend als de *boommethode* en is een variant van de zogenaamde *tableaumethode* van de Nederlandse logicus E.W. Beth. De boommethode is een *semantische* methode, waarmee wordt bedoeld dat deze is gebaseerd op semantische concepten.

In §5.1 behandelen we de theorie waarop de boommethode is gebaseerd, en in §5.2 wordt de methode zelf uit de doeken gedaan.

5.1 De onderbouwing van de boommethode

De boommethode is een methode waarmee men kan nagaan of een formule F het logisch gevolg is van een verzameling formules Γ. De eerste stelling in deze paragraaf reduceert deze vraag tot het onvervulbaar zijn van de verzameling $\Gamma \cup \{\neg F\}$ (stelling 5.1.1). Met behulp van de stellingen die hierop volgen, kan het probleem van de onvervulbaarheid van een verzameling complexe formules worden gereduceerd tot de vraag van het onvervulbaar zijn van verzameling(en) van meer eenvoudige formules (corollarium 5.1.5).

5.1.1 STELLING *Zij $\Gamma \subseteq PROP$ en $F \in PROP$, dan geldt:*

$$\Gamma \models F \quad \Leftrightarrow \quad \Gamma \cup \{\neg F\} \textit{ is niet vervulbaar.}$$

BEWIJS

($\Rightarrow$) Zij v een valuatie. Dan zijn er twee mogelijkheden:

- Er is een $A \in \Gamma$ met $v(A) = 0$. Maar dan is v zeker geen model voor $\Gamma \cup \{\neg F\}$.

- Voor alle $A \in \Gamma$ is $v(A) = 1$, ofwel v is een model voor Γ. Uit $\Gamma \models F$ volgt nu dat v ook een model is voor F. Dit betekent dat $v(\neg F) = 0$ en dat wederom v geen model is voor $\Gamma \cup \{\neg F\}$.

Hieruit volgt dat geen enkele v een model is voor $\Gamma \cup \{\neg F\}$, zodat deze verzameling onvervulbaar is.

($\Leftarrow$) Zij v een model voor Γ. In combinatie met het gegeven dat $\Gamma \cup \{\neg F\}$ onvervulbaar is, levert dit $v(\neg F) = 0$. Dit betekent dat $v(F) = 1$, zodat v een model is voor F. We concluderen dat $\Gamma \models F$. ∎

De bovenstaande stelling geeft ons een methode om na te gaan of $\Gamma \models F$. Deze methode luidt: Ga na of de verzameling $\Gamma \cup \{\neg F\}$ vervulbaar is. Is de verzameling *niet* vervulbaar, dan geldt inderdaad $\Gamma \models F$. Is deze verzameling *wel* vervulbaar, dan geldt *niet* $\Gamma \models F$.

Om te bepalen of een verzameling vervulbaar is, komen de volgende stellingen van pas. Deze vertellen ons precies wanneer een valuatie v een gegeven verzameling formules vervult, of, anders gezegd, wanneer v een model is voor die verzameling. In deze stellingen wordt dit probleem herleid tot de vraag of v een verzameling (of verzamelingen) van meer eenvoudige formules vervult.

5.1.2 STELLING *Zij $\Gamma \subseteq PROP$, $A, B \in PROP$ en $A \approx B$. Zij verder v een valuatie. Dan geldt:*

$$v \textit{ vervult } \Gamma \cup \{A\} \qquad \Leftrightarrow \qquad v \textit{ vervult } \Gamma \cup \{B\}.$$

BEWIJS

($\Rightarrow$) Stel v vervult $\Gamma \cup \{A\}$. Dit betekent dat $v(A) = 1$ en $v(F) = 1$ voor alle $F \in \Gamma$. Aangezien $A \approx B$, volgt hieruit dat ook $v(B) = 1$. Derhalve is v een model voor $\Gamma \cup \{B\}$.

($\Leftarrow$) Op dezelfde wijze. ∎

5.1.3 STELLING *Zij $\Gamma \subseteq PROP$ en $A, B \in PROP$. Zij verder v een valuatie. Dan geldt:*

$$v \textit{ vervult } \Gamma \cup \{A \vee B\} \qquad \Leftrightarrow \qquad v \textit{ vervult } \Gamma \cup \{A\} \textit{ of } \Gamma \cup \{B\}.$$

BEWIJS

($\Rightarrow$) Stel v vervult $\Gamma \cup \{A \vee B\}$. Dit betekent dat $v(A \vee B) = 1$ en $v(F) = 1$ voor alle $F \in \Gamma$. Hieruit volgt dat $v(A) = 1$ of $v(B) = 1$. Derhalve, is v een model voor $\Gamma \cup \{A\}$ of $\Gamma \cup \{B\}$.

($\Leftarrow$) Omgekeerd, als v een model is voor $\Gamma \cup \{A\}$ of $\Gamma \cup \{B\}$, dan geldt dat $v(F) = 1$ voor alle $F \in \Gamma$, en dat $v(A) = 1$ of $v(B) = 1$. Uit het laatste volgt dat $v(A \vee B) = 1$. Derhalve is v een model voor $\Gamma \cup \{A \vee B\}$. ∎

Het bewijs van de volgende stelling, dat volkomen analoog is aan het bewijs van de stellingen 5.1.2 en 5.1.3, wordt aan de lezer overgelaten.

5.1.4 STELLING *Zij $\Gamma \subseteq PROP$ en $A, B \in PROP$. Zij verder v een valuatie. Dan geldt:*

$$v \textit{ vervult } \Gamma \cup \{A \wedge B\} \quad \Leftrightarrow \quad v \textit{ vervult } \Gamma \cup \{A, B\}.$$

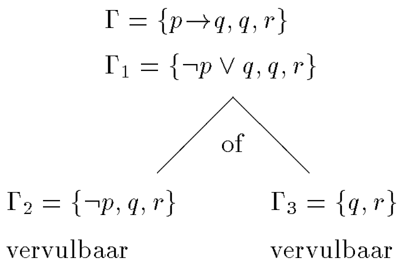
Uit de stellingen 5.1.2, 5.1.3 en 5.1.4 volgt direct wanneer een verzameling formules *onvervulbaar* is. De resultaten zijn bijeen gezet in het volgende corollarium.

5.1.5 COROLLARIUM *Zij $\Gamma \subseteq PROP$, $A, B, C, D \in PROP$ en $A \approx B$. Dan geldt:*

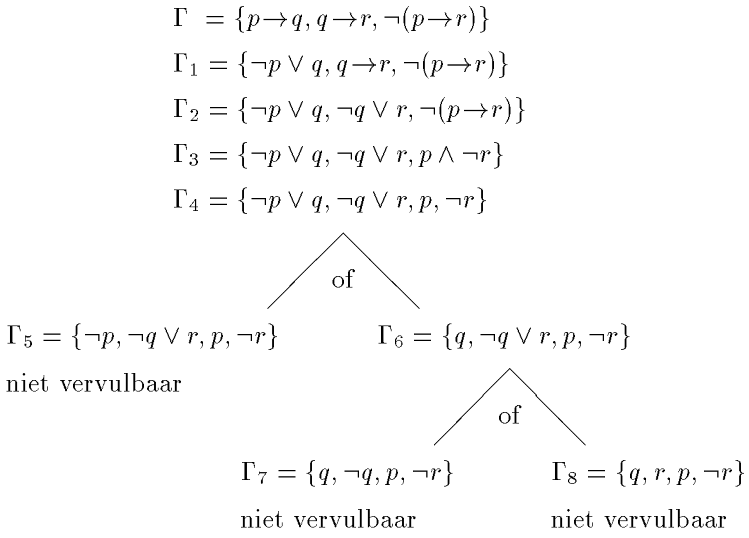
1. $\Gamma \cup \{A\}$ *is onvervulbaar* $\Leftrightarrow$ $\Gamma \cup \{B\}$ *is onvervulbaar.*
2. $\Gamma \cup \{C \vee D\}$ *is onvervulbaar* $\Leftrightarrow$ $\Gamma \cup \{C\}$ *en* $\Gamma \cup \{D\}$ *zijn beide onvervulbaar.*
3. $\Gamma \cup \{C \wedge D\}$ *is onvervulbaar* $\Leftrightarrow$ $\Gamma \cup \{C, D\}$ *is onvervulbaar.*

5.1.6 VOORBEELD Vervulbare en onvervulbare verzamelingen.

1. De verzameling $\{p, \neg p\}$ is onvervulbaar.
2. De verzameling $\{p, q, r\}$ wordt vervuld door een valuatie v waarvoor $v(p) = v(q) = v(r) = 1$.
3. De verzameling $\Gamma = \{p \rightarrow q, q, r\}$ is vervulbaar. Door toepassing van de stellingen uit deze paragraaf kan dit op systematische wijze worden geverifieerd. Zie hiervoor figuur 5.1. Het plaatje kan als volgt worden gelezen. Γ is vervulbaar dan en slechts dan als Γ_1 vervulbaar is (stelling 5.1.2). Γ_1, op haar beurt, is vervulbaar dan en slechts dan als Γ_2 of Γ_3 vervulbaar is (stelling 5.1.3). Nu wordt Γ_2 vervuld door een valuatie v waarvoor $v(p) = 0$, $v(q) = 1$ en $v(r) = 1$. Hieruit volgt dus dat Γ vervuld wordt door v. Echter, ook Γ_3 is vervulbaar. Neem bijvoorbeeld een valuatie w waarvoor $w(q) = w(r) = 1$. Dit alles betekent dat Γ vervuld wordt door v en door w.



Figuur 5.1: $\Gamma = \{p \to q, q, r\}$ is vervulbaar.



Figuur 5.2: $\Gamma = \{p \to q, q \to r, \neg(p \to r)\}$ is niet vervulbaar.

4. Bewijs dat $p \rightarrow q, q \rightarrow r \models p \rightarrow r$. Wegens stelling 5.1.1 volstaat het om te laten zien dat de verzameling $\Gamma = \{p \rightarrow q, q \rightarrow r, \neg(p \rightarrow r)\}$ niet vervulbaar is. In figuur 5.2 is de bijbehorende boom weergegeven. Uit dit plaatje blijkt dat de verzamelingen Γ_7 en Γ_8 beide niet vervulbaar zijn. Maar dan is de verzameling Γ_6 ook niet vervulbaar. Aangezien Γ_5 niet vervulbaar is, moet vervolgens Γ_4 eveneens onvervulbaar zijn. Dit betekent echter dat de verzamelingen Γ_3, Γ_2, Γ_1 en Γ dan ook niet vervulbaar zijn. Hieruit volgt $p \rightarrow q, q \rightarrow r \models p \rightarrow r$. ■

Uit voorbeeld 5.1.6 blijkt, dat men voor de beantwoording van de vraag 'Is Γ vervulbaar?' de formules uit Γ of uit de hieruit afgeleide verzamelingen, systematisch afbreekt tot literalen. Gedurende dit 'afbraakproces' worden verzamelingen door andere verzamelingen vervangen die slechts in één of twee formules verschillen van de oorspronkelijke. Het proces stopt als alle verkregen verzamelingen onvervulbaar zijn of als geen enkele verzameling verder afbreekbaar is. De oorspronkelijke verzameling is onvervulbaar als alle verkregen verzamelingen onvervulbaar zijn, en vervulbaar als minstens één van deze verzamelingen vervulbaar is. Deze methode om na te gaan of een eindige verzameling Γ vervulbaar is of niet, zullen we nu precies omschrijven:

1. Pas stelling 5.1.2 net zo vaak toe als het mogelijk is één van de volgende formules links te vervangen door de ermee equivalente formule rechts:

$$\begin{array}{rcl} \neg(A \wedge B) & \approx & \neg A \vee \neg B, \\ \neg(A \vee B) & \approx & \neg A \wedge \neg B, \\ A \rightarrow B & \approx & \neg A \vee B, \\ \neg(A \rightarrow B) & \approx & A \wedge \neg B, \\ A \leftrightarrow B & \approx & (A \rightarrow B) \wedge (B \rightarrow A), \\ \neg(A \leftrightarrow B) & \approx & \neg(A \rightarrow B) \vee \neg(B \rightarrow A). \end{array}$$

2. Pas zo vaak als mogelijk de stellingen 5.1.3 en 5.1.4 toe. Bij toepassing van stelling 5.1.3 ontstaat een *vertakking*.

3. Herhaal de stappen 1 en 2 totdat iedere verkregen verzameling een paar complementaire formules F en $\neg F$ bevat, of slechts uit literalen bestaat.

4. Als alle verkregen verzamelingen een paar complementaire formules bevatten, dan is Γ onvervulbaar. Als een verkregen verzameling Γ_i geen complementaire formules bevat en alleen uit literalen bestaat, dan is een valuatie v waarvoor $v(A) = 1$ voor alle $A \in \Gamma_i$ een model voor Γ.

$\neg$-regel	**$\wedge$-regels**		**$\vee$-regels**	
$\neg\neg A$ $\vdots$ $\boldsymbol{A}$	$A \wedge B$ $\vdots$ $\boldsymbol{A}$ $\boldsymbol{B}$	$\neg(A \wedge B)$ $\vdots$ $\boldsymbol{\neg A \vee \neg B}$	$A \vee B$ $\vdots$ $\diagup\ \diagdown$ $\boldsymbol{A}\quad\boldsymbol{B}$	$\neg(A \vee B)$ $\vdots$ $\boldsymbol{\neg A \wedge \neg B}$

$\rightarrow$-regels		**$\leftrightarrow$-regels**	
$A \rightarrow B$ $\vdots$ $\boldsymbol{\neg A \vee B}$	$\neg(A \rightarrow B)$ $\vdots$ $\boldsymbol{A \wedge \neg B}$	$A \leftrightarrow B$ $\vdots$ $\boldsymbol{A \rightarrow B}$ $\boldsymbol{B \rightarrow A}$	$\neg(A \leftrightarrow B)$ $\vdots$ $\boldsymbol{\neg(A \rightarrow B) \vee \neg(B \rightarrow A)}$

Tabel 5.1: De regels van de *boommethode*.

Het is evident dat dit proces altijd in een eindig aantal stappen stopt. Immers, door bovenstaande methode worden alle formules in Γ —dit zijn er eindig veel— afgebroken tot literalen. Als dit is gebeurd, zijn de stappen 1 en 2 hierboven niet meer toepasbaar, zodat het proces termineert. Men zegt daarom dat de vraag 'Is Γ vervulbaar?' *beslisbaar* is.

5.2 Hoe men bomen maakt

In de vorige paragraaf hebben we gezien hoe we met behulp van de stellingen 5.1.2, 5.1.3 en 5.1.4 stelselmatig en in een eindig aantal stappen kunnen nagaan of een gegeven eindige verzameling $\Gamma \subset PROP$ vervulbaar of onvervulbaar is. De methode is notationeel echter nogal omslachtig: iedere keer dat een verzameling wordt vervangen door een of twee andere verzamelingen worden alle formules, behalve de formule die wordt afgebroken, ongewijzigd overgenomen. De *boommethode* die in deze paragraaf zal worden beschreven, heeft dit bezwaar niet. In feite is het een handiger manier om het stelsel verzamelingen in het afbraakproces uit de vorige paragraaf te noteren.

Bij de boommethode worden voor het afbreken van de formules uit de verzameling Γ de regels uit tabel 5.1 gebruikt. Deze regels dient men van boven naar beneden te lezen. De eerste formule in elke regel duidt de *premisse* aan, terwijl de overige formules, die vet zijn gedrukt, de *conclusies* zijn. Voor A en B dient men daarbij formules uit $PROP$ te lezen. De premisse van een regel geeft aan op welke formules men de regel mag toepassen. Wil men dus bijvoorbeeld de $\neg$-regel toepassen op een zekere formule F, dan dient F de formule $\neg\neg A$ te zijn voor zekere $A \in PROP$. Het resultaat van de toepassing van die regel is dan de formule A. Het is *niet* toegestaan om een regel toe te passen op echte subformules van een formule. Zo is de $\neg$-regel niet van toepassing op de formule $\neg\neg A \wedge B$ omdat hierin $\neg\neg A$ een echte subformule is. Wel is de eerste $\wedge$-regel in dit geval toepasbaar, met als conclusies $\neg\neg A$ en B.

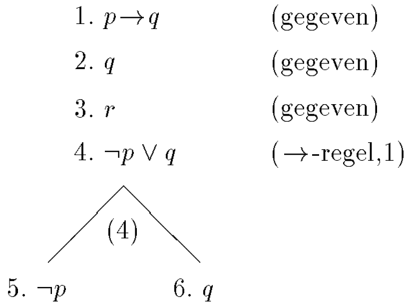
In feite geven de regels bijzondere gevallen van de stellingen 5.1.2, 5.1.3 en 5.1.4 weer. Een toepassing van de tweede $\wedge$-regel komt bijvoorbeeld neer op het toepassen van stelling 5.1.2, waarbij men voor 'A' de formule $\neg(A \wedge B)$ neemt en voor 'B' het daarmee equivalente $\neg A \vee \neg B$. Wat de verzameling 'Γ' hierbij is, wordt in het midden gelaten. Het doet er ook niet toe, want de formules in deze verzameling veranderen niet bij toepassing van de stelling.

Op analoge manier kan men de $\neg$-regel, de tweede $\vee$-regel, de $\rightarrow$-regels en de tweede $\leftrightarrow$-regel lezen. De eerste $\wedge$-regel en de eerste $\leftrightarrow$-regel zijn gebaseerd op stelling 5.1.4, terwijl de eerste $\vee$-regel op stelling 5.1.3 is terug te voeren. Deze laatste regel geeft aanleiding tot een *vertakking* die volkomen vergelijkbaar is met de 'of'-vertakkingen die men in de figuren 5.1 en 5.2 aantreft.

Bij toepassing van de boommethode worden de formules uit de verzameling Γ onder elkaar geplaatst en vervolgens met behulp van de regels uit figuur 5.1 afgebroken, totdat er geen enkele regel meer toepasbaar is. Indien de eerste $\vee$-regel wordt toegepast op een formule van de vorm $A \vee B$, ontstaat er een vertakking, waarbij de formule A links, en de formule B rechts onder de vertakking geplaatst dient te worden. Op deze wijze ontstaat er een *boom*, waarvan de bovenste formule de *wortel* van de boom wordt genoemd. Formules onderaan de boom die geen afstammelingen bezitten, noemt men de *bladeren* van de boom. Een *tak* van de boom is een pad dat bij de wortel begint en eindigt bij een blad, zodanig dat iedere knoop op dat pad slechts één keer wordt gepasseerd.

Als dan iedere tak van de aldus geconstrueerde boom een paar complementaire formules F en $\neg F$ bevat, dan is de oorspronkelijke verzameling Γ onvervulbaar. In dit geval zegt men dat de boom *sluit*. Als er echter een tak bestaat die geen paar complementaire formules bevat, dan is Γ wél vervulbaar. Zo'n tak, waarvan we zeggen dat hij niet sluit of *open* blijft, bevat dan alle informatie die nodig is om een model voor Γ te construeren. Een valuatie v waarvoor $v(A) = 1$ voor alle literalen A in de open blijvende tak is dan namelijk een model voor Γ. Indien een propositiesymbool p of zijn ontkenning niet voorkomt in deze tak, dan kan $v(p)$ willekeurig worden gekozen.

De boommethode kan nu als volgt worden gebruikt om de geldigheid van een redenering van de vorm $\Gamma, \therefore F$ te testen:



Figuur 5.3: Een boom voor $\Gamma = \{p \to q, q, r\}$.

- Ga na of de verzameling $\Gamma \cup \{\neg F\}$ vervulbaar is door een boom te construeren.

- Sluiten alle takken van de boom doordat deze een paar complementaire formules bevatten, dan is de verzameling niet vervulbaar, hetgeen betekent dat de gegeven redenering logisch geldig is (stelling 5.1.1).

- Is er een tak die open blijft en zijn alle mogelijkheden om regels toe te passen uitgeput, dan is de verzameling $\Gamma \cup \{\neg F\}$ vervulbaar en levert de betreffende tak een model voor deze verzameling. Dit model is dan een *tegenvoorbeeld* voor de gegeven redenering.

Aan de hand van een aantal voorbeelden zullen we dit alles toelichten. We introduceren eerst nog een nieuwe notatie. Met $B(\Gamma)$ zullen we in het vervolg een boom voor Γ aanduiden. Aangezien er bij een verzameling meerdere verschillende bomen geconstrueerd kunnen worden —afhankelijk van de volgorde waarin de regels worden toegepast—, is $B(\Gamma)$ een metavariabele waarvan de waarden bomen voor Γ zijn.

5.2.1 VOORBEELD $\Gamma = \{p \to q, q, r\}$ is vervulbaar.
In figuur 5.3 is een boom voor de verzameling Γ weergegeven. Merk op dat de formules uit de verzamelingen Γ_i behorende bij de boom uit figuur 5.1 hier 'verticaal' als labels bij de knopen genoteerd zijn. Geen van de takken van de boom bevat een paar complementaire formules, zodat de boom $B(\Gamma)$ niet sluit.

Om een model voor Γ te verkrijgen 'lezen' we een open blijvende tak af. Met iedere open blijvende tak correspondeert een (mogelijk verschillend) model.

Een valuatie v is zo'n model, als deze zo gekozen is dat $v(A) = 1$ voor alle literalen A in een open blijvende tak van $B(\Gamma)$.

De boom in dit voorbeeld heeft twee open blijvende takken. Aflezen van de linker tak levert op dat een valuatie v waarbij $v(p) = 0$ en $v(q) = v(r) = 1$, een model is voor Γ. Ook de rechter tak levert een model: een valuatie w waarvoor $w(q) = w(r) = 1$ (hierbij is $w(p)$ vrij te kiezen). Uit het bovenstaande volgt dat —in dit voorbeeld— een model voor de formules in de linker tak tevens een model is voor de formules in de rechter tak.

Merk op dat alle formules in de boom *geannoteerd* zijn. Zo verwijst de annotatie ($\rightarrow$-regel,1) op regel 4 naar een toepassing van de eerste $\rightarrow$-regel op de formule op regel 1. De vermelding '(4)' onder de vertakking verwijst naar een toepassing van de eerste $\vee$-regel op de formule op regel 4. ■

5.2.2 VOORBEELD $p \rightarrow q, q \rightarrow r \models p \rightarrow r$
We zullen dit aantonen door met behulp van de boommethode te laten zien dat de verzameling $\Gamma = \{p \rightarrow q, q \rightarrow r, \neg(p \rightarrow r)\}$ onvervulbaar is (zie ook voorbeeld 5.1.6). De boom $B(\Gamma)$ is weergegeven in figuur 5.4. Alle takken van $B(\Gamma)$ sluiten. Dit is hier aangegeven door $\mathbf{X}(m, n)$, hetgeen betekent dat de betreffende tak sluit op de formules met de nummers m en n. Hieruit volgt dat $\Gamma = \{p \rightarrow q, q \rightarrow r, \neg(p \rightarrow r)\}$ onvervulbaar is, zodat $p \rightarrow q, q \rightarrow r \models p \rightarrow r$. Vergelijk de boom $B(\Gamma)$ met de boom in figuur 5.2! ■

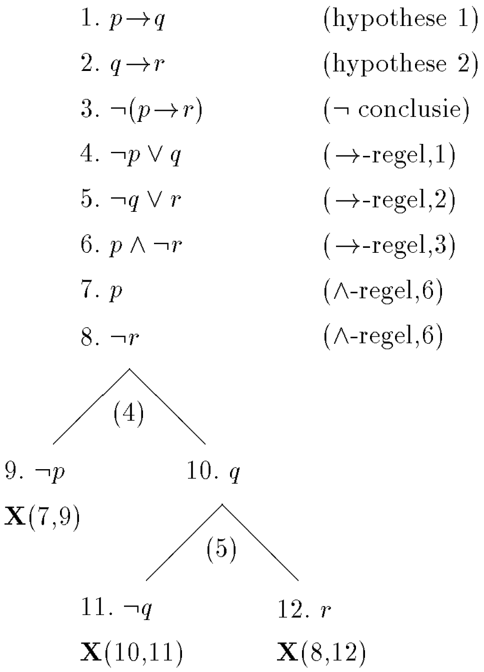
5.2.3 VOORBEELD *niet* $\models (p \rightarrow q) \rightarrow (q \rightarrow p)$.
We laten zien hoe met behulp van de boommethode een tegenvoorbeeld voor de uitspraak $\models (p \rightarrow q) \rightarrow (q \rightarrow p)$ verkregen kan worden. De boom in figuur 5.5 laat zien dat de verzameling $\Gamma = \{\neg[(p \rightarrow q) \rightarrow (q \rightarrow p)]\}$ vervulbaar is. Geen van de takken van de boom $B(\Gamma)$ sluit. Een valuatie v die een model is voor Γ lezen we bijvoorbeeld af uit de linker tak: $v(p) = 0$ en $v(q) = 1$ (de rechter tak geeft dezelfde valuatie). Hiermee is bewezen dat *niet* $\models (p \rightarrow q) \rightarrow (q \rightarrow p)$. ■

Uit het voorafgaande kan de volgende stelling voor de boommethode worden afgeleid.

5.2.4 STELLING *Voor iedere eindige niet-lege verzameling* $\Gamma \subset PROP$ *geldt:*

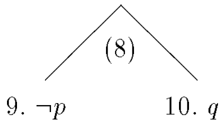
$$\Gamma \textit{ is onvervulbaar} \quad \Leftrightarrow \quad \textit{iedere boom } B(\Gamma) \textit{ sluit.}$$

Dit betekent dat als één der takken van een boom $B(\Gamma)$ niet sluit, de verzameling Γ vervulbaar is. Het maakt ook niet uit in welke volgorde men de regels van de boommethode toepast bij het construeren van een boom; alleen is de ene manier bewerkelijker dan de andere. Over het algemeen verkrijgt men de eenvoudigste boom door het toepassen van de $\vee$-regel, waarbij een vertakking optreedt, zolang mogelijk uit te stellen.



Figuur 5.4: $p \rightarrow q, q \rightarrow r \models p \rightarrow r$.

1. $\neg[(p\rightarrow q)\rightarrow(q\rightarrow p)]$	($\neg$ conclusie)
2. $(p\rightarrow q)\wedge\neg(q\rightarrow p)$	($\rightarrow$-regel,1)
3. $p\rightarrow q$	($\wedge$-regel,2)
4. $\neg(q\rightarrow p)$	($\wedge$-regel,2)
5. $q\wedge\neg p$	($\rightarrow$-regel,4)
6. q	($\wedge$-regel,5)
7. $\neg p$	($\wedge$-regel,5)
8. $\neg p\vee q$	($\rightarrow$-regel,3)



Figuur 5.5: $\Gamma=\{\neg[(p\rightarrow q)\rightarrow(q\rightarrow p)]\}$ is vervulbaar.

5.3 Opgaven

1. Zij $A,B,C,D\in PROP$. Bewijs met de boommethode dat:
   (a) $A,\ (A\wedge\neg B)\rightarrow C,\ \neg(A\wedge C)\models B$.
   (b) $\neg A\leftrightarrow(B\wedge C),A\wedge C\models\neg B$.
   (c) $A\leftrightarrow(\neg B\vee\neg C),\neg(A\rightarrow\neg C)\models\neg B$.
   (d) $A\rightarrow((B\rightarrow C)\rightarrow D),\neg(A\rightarrow D)\models\neg(B\rightarrow C)$.
   (e) $(A\wedge B)\rightarrow C,\neg A\vee(A\rightarrow B)\models A\rightarrow C$.
   (f) $(A\rightarrow\neg C)\vee(A\rightarrow B),C\rightarrow A\models C\rightarrow B$.

2. Ga voor elk van de onderstaande metabeweringen na of deze juist dan wel onjuist is. Geef in het eerste geval een bewijs met behulp van de boommethode. Construeer in het tweede geval een *tegenvoorbeeld*. Gegeven is dat $A,B,C,D,E\in PROP$.
   (a) $A\rightarrow B,\neg C\rightarrow B\models A\rightarrow C$.
   (b) $\models((A\vee(A\rightarrow B))\rightarrow B)\rightarrow B$.
   (c) $(A\wedge B)\rightarrow D,\neg(D\rightarrow E),B\vee E\models\neg A$.
   (d) $\neg A\rightarrow A\models A$.

   (e) $\models (A \vee B) \rightarrow (A \wedge B)$.
   (f) $\neg A, A \models B$.

3. Vertaal de volgende redeneringen naar *propositielogica* en ga met behulp van de boommethode na of deze logisch geldig zijn of niet. Produceer in het laatste geval ook een tegenvoorbeeld.

   (a) *Als Tom nuchter is, dan handelt hij rationeel maar hij is dan wel oninteressant.*
   Als hij beleefd en oninteressant is, dan vindt Sylvia hem niet aardig.
   Dus, als Tom nuchter en beleefd is, dan vindt Sylvia hem niet aardig.

   (b) *Als Anton lid wordt van de Nederlandse Vereniging van Terraszitters, dan worden Bob en Cees dat ook.*
   Als Anton of Bob lid wordt, dan zal David zijn lidmaatschap opzeggen.
   Dus zal David zijn lidmaatschap opzeggen, ongeacht of Cees nou lid wordt of niet.

   (c) *Je kunt Hans geloven als je Bert kunt geloven, en omgekeerd.*
   Als je noch Hans, noch Bert kunt geloven, dan kun je ook Greetje niet geloven.
   Je kunt Greetje geloven.
   Dus kun je Bert geloven.

Hoofdstuk 6

Natuurlijke Deductie volgens Fitch

In de vorige hoofdstukken is de propositielogica hoofdzakelijk vanuit een semantisch standpunt beschouwd. De sleutelconcepten hierbij waren *valuatie*, *model*, *tautologie* en *logisch gevolg*. In de semantiek van de logica is echter het trekken van een conclusie uit gegeven premissen, zo kenmerkend voor het menselijk wiskundig redeneren, nauwelijks herkenbaar.

De methode van de *natuurlijke deductie* is een formele opbouw van de logica, waarin het trekken van conclusies uit premissen centraal staat. Deze methode is voor het eerst formeel beschreven door de Duitse logicus G. Gentzen (1934). In dit hoofdstuk zullen we het natuurlijke deductiesysteem van F.B. Fitch (1952), in het vervolg aangeduid als *systeem* $\mathcal{F}$, bespreken.

De volgende onderwerpen zullen aan de orde komen: de afleidingsregels van systeem $\mathcal{F}$ en het begrip 'afleiding' (§6.1), afleidingsstrategieën en voorbeelden van afleidingen (§6.2), en uitbreidingen van het systeem van Fitch (§6.3).

6.1 Afleidingsregels en afleidingen

Het systeem $\mathcal{F}$ is zuiver syntactisch, hetgeen wil zeggen dat het alleen gaat om de *vorm* van de formules. Er behoeft dus geen beroep gedaan te worden op het waarheidsbegrip, zoals in het vorige hoofdstuk. Een karakterisering van het systeem $\mathcal{F}$ omvat een opsomming van de zogenaamde *afleidingsregels* die gebruikt mogen worden, en een definitie die vastlegt wat onder een *afleiding* wordt verstaan. Om een intuïtief idee te krijgen wat een afleiding in systeem $\mathcal{F}$ is, bekijken we een voorbeeld. Stel we willen de volgende uitspraak bewijzen:

> Indien geldt als het regent dan wordt de straat nat en de straat wordt niet nat, dan regent het niet.

Deze uitspraak kunnen we vertalen als $[(p \rightarrow q) \wedge \neg q] \rightarrow \neg p$. De afleiding van deze formule in het systeem $\mathcal{F}$ staat in figuur 6.1. De redenering die door deze afleiding wordt weergegeven, kan als volgt worden geparafraseerd.

1.	$(p \rightarrow q) \wedge \neg q$	(hypothese 1)
2.	$\quad p$	(hypothese 2)
3.	$\quad (p \rightarrow q) \wedge \neg q$	(rei,1)
4.	$\quad p \rightarrow q$	($\wedge$-elim,3)
5.	$\quad q$	($\rightarrow$-elim,2,4)
6.	$\quad \neg q$	($\wedge$-elim,3)
7.	$\neg p$	($\neg$-intro,2,5,6)
8.	$[(p \rightarrow q) \wedge \neg q] \rightarrow \neg p$	($\rightarrow$-intro,1,7)

Figuur 6.1: Afleiding van $[(p \rightarrow q) \wedge \neg q] \rightarrow \neg p$.

1. Stel: als het regent wordt de straat nat, en de straat wordt niet nat; ofwel $(p \rightarrow q) \wedge \neg q$. We willen dan laten zien dat het niet regent, ofwel $\neg p$.

2. Neem vervolgens aan dat het regent, ofwel p. De bedoeling is om te laten zien dat deze aanname fout is door er een tegenspraak uit af te leiden.

3. Omdat we in 1 hebben gesteld dat $(p \rightarrow q) \wedge \neg q$ het geval is, mogen we dat onder de additionele aanname 2 nog steeds aannemen.

4. Maar dan geldt als het regent wordt de straat nat, ofwel $p \rightarrow q$.

5. Aangezien we p hebben gesteld in 2, en we in 4 zagen dat $p \rightarrow q$ het geval is, leiden we af dat de straat nat wordt: q.

6. Uit 3 volgt ook dat de straat niet nat wordt: $\neg q$.

7. Nu bevinden we ons in de situatie dat we uit aanname 2 dat p geldt, tegenstrijdige zaken hebben afgeleid, te weten q (in 5) en $\neg q$ (in 6). Dit betekent dat de genoemde aanname verkeerd moet zijn geweest, zodat we concluderen dat het niet regent: $\neg p$. We hebben nu aanname 2 weerlegd!

8. Al redenerende hebben we uit aanname 1 dat $(p \rightarrow q) \wedge \neg q$ het geval is, afgeleid dat $\neg p$ het geval moet zijn (zie 7). Dit betekent dat moet gelden $[(p \rightarrow q) \wedge \neg q] \rightarrow \neg p$. Het bewijs is voltooid.

Merk op dat de afleiding in figuur 6.1 er veel overzichtelijker uitziet dan haar geparafraseerde versie. De verticale strepen in de figuur geven aan hoe lang een aanname aktief is. De horizontale streep behorende bij een verticale streep markeert de aanname. Zo is de aanname dat p het geval is, aktief in de regels met de nummers 2 tot en met 6. De aanname dat $(p \rightarrow q) \wedge \neg q$ het geval is, is het langst aktief: in de regels met de nummers 1 tot en met 7. In regel 8 is deze aanname dus niet langer geldig. Het resultaat $[(p \rightarrow q) \wedge \neg q] \rightarrow \neg p$ geldt onvoorwaardelijk. De aannamen 1 en 2 zijn alleen maar tussentijdse aannamen die in het bewijs worden gebruikt. Een andere eigenschap van de afleiding is dat iedere formule ofwel een aanname is, ofwel verkregen is uit eerdere formules door toepassing van een zogenaamde *afleidingsregel* (zie verderop in deze paragraaf). Op dit moment volstaan we met op te merken dat een afleidingsregel een redeneerwijze formaliseert. Voor de duidelijkheid zijn alle formules uit de afleiding *geannoteerd*. Als een formule een hypothese is, wordt dit vermeld met het volgnummer van die hypothese. Als een formule is afgeleid door toepassing van een afleidingsregel, dan wordt de afleidingsregel genoemd samen met de regelnummers van de formules waaruit de formule is afgeleid. Deze annotaties behoren formeel gesproken niet tot de afleiding.

Voordat we met mathematische precisie het begrip *afleiding* en de begrippen die hierbij een rol spelen, zullen definiëren, bespreken we deze eerst op informele wijze. Neem aan dat alle formules in een afleiding als volgt oplopend genummerd zijn: $F_1, F_2, \ldots, F_n$. Omdat een formule meerdere keren in dezelfde afleiding kan voorkomen (bijvoorbeeld de formules F_1 en F_3 in de afleiding in figuur 6.1), zullen we formules identificeren met hun nummer. Als $i < j$, dan zeggen we dat formule F_i *vooraf gaat* aan F_j. De *graad* van de formule F_i, notatie $gr(i)$, is gedefinieerd als het aantal verticale strepen links van F_i. De graad van de formule F_i geeft dus aan hoeveel aannamen aktief zijn en mogelijkerwijs gebruikt om F_i af te leiden. Zo geldt bijvoorbeeld voor de formule $F_4 = p \rightarrow q$ in onze voorbeeld-afleiding dat $gr(4) = 2$.

Zij $F_k \in PROP$ een hypothese (aanname) op regel k, zodat F_k de eerste formule is rechts van een (nieuwe) verticale streep en zij $F_l \in PROP$ op regel l de laatste formule rechts van de genoemde verticale streep. Het *hypothese-interval* van F_k is dan gedefinieerd als het discrete, gesloten interval $[k, l] \subset \mathbb{N}$. Het hypothese-interval behorende bij een hypothese bepaalt welke formules afgeleid zijn onder die hypothese. In figuur 6.1 is er sprake van twee hypothese-intervallen, namelijk $[1, 7]$ en $[2, 6]$ met als hypothesen de formules op de regels 1, respectievelijk 2. We zien tevens dat hypothese-intervallen op een bepaalde wijze genest kunnen zijn, zoals hier $[2, 6] \subset [1, 7]$. We zullen van twee hypothese-intervallen eisen dat deze ofwel genest zijn zoals in ons voorbeeld, ofwel geheel disjunct zijn. Ook dienen ze minstens twee formules te bevatten. Als de gehele afleiding geen hypothese-interval is —omdat er geen verticale streep vóór de hele afleiding staat—, dan spreekt men over een *nulde interval*. In ons voorbeeld is $[1, 8]$ dus een nulde interval. Verder zeggen we dat een formule F_i *in* het interval I *ligt*, als I het kleinste interval is zodanig dat $i \in I$. Zo ligt de formule F_4 bijvoorbeeld wel in het interval $[2, 6]$, maar niet in

$[1,7]$. Op dezelfde wijze kan men definiëren wanneer een hypothese-interval I in een interval J ligt. Het interval $[2,6]$ ligt bijvoorbeeld in het interval $[1,7]$.

Het laatste begrip dat we nodig hebben is het begrip *bewijsfiguur*. Het idee is dat een bewijsfiguur een stelsel hypothese-intervallen met formules is, waarbij de hypothese-intervallen voldoen aan de hierboven gestelde eisen: op de juiste wijze genest òf disjunct, en minstens twee formules bevattend. Een bewijsfiguur is een blauwdruk voor een afleiding. Iedere afleiding is een bewijsfiguur, maar niet iedere bewijsfiguur is een afleiding.

6.1.1 Definitie **Bewijsfiguur**

Een bewijsfiguur $\mathbf{D}$ *is een mathematische structuur bestaande uit:*

1. *een gesloten interval* $D = [1,n]$, *waarbij* $D \subset \mathbb{N}$,
2. *een functie* $F : D \to PROP$, *en*
3. *een collectie* $\mathbf{H}$ *van gesloten deelintervallen van* D, *zodanig dat voor elk interval* $[i,j] \in \mathbf{H}$ *geldt dat* $i < j$, *en zodanig dat voor elk tweetal verschillende intervallen* $[i,j],[k,l] \in \mathbf{H}$ *geldt dat* $i < k < l \leq j$, *òf* $k < i < j \leq l$, *òf* $[i,j] \cap [k,l] = \emptyset$.

Als $i \in D$, *dan duidt* $F(i)$, *in het vervolg genoteerd als* F_i, *de formule aan op regel* i *van de bewijsfiguur. Men zegt dat* F_i voorafgaat *aan* F_j *als* $i < j$.

De elementen van $\mathbf{H}$ *noemt men de* hypothese-intervallen *van de bewijsfiguur. Indien* $D \notin \mathbf{H}$, *dan wordt* D *een* nulde *interval genoemd. Als* $[k,l] \in \mathbf{H}$ *dan noemt men de formule* F_k *de* hypothese *van* $[k,l]$.

Als $i \in I$ *voor een zeker interval* $I \in \mathbf{H} \cup \{D\}$ *en er geen* $J \in \mathbf{H}$ *bestaat zodanig dat* $i \in J \subset I$, *dan zegt men dat de formule* F_i in I ligt, *notatie* $F_i \in I$. *Een hypothese-interval* I ligt in *het interval* $J \in \mathbf{H} \cup \{D\}$ *als* $I \subset J$ *en er geen* $K \in \mathbf{H}$ *bestaat zodanig dat* $I \subset K \subset J$. *(N.B. We gebruiken '*$A \subset B$*' in de betekenis* $A \subseteq B$ *en* $A \neq B$*.)*

De lengte $len(\mathbf{D})$ *van de bewijsfiguur* $\mathbf{D}$ *is gedefinieerd als* $len(\mathbf{D}) = n$; *de* lengte $len(I)$ *van een interval* $I = [k,l] \in \mathbf{H} \cup \{D\}$ *is gedefinieerd als* $len([k,l]) = k - l + 1$.

Tenslotte is de graad *van een formule* F_i, *notatie* $gr(i)$, *gedefinieerd door* $gr(i) = \mathsf{card}\,\{I \in \mathbf{H} \mid i \in I\}$, *waarin* card *de* cardinaliteit *(aantal elementen) van een verzameling aanduidt.*

In de definities 6.1.3 en 6.1.4 zal worden beschreven wanneer een bewijsfiguur een afleiding is. In definitie 6.1.2 wordt vastgelegd hoe de afleidingsregels van systeem $\mathcal{F}$ die in tabel 6.1 staan, in een bewijsfiguur mogen worden toegepast. Maar eerst bespreken van de afleidingsregels op informele wijze.

Ten aanzien van de notatie van de afleidingsregels in tabel 6.1 geldt het volgende. De laatste formule van een afleidingsregel, die vet is gedrukt, noemt men de *conclusie* en de overige formules de *premissen*. Voor A en B dient men daarbij formules uit $PROP$ te lezen. De premissen van een afleidingsregel geven aan welke formules in de afleiding aanwezig moeten zijn, wil men de conclusie behorende bij die afleidingsregel kunnen trekken. Net als bij de boommethode mogen afleidingsregels *nooit* op echte subformules van een formule worden toegepast! Bij de naamgeving van de regels staat *intro* voor *introductie*, *elim* voor *eliminatie*, en *rei* voor *reïteratie*.

$\vee$-**intro**		$\vee$-**elim**	$\neg$-**intro**	$\neg$-**elim**
A $\vdots$ $\boldsymbol{A \vee B}$	B $\vdots$ $\boldsymbol{A \vee B}$	$A \vee B$ $\vdots$ │ A │ $\vdots$ │ C $\vdots$ │ B │ $\vdots$ │ C $\vdots$ $\boldsymbol{C}$	│ A │ $\vdots$ │ B │ $\vdots$ │ $\neg B$ $\vdots$ $\boldsymbol{\neg A}$	$\neg\neg A$ $\vdots$ $\boldsymbol{A}$

$\rightarrow$-**intro**	$\rightarrow$-**elim**	$\wedge$-**intro**	$\wedge$-**elim**	
│ A │ $\vdots$ │ B $\vdots$ $\boldsymbol{A \rightarrow B}$	$A \rightarrow B$ $\vdots$ A $\vdots$ $\boldsymbol{B}$	A $\vdots$ B $\vdots$ $\boldsymbol{A \wedge B}$	$A \wedge B$ $\vdots$ $\boldsymbol{A}$	$A \wedge B$ $\vdots$ $\boldsymbol{B}$

$\leftrightarrow$-**intro**	$\leftrightarrow$-**elim**		**rei**
$A \rightarrow B$ $\vdots$ $B \rightarrow A$ $\vdots$ $\boldsymbol{A \leftrightarrow B}$	$A \leftrightarrow B$ $\vdots$ $\boldsymbol{A \rightarrow B}$	$A \leftrightarrow B$ $\vdots$ $\boldsymbol{B \rightarrow A}$	│ A │ $\vdots$ │ │ … │ $\vdots$ │ │ … │ $\boldsymbol{A}$

Tabel 6.1: Afleidingsregels van het systeem $\mathcal{F}$ van Fitch.

Een belangrijke eigenschap van de afleidingsregels is dat er voor ieder logisch connectief minstens één *introductieregel* en minstens één *eliminatieregel* bestaat. Met behulp van een introductieregel voor een bepaald connectief kan men formules afleiden die samengesteld zijn met dat connectief, terwijl met behulp van een eliminatieregel conclusies kunnen worden afgeleid uit formules die zijn samengesteld met dat connectief. Beeldend uitgedrukt, met eliminatieregels kan men *knippen* en met introductieregels *plakken*.

Voor alle afleidingsregels geldt, dat deze slechts kunnen worden toegepast als de premissen ervan op de 'goede' plaats in de bewijsfiguur voorkomen: alle premissen en/of hypothese-intervallen met premissen dienen samen met de conclusie in *hetzelfde* hypothese-interval te liggen. Alle premissen dienen altijd vooraf te gaan aan de conclusie. De reïteratieregel wordt gebruikt om formules die in een omvattend hypothese-interval liggen, op de goede plaats te krijgen, of, anders gezegd, te *importeren*.

De beide regels $\vee$-intro (introductie van de disjunctie) komen overeen met de intuïtie. Immers, als men A (of B) heeft afgeleid dan kan men tot $A \vee B$ besluiten. Dus $A \vee B$ is afleidbaar uit A en ook uit B.

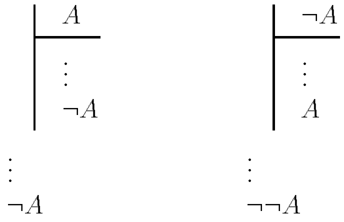
Hetzelfde geldt voor de regels $\wedge$-intro (introductie van de conjunctie) en $\wedge$-elim (eliminatie van de conjunctie). Als beide formules A en B afleidbaar zijn, dan is $A \wedge B$ afleidbaar en omgekeerd zijn uit $A \wedge B$ zowel A als B afleidbaar.

De regel $\vee$-elim (eliminatie van de disjunctie) drukt uit dat C afleidbaar is uit $A \vee B$, als C kan worden afgeleid uit de hypothese A en als C kan worden afgeleid uit de hypothese B.

De regel $\rightarrow$-intro (introductie van de implicatie) geeft aan dat men $A \rightarrow B$ kan afleiden als men uit de hypothese A de formule B kan afleiden. Bij de laatste stap wordt de regel $\rightarrow$-intro toegepast. Hierdoor wordt de hypothese A, waaruit B is afgeleid, *geëlimineerd* en concludeert men $A \rightarrow B$. Dit stemt overeen met het gebruik van de implicatie in de wiskunde.

De regel $\rightarrow$-elim (eliminatie van de implicatie), ook wel *modus ponens* genaamd, is de tegenhanger van de regel $\rightarrow$-intro. De regel drukt uit dat uit de premissen $A \rightarrow B$ en A de formule B kan worden afgeleid.

Een tweede regel waarmede men hypothesen kan elimineren, is de regel $\neg$-intro (introductie van de negatie). In de wiskunde staat de toepassing van deze regel bekend als *bewijs uit het ongerijmde* of *reductio ad absurdum*. Stel men wil aantonen dat $\neg A$ geldt. Men stelt in zo'n geval dat A geldt (de hypothese) en laat vervolgens zien dat men voor een zekere formule B zowel B als $\neg B$ kan afleiden, hetgeen een tegenspraak is. De hypothese A is dus blijkbaar onjuist, zodat men $\neg A$ concludeert. Op deze manier wordt de hypothese A geelimineerd. Deze afleidingsregel heeft twee bijzondere gevallen (zie figuur 6.2). Het eerste geval treedt op als men uit de aanname A de formule $\neg A$ kan afleiden.



Figuur 6.2: De twee speciale gevallen van afleidingsregel ¬-intro.

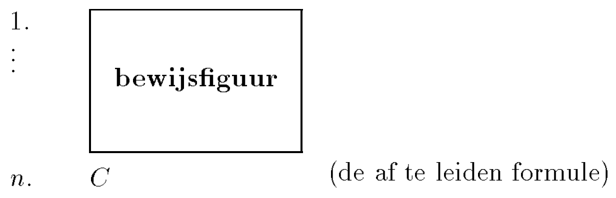
Dit betekent dat de aanname A onjuist is, zodat $\neg A$ moet gelden. Bij deze toepassing van ¬-intro vallen de formules A en B uit tabel 6.1 samen. In het tweede geval leidt men uit de aanname $\neg A$ de formule A af. Nu is de aanname $\neg A$ onjuist en leidt men $\neg\neg A$ af. Hier vallen de formules A en $\neg B$ samen.

De regel ¬-elim (eliminatie van de negatie) drukt uit dat men A mag afleiden uit *niet-niet* A. Deze regel ontleent zijn geldigheid aan het feit dat de klassieke logica slechts twee waarheidswaarden kent.

De regels ↔-intro (introductie van de equivalentie) en ↔-elim (eliminatie van de equivalentie) zijn voor de hand liggend, als men bedenkt dat in stelling 4.1.3 bewezen is dat $A \leftrightarrow B \approx (A \rightarrow B) \wedge (B \rightarrow A)$.

6.1.2 DEFINITIE **Toepassing van een afleidingsregel**
Zij gegeven een bewijsfiguur $\mathbf{D}$ *met interval* $D = [1, n]$, *formules* $F_1, \ldots, F_n$ *en hypothese-intervallen* $\mathbf{H}$. *Dan is formule* E *het resultaat van een* toepassing *van de afleidingsregel* R, *indien* E *de conclusie is van* R, *de premissen van* R *voorafgaan aan* E, *en indien voldaan is aan één van de volgende voorwaarden:*

1. $R \in \{\vee\text{-}intro, \neg\text{-}elim, \rightarrow\text{-}elim, \wedge\text{-}intro, \wedge\text{-}elim, \leftrightarrow\text{-}intro, \leftrightarrow\text{-}elim\}$.
   In dit geval moeten de premissen en de conclusie E *alle in* hetzelfde *interval liggen. De volgorde waarin de premissen voorkomen, mag daarbij afwijken van de volgorde zoals aangegeven in tabel 6.1.*
2. $R = \neg\text{-}intro$.
   Er moet een hypothese-interval $[k, l] \in \mathbf{H}$ *bestaan zodanig dat* $F_k = A$, *en zodanig dat ofwel* $F_l = \neg B$ *en* $B = F_m$ *voor zekere* $m \in [k, l]$, *ofwel* $F_l = B$ *en* $\neg B = F_m$ *voor zekere* $m \in [k, l]$. *De conclusie* $E = \neg A$ *en het interval* $[k, l]$ *dienen in hetzelfde interval te liggen. In beide gevallen is het toegestaan dat* $m = k$ *(*A *en* B *vallen samen, of* A *en* $\neg B$ *vallen samen).*
3. $R = \rightarrow\text{-}intro$.
   Er moet een hypothese-interval $[k, l] \in \mathbf{H}$ *bestaan zodanig dat* $F_k = A$ *en* $F_l = B$. *De conclusie* $E = A \rightarrow B$ *en het interval* $[k, l]$ *dienen in het zelfde interval te liggen.*
4. $R = \vee\text{-}elim$.
   Er moeten hypothese-intervallen $[i, j], [k, l] \in \mathbf{H}$ *bestaan zodanig dat* $F_i = A$, $F_j = C$, $F_k = B$ *en* $F_l = C$, *en zodanig dat ofwel* $j < k$, *ofwel* $l < i$. *De conclusie* $E = C$, *de hypothese* $A \vee B$ *en de intervallen* $[i, j]$ *en* $[k, l]$ *dienen in hetzelfde interval te liggen.*
5. $R =$ *rei.*
   Als de hypothese A *in het interval* $I \in \mathbf{H} \cup \{D\}$ *ligt en de conclusie* $E = A$ *in het interval* $J \in \mathbf{H} \cup \{D\}$, *dan moet gelden* $J \subseteq I$.



Figuur 6.3: Schema van een afleiding zonder hypothesen.

Nu gedefinieerd is wat de afleidingsregels zijn en hoe deze moeten worden toegepast, nemen we de begrippen *afleiding* en *afleidbaarheid* onder de loep.

6.1.3 Definitie **Afleiding zonder hypothesen**
Een afleiding van de formule C *is een bewijsfiguur* $\mathbf{D}$ *met interval* $D = [1, n]$ *en formules* $F_1, \ldots, F_n$ *dat aan de volgende voorwaarden voldoet:*

1. $F_n = C$ *en* $gr(n) = 0$*;*
2. *iedere formule* F_i $(1 \leq i \leq n)$ *is een hypothese of het resultaat van de toepassing van een afleidingsregel op een aantal aan* F_i *voorafgaande formules.*

Onder de bewijslengte *van* C *met betrekking tot* $\mathbf{D}$ *verstaat men* $len(\mathbf{D})$.

In figuur 6.3 wordt geïllustreerd hoe een afleiding zonder hypothesen eruit ziet. Een concreet voorbeeld van een afleiding zonder hypothesen treft men aan in figuur 6.1.

6.1.4 Definitie **Afleiding met hypothesen**
Een afleiding van de formule C uit de formules $P_1, \ldots, P_m$ $(m \geq 1)$ *is een bewijsfiguur* $\mathbf{D}$ *met interval* $D = [1, n]$ $(n > m)$ *en formules* $F_1, \ldots, F_n$ *waarvoor geldt:*

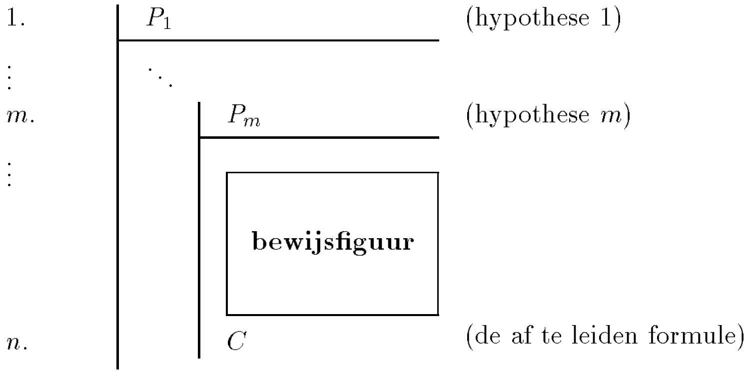
1. $F_i = P_i$ *is een hypothese voor* $1 \leq i \leq m$*, zodat* $gr(i) = i$*;*
2. $F_n = C$*, en* C *en* P_m *liggen in hetzelfde hypothese-interval, zodat* $gr(n) = m$*;*
3. *iedere formule* F_i $(1 \leq i \leq n)$ *is een hypothese of het resultaat van de toepassing van een afleidingsregel op een aantal aan* F_i *voorafgaande formules.*

Onder de bewijslengte *van* C *uit* $P_1, \ldots, P_m$ *met betrekking tot* $\mathbf{D}$ *verstaat men* $len(\mathbf{D})$.

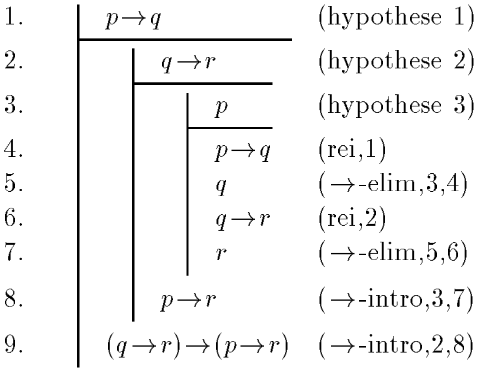
In figuur 6.4 treft men een schema aan voor een afleiding van de formule C uit de hypothesen $P_1, \ldots, P_m$. Een concreet voorbeeld van een afleiding van een formule uit één hypothese vindt men in figuur 6.5.

6.1.5 Definitie **Afleidbaarheid**

1. *Een formule* C *is* afleidbaar *als er een afleiding van* C *bestaat. Dit wordt genoteerd als:* $\vdash C$.
2. *Een formule* C *is* afleidbaar uit de formules $P_1, \ldots, P_m$ *als er een afleiding van* C *uit* $P_1, \ldots, P_m$ *bestaat. Dit wordt genoteerd als:* $P_1, \ldots, P_m \vdash C$.



Figuur 6.4: Schema van een afleiding met hypothesen.



1.	$p \to q$	(hypothese 1)
2.	$q \to r$	(hypothese 2)
3.	p	(hypothese 3)
4.	$p \to q$	(rei,1)
5.	q	($\to$-elim,3,4)
6.	$q \to r$	(rei,2)
7.	r	($\to$-elim,5,6)
8.	$p \to r$	($\to$-intro,3,7)
9.	$(q \to r) \to (p \to r)$	($\to$-intro,2,8)

Figuur 6.5: Afleiding van $(q \to r) \to (p \to r)$ uit $p \to q$.

3. *Zij* $\Gamma \subseteq PROP$ *een verzameling formules. Een formule* C *is* afleidbaar uit Γ *als er* m *formules* $P_1, \ldots, P_m \in \Gamma$ *bestaan zodanig dat* $P_1, \ldots, P_m \vdash C$. *Dit wordt genoteerd als:* $\Gamma \vdash C$. *Als* $\Gamma = \emptyset$, *dan schrijft men:* $\vdash C$.

De lezer zal zich wellicht afvragen waarom we ons zoveel moeite hebben getroost om de begrippen uit deze paragraaf zo precies te definiëren. De reden is dat we hiervan profijt zullen hebben als we eigenschappen over het systeem van Fitch willen bewijzen, zoals in hoofdstuk 7.

6.2 Afleidingen en afleidingsstrategieën

In de afleidingen die we in deze paragraaf zullen geven, zullen we geen gebruik maken van propositiesymbolen, maar van metavariabelen voor formules. De afleiding van een specifieke formule verkrijgt men dan door geschikte substituties te doen voor deze metavariabelen. Vervangt men bijvoorbeeld in de afleiding van formule 1 in het volgende voorbeeld de metavariabelen A en B door de propositiesymbolen p en q, dan verkrijgt men een afleiding van de formule $p \to (q \to p)$. Vervangt men echter in deze afleiding A en B door $p \wedge q$ en $p \to q$, dan verkrijgt men een afleiding van de formule $[p \wedge q] \to [(p \to q) \to (p \wedge q)]$.

6.2.1 VOORBEELD Zij $A, B, C \in PROP$. Dan zijn de onderstaande formules afleidbaar in het systeem $\mathcal{F}$.

1. $A \to (B \to A)$.
2. $(A \to B) \to [(B \to C) \to (A \to C)]$.
3. $(A \to B) \to (\neg B \to \neg A)$ *(contrapositie).*
4. $(\neg B \to \neg A) \to (A \to B)$.
5. $A \vee \neg A$ *(tertium non datur).*
6. $(A \vee (B \vee C)) \to ((A \vee B) \vee C)$ (distributiviteit).

Van de formules 1, 3, 4, 5 en 6 wordt een afleiding gegeven in de figuren 6.6, 6.7, 6.8, 6.9 en 6.10. In figuur 6.5 staat een afleiding van formule 2 voor het geval dat $A = p$, $B = q$ en $C = r$. ∎

Uit voorbeelden van afleidingen blijkt dat men voor het vinden van een afleiding vaak gebruik kan maken van een aantal vuistregels. Neem aan dat we een formule $F \in PROP$ moeten bewijzen. We kunnen dan afhankelijk van de vorm van F als volgt te werk gaan.

1.	│ A	(hypothese 1)
2.	│ │ B	(hypothese 2)
3.	│ │ A	(rei,1)
4.	│ $B\to A$	($\to$-intro,2,3)
5.	$A\to(B\to A)$	($\to$-intro,1,4)

Figuur 6.6: Bewijs van $\vdash A\to(B\to A)$.

1.	│ $A\to B$	(hypothese 1)
2.	│ │ $\neg B$	(hypothese 2)
3.	│ │ │ A	(hypothese 3)
4.	│ │ │ $A\to B$	(rei,1)
5.	│ │ │ B	($\to$-elim,3,4)
6.	│ │ │ $\neg B$	(rei,2)
7.	│ │ $\neg A$	($\neg$-intro,3,5,6)
8.	│ $\neg B\to\neg A$	($\to$-intro,2,7)
9.	$(A\to B)\to(\neg B\to\neg A)$	($\to$-intro,1,8)

Figuur 6.7: Bewijs van $\vdash (A\to B)\to(\neg B\to\neg A)$.

1.	│ $\neg B\to\neg A$	(hypothese 1)
2.	│ │ A	(hypothese 2)
3.	│ │ │ $\neg B$	(hypothese 3)
4.	│ │ │ A	(rei,2)
5.	│ │ │ $\neg B\to\neg A$	(rei,1)
6.	│ │ │ $\neg A$	($\to$-elim,3,5)
7.	│ │ $\neg\neg B$	($\neg$-intro,3,4,6)
8.	│ │ B	($\neg$-elim,7)
9.	│ $A\to B$	($\to$-intro,2,8)
10.	$(\neg B\to\neg A)\to(A\to B)$	($\to$-intro,1,9)

Figuur 6.8: Bewijs van $\vdash (\neg B\to\neg A)\to(A\to B)$.

1.	$\quad \neg(A \vee \neg A)$	(hypothese 1)
2.	$\quad\quad A$	(hypothese 2)
3.	$\quad\quad A \vee \neg A$	($\vee$-intro,2)
4.	$\quad\quad \neg(A \vee \neg A)$	(rei,1)
5.	$\quad \neg A$	($\neg$-intro,2,3,4)
6.	$\quad A \vee \neg A$	($\vee$-intro,5)
7.	$\neg\neg(A \vee \neg A)$	($\neg$-intro,1,6)
8.	$A \vee \neg A$	($\neg$-elim,7)

Figuur 6.9: Bewijs van $\vdash A \vee \neg A$.

1.	$\quad A \vee (B \vee C)$	(hypothese 1)
2.	$\quad\quad A$	(hypothese 2)
3.	$\quad\quad A \vee B$	($\vee$-intro,2)
4.	$\quad\quad (A \vee B) \vee C$	($\vee$-intro,3)
5.	$\quad\quad B \vee C$	(hypothese 3)
6.	$\quad\quad\quad B$	(hypothese 4)
7.	$\quad\quad\quad A \vee B$	($\vee$-intro,6)
8.	$\quad\quad\quad (A \vee B) \vee C$	($\vee$-intro,7)
9.	$\quad\quad\quad C$	(hypothese 5)
10.	$\quad\quad\quad (A \vee B) \vee C$	($\vee$-intro,9)
11.	$\quad\quad (A \vee B) \vee C$	($\vee$-elim,5,8,10)
12.	$\quad (A \vee B) \vee C$	($\vee$-elim,1,4,11)
13.	$(A \vee (B \vee C)) \rightarrow ((A \vee B) \vee C)$	($\rightarrow$-intro,1,12)

Figuur 6.10: Bewijs van $(A \vee (B \vee C)) \rightarrow ((A \vee B) \vee C)$.

1. $F = (A \wedge B)$.
   Probeer afzonderlijk A en B af te leiden. Hieruit volgt door toepassing van de regel $\wedge$-intro $(A \wedge B)$.

2. $F = (A \rightarrow B)$.
   Open een nieuw hypothese-interval met A als hypothese en tracht hierin B af te leiden. Door toepassing van de regel $\rightarrow$-intro volgt hieruit $(A \rightarrow B)$.

3. $F = \neg A$.
   Open een nieuw hypothese-interval met A als hypothese en probeer hierin een contradictie van de vorm C en $\neg C$ af te leiden. Door toepassing van de regel $\neg$-intro verkrijgt men als resultaat $\neg A$.

4. $F = (A \vee B)$.
   Probeer F uit het ongerijmde te bewijzen, dat wil zeggen open een hypothese-interval met $\neg(A \vee B)$ als hypothese en probeer hieruit een contradictie van de vorm C en $\neg C$ af te leiden. Door toepassing van de regel $\neg$-intro verkrijgt men dan $\neg\neg(A \vee B)$. Hieruit leidt men dan met behulp van $\neg$-elim $(A \vee B)$ af. In sommige gevallen kan F worden bewezen met $\vee$-intro. Dit geval doet zich bijvoorbeeld voor indien één van de aannameformules een disjunctie van de vorm $C \vee D$ is. Men kan dan proberen op $C \vee D$ de regel $\vee$-elim toe te passen en vervolgens in de twee hypothese-intervallen met C en D als hypothesen de formule F af leiden met behulp van $\vee$-intro. Deze strategie is gebruikt in de afleiding in figuur 6.10.

5. $F = (A \leftrightarrow B)$.
   Tracht de formules $A \rightarrow B$ en $B \rightarrow A$ af te leiden. Met behulp van de regel $\leftrightarrow$-intro volgt hieruit $(A \leftrightarrow B)$.

6. Bij de bovenstaande gevallen 3 en 4 is er sprake van het vinden van een contradictie van de vorm C en $\neg C$. Hoe vindt men die formule C? Hier dient enige creativiteit aan de dag te worden gelegd. Vaak zal een geschikte C een *subformule* van F zijn (bijvoorbeeld F zelf).

7. Als het met bovenstaande vuistregels niet lukt om een afleiding van formule F te vinden, kan men proberen om F uit het ongerijmde af te leiden, zoals in geval 4 hierboven.

8. Het kan voorkomen dat men een aannameformule niet kan gebruiken omdat geen eliminatieregel erop toepasbaar is. In zo'n geval dient men een nieuw hypothese-interval te openen met een geschikte subformule van die aannameformule als hypothese. Stel men heeft als aannameformule

$A \rightarrow B$, dan zou men in dat geval een interval openen met A als hypothese, zodat B kan worden afgeleid met de regel $\rightarrow$-elim (dit is gedaan in stap 3 van de afleiding in figuur 6.7). Bij een aannameformule van de vorm $\neg(A \vee B)$ is het soms slim om een hypothese-interval met A of B als hypothese te openen (zie stap 2 van de afleiding in figuur 6.9).

We besluiten deze paragraaf met het bewijzen van een aantal elementaire eigenschappen van logische connectieven.

6.2.2 STELLING *Zij $A, B \in PROP$. Dan geldt:*

1. $\vdash \neg(A \wedge B) \rightarrow (\neg A \vee \neg B)$.

2. $\vdash (\neg A \vee \neg B) \rightarrow \neg(A \wedge B)$.

3. $\vdash \neg(A \vee B) \rightarrow (\neg A \wedge \neg B)$.

4. $\vdash (\neg A \wedge \neg B) \rightarrow \neg(A \vee B)$.

5. $\vdash \neg(A \rightarrow B) \rightarrow (A \wedge \neg B)$.

6. $\vdash (A \wedge \neg B) \rightarrow \neg(A \rightarrow B)$.

7. $\vdash (A \rightarrow B) \rightarrow (\neg A \vee B)$.

8. $\vdash (\neg A \vee B) \rightarrow (A \rightarrow B)$.

9. $\vdash A \rightarrow \neg\neg A$.

10. $\vdash \neg\neg A \rightarrow A$.

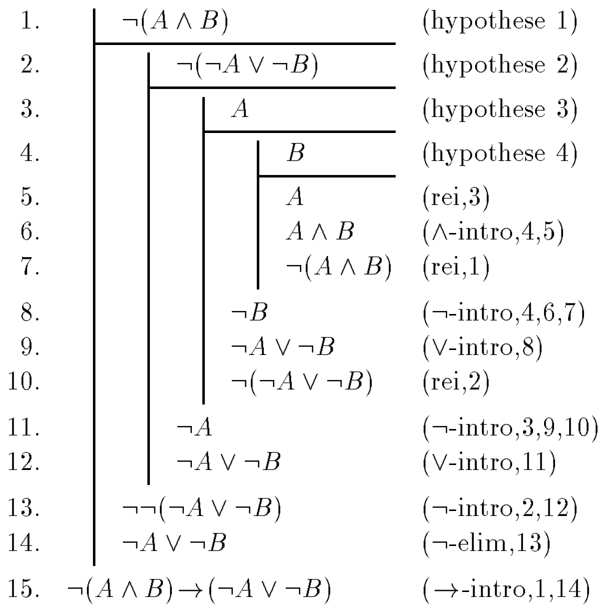
11. $\vdash (A \leftrightarrow B) \rightarrow [(A \rightarrow B) \wedge (B \rightarrow A)]$.

12. $\vdash [(A \rightarrow B) \wedge (B \rightarrow A)] \rightarrow (A \leftrightarrow B)$.

13. $\vdash \neg(A \leftrightarrow B) \rightarrow [\neg(A \rightarrow B) \vee \neg(B \rightarrow A)]$.

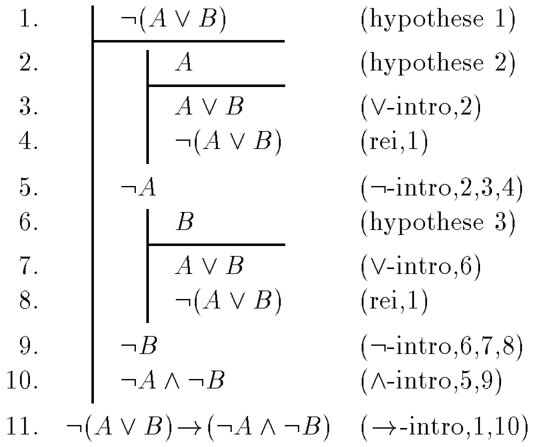
14. $\vdash [\neg(A \rightarrow B) \vee \neg(B \rightarrow A)] \rightarrow \neg(A \leftrightarrow B)$.

BEWIJS Van de formules 1, 3 en 8 wordt een afleiding gegeven in de figuren 6.11, 6.12 en 6.13. De overige gevallen worden aan de lezer overgelaten. ∎



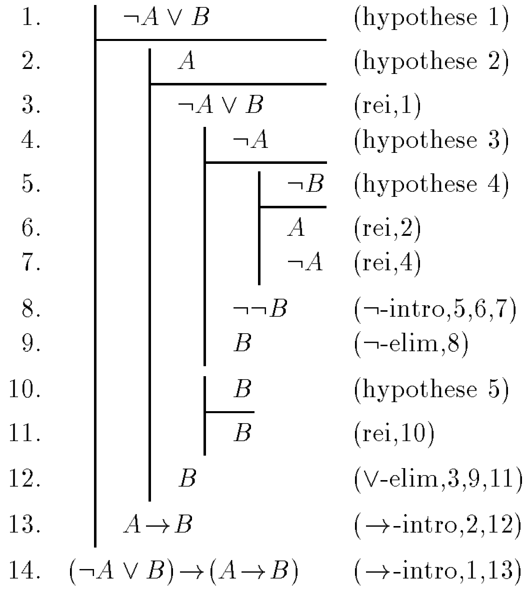
1.	$\neg(A \wedge B)$	(hypothese 1)
2.	$\neg(\neg A \vee \neg B)$	(hypothese 2)
3.	A	(hypothese 3)
4.	B	(hypothese 4)
5.	A	(rei,3)
6.	$A \wedge B$	($\wedge$-intro,4,5)
7.	$\neg(A \wedge B)$	(rei,1)
8.	$\neg B$	($\neg$-intro,4,6,7)
9.	$\neg A \vee \neg B$	($\vee$-intro,8)
10.	$\neg(\neg A \vee \neg B)$	(rei,2)
11.	$\neg A$	($\neg$-intro,3,9,10)
12.	$\neg A \vee \neg B$	($\vee$-intro,11)
13.	$\neg\neg(\neg A \vee \neg B)$	($\neg$-intro,2,12)
14.	$\neg A \vee \neg B$	($\neg$-elim,13)
15.	$\neg(A \wedge B) \rightarrow (\neg A \vee \neg B)$	($\rightarrow$-intro,1,14)

Figuur 6.11: Bewijs van $\vdash \neg(A \wedge B) \rightarrow (\neg A \vee \neg B)$.



1.	$\neg(A \vee B)$	(hypothese 1)
2.	A	(hypothese 2)
3.	$A \vee B$	($\vee$-intro,2)
4.	$\neg(A \vee B)$	(rei,1)
5.	$\neg A$	($\neg$-intro,2,3,4)
6.	B	(hypothese 3)
7.	$A \vee B$	($\vee$-intro,6)
8.	$\neg(A \vee B)$	(rei,1)
9.	$\neg B$	($\neg$-intro,6,7,8)
10.	$\neg A \wedge \neg B$	($\wedge$-intro,5,9)
11.	$\neg(A \vee B) \rightarrow (\neg A \wedge \neg B)$	($\rightarrow$-intro,1,10)

Figuur 6.12: Bewijs van $\vdash \neg(A \vee B) \rightarrow (\neg A \wedge \neg B)$.



1.	$\neg A \vee B$	(hypothese 1)
2.	A	(hypothese 2)
3.	$\neg A \vee B$	(rei,1)
4.	$\neg A$	(hypothese 3)
5.	$\neg B$	(hypothese 4)
6.	A	(rei,2)
7.	$\neg A$	(rei,4)
8.	$\neg\neg B$	($\neg$-intro,5,6,7)
9.	B	($\neg$-elim,8)
10.	B	(hypothese 5)
11.	B	(rei,10)
12.	B	($\vee$-elim,3,9,11)
13.	$A \rightarrow B$	($\rightarrow$-intro,2,12)
14.	$(\neg A \vee B) \rightarrow (A \rightarrow B)$	($\rightarrow$-intro,1,13)

Figuur 6.13: Bewijs van $\vdash (\neg A \vee B) \rightarrow (A \rightarrow B)$.

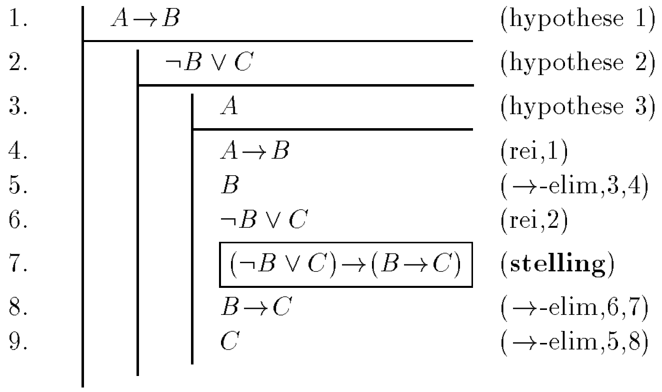
6.3 Het uitgebreide systeem van Fitch

Het systeem $\mathcal{F}$ kan op verschillende manieren gewijzigd worden, bijvoorbeeld door extra afleidingsregels aan $\mathcal{F}$ toe te voegen, door verandering van de voorwaarden waaronder de afleidingsregels in $\mathcal{F}$ toegepast mogen worden of door het aantal afleidingsregels te verminderen. Op deze manier kan men allerlei varianten van het natuurlijke deductiesysteem $\mathcal{F}$ verkrijgen. In deze paragraaf bespreken we een aantal uitbreidingen die het deductiesysteem handiger in het gebruik maken.

In de wiskunde maakt men in een bewijs doorgaans gebruik van reeds eerder bewezen stellingen. In het systeem $\mathcal{F}$, zoals beschreven in dit hoofdstuk, kan men in een afleiding geen gebruik maken van reeds afgeleide formules. Om dit te ondervangen voeren we een nieuwe notatie in. In afleidingen zullen we omkaderde formules

$$\boxed{F}$$

toelaten. Voor zo'n omkaderde formule dient men dan de *afleiding* van die formule te lezen. Door het gebruik van deze notatie wordt een afleiding vaak veel overzichtelijker. In informatica-terminologie: een afleiding wordt op deze wijze gestructureerd in 'modulen'.



Figuur 6.14: Gebruik van een stelling in een afleiding van $A\rightarrow B, \neg B \vee C, A \vdash C$.

Stel men wil bewijzen $A\rightarrow B, \neg B \vee C, A \vdash C$. In de bewijsfiguur in figuur 6.14 maken we gebruik van de nieuwe notatie. De afleiding van de omkaderde formule staat in figuur 6.13. De bewijsfiguur is uiteraard geen afleiding van $A\rightarrow B, \neg B \vee C, A \vdash C$ in het systeem $\mathcal{F}$. Als men echter de rechthoek

$$\boxed{(\neg B \vee C)\rightarrow(B\rightarrow C)}$$

vervangt door een afleiding van de formule $(\neg B \vee C)\rightarrow(B\rightarrow C)$, dan verkrijgt men een afleiding van C uit de hypothesen $A\rightarrow B$, $\neg B \vee C$ en A in het systeem $\mathcal{F}$. Dit wordt het *expanderen* van de bewijsfiguur genoemd. De expansie van de bewijsfiguur in figuur 6.14 kan men aantreffen in figuur 6.15.

Een andere manier om het vinden van afleidingen te vereenvoudigen is het invoeren van nieuwe, 'krachtige' afleidingsregels. Deze nieuwe afleidingsregels zijn dan gebaseerd op reeds afgeleide stellingen. Zo wordt bijvoorbeeld in het afleidingsfragment in figuur 6.16 de formule $\neg A \wedge \neg B$ afgeleid uit de formule $\neg(A \vee B)$ met behulp van de stelling $\neg(A \vee B)\rightarrow(\neg A \wedge \neg B)$. Aangezien de formule $\neg(A \vee B)\rightarrow(\neg A \wedge \neg B)$ afleidbaar is, kan men deze bewijsfiguur expanderen tot een afleiding. Men kan echter ook het systeem $\mathcal{F}$ uitbreiden met een nieuwe afleidingsregel:

$\vee$-regel

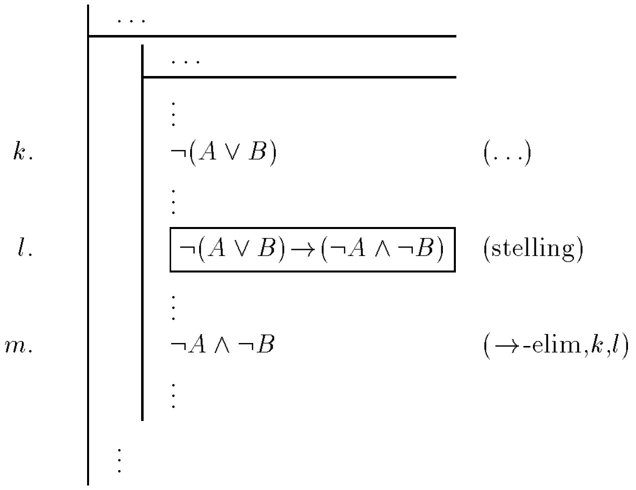
$\neg(A \vee B)$

$\vdots$

$\boldsymbol{\neg A \wedge \neg B}$

1.	$A \rightarrow B$	(hypothese 1)
2.	$\neg B \vee C$	(hypothese 2)
3.	A	(hypothese 3)
4.	$A \rightarrow B$	(rei,1)
5.	B	($\rightarrow$-elim,3,4)
6.	$\neg B \vee C$	(rei,2)
7.	$\neg B \vee C$	(hypothese 4)
8.	B	(hypothese 5)
9.	$\neg B \vee C$	(rei,7)
10.	$\neg B$	(hypothese 6)
11.	$\neg C$	(hypothese 7)
12.	B	(rei,8)
13.	$\neg B$	(rei,10)
14.	$\neg\neg C$	($\neg$-intro,11,12,13)
15.	C	($\neg$-elim,14)
16.	C	(hypothese 8)
17.	C	(rei,16)
18.	C	($\vee$-elim,9,15,17)
19.	$B \rightarrow C$	($\rightarrow$-intro,8,18)
20.	$(\neg B \vee C) \rightarrow (B \rightarrow C)$	($\rightarrow$-intro,7,19)
21.	$B \rightarrow C$	($\rightarrow$-elim,6,20)
22.	C	($\rightarrow$-elim,5,21)

Figuur 6.15: De expansie van de bewijsfiguur in figuur 6.14.

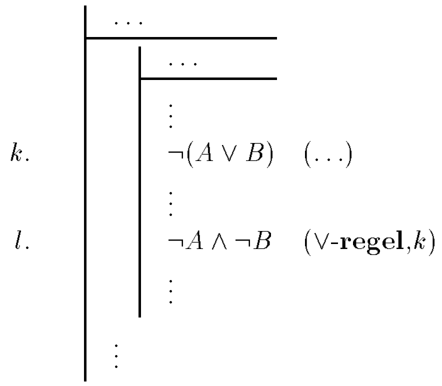


Figuur 6.16: Afleidingsfragment waarin een stelling wordt gebruikt.

In dit geval kan in de afleiding het gebruik van de stelling $\neg(A \vee B) \rightarrow (\neg A \wedge \neg B)$ vervangen worden door een toepassing van $\vee$-regel op de formule $\neg(A \vee B)$. Het afleidingsfragment uit figuur 6.16 kan dan vereenvoudigd worden tot het fragment in figuur 6.17.

Uit het voorafgaande volgt, dat iedere toepassing van de $\vee$-regel in een afleiding vervangen kan worden door het gebruik van de afleidbare formule $\neg(A \vee B) \rightarrow (\neg A \wedge \neg B)$ als stelling. Door deze stelling (omkaderde formule) dan te expanderen tot een afleiding verkrijgt men een afleiding in systeem $\mathcal{F}$. Dus iedere formule F die met behulp van de $\vee$-regel is afgeleid, kan ook afgeleid worden zonder deze regel. Het is duidelijk dat men op deze wijze het systeem $\mathcal{F}$ met een willekeurig aantal nieuwe afleidingsregels kan uitbreiden. Het voordeel van het gebruik van deze regels is dat men over het algemeen kortere bewijzen krijgt. Bovendien zal de gezochte afleiding vaak sneller worden gevonden.

Een aantal van deze 'nieuwe' afleidingsregels is bij elkaar gezet in tabel 6.2. De laatste, vet gedrukte formule van een afleidingsregel in deze tabel is de conclusie en de overige zijn de premissen. Voor de toepassing van deze regels gelden dezelfde voorwaarden als voor de regels $\vee$-intro, $\neg$-elim etc. (zie definitie 6.1.2 clausule 1). Dit betekent dat de premissen vooraf moeten gaan aan en in hetzelfde interval moeten liggen als de conclusie. Bij toepassing van de regel 'contra' mag de volgorde van de premissen afwijken van die in tabel 6.2. Een voorbeeld van een afleiding waarin gebruik wordt gemaakt van de nieuwe afleidingsregels treft men aan in figuur 6.18.



Figuur 6.17: Afleidingsfragment waarin een nieuwe afleidingsregel wordt gebruikt.

$\wedge$**-regel**	$\vee$**-regel**	**contra**
$\neg(A \wedge B)$ $\vdots$ $\boldsymbol{\neg A \vee \neg B}$	$\neg(A \vee B)$ $\vdots$ $\boldsymbol{\neg A \wedge \neg B}$	A $\vdots$ $\neg A$ $\vdots$ $\boldsymbol{B}$

$\rightarrow$**-regel1**	$\rightarrow$**-regel2**	$\leftrightarrow$**-regel**
$\neg(A \rightarrow B)$ $\vdots$ $\boldsymbol{A \wedge \neg B}$	$A \rightarrow B$ $\vdots$ $\boldsymbol{\neg A \vee B}$	$\neg(A \leftrightarrow B)$ $\vdots$ $\boldsymbol{\neg(A \rightarrow B) \vee \neg(B \rightarrow A)}$

Tabel 6.2: Afleidingsregels van het systeem $\mathcal{F}_{uit}$.

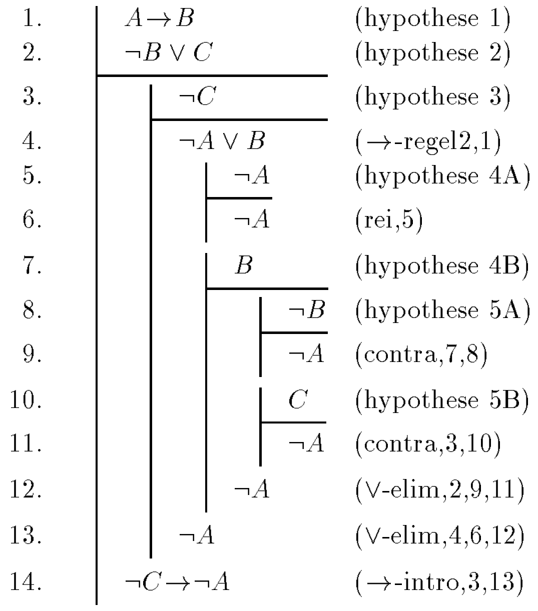
1.	$\neg[(A\rightarrow B)\vee(B\rightarrow A)]$	(hypothese 1)
2.	$\neg(A\rightarrow B)\wedge\neg(B\rightarrow A)$	($\wedge$-regel,1)
3.	$\neg(A\rightarrow B)$	($\wedge$-elim,2)
4.	$\neg(B\rightarrow A)$	($\wedge$-elim,2)
5.	$A\wedge\neg B$	($\rightarrow$-regel1,3)
6.	$B\wedge\neg A$	($\rightarrow$-regel1,4)
7.	A	($\wedge$-elim,5)
8.	$\neg A$	($\wedge$-elim,6)
9.	$\neg\neg[(A\rightarrow B)\vee(B\rightarrow A)]$	($\neg$-intro,1,7,8)
10.	$(A\rightarrow B)\vee(B\rightarrow A)$	($\neg$-elim,9)

Figuur 6.18: Bewijs van $\vdash_{\mathcal{F}_{uit}} (A\rightarrow B)\vee(B\rightarrow A)$.

Tot nog toe hebben we tamelijk strenge voorwaarden gesteld aan het toepassen van de afleidingsregels: premissen en conclusies moesten aan allerlei eisen voldoen met betrekking tot hun graad en hypothese-interval. Als we bedenken dat de reïteratieregel van het systeem $\mathcal{F}$ alleen gebruikt wordt om premissen in de 'goede' hypothese-intervallen te plaatsen, zodat de afleidingsregels toegepast mogen worden, dan ligt hier een mogelijkheid tot een nieuwe uitbreiding. Het effect van de reïteratieregel is dat formules gecopieerd worden. Als we nu zouden toestaan dat de reïteratieregel niet *expliciet* behoeft te worden toegepast, dan zouden we dit copiëren kunnen vermijden, zodat de bewijslengte afneemt. We kunnen het systeem $\mathcal{F}$ dus op deze wijze uitbreiden met de volgende afspraak:

De afleidingsregels uit de tabellen 6.1 en 6.2 mogen worden toegepast, als het in principe mogelijk is om de premissen door middel van de reïteratieregel in de goede hypothese-intervallen te krijgen. Het is dan niet verplicht om de reïteratieregel daadwerkelijk toe te passen.

Een laatste uitbreiding, die van puur notationele aard is, heeft betrekking op afleidingen van formules uit aannameformules. We zullen niet meer eisen dat er voor iedere aannameformule een nieuw hypothese-interval wordt geopend. In plaats daarvan worden alle aannameformules onder elkaar boven de verticale streep van het eerste hypothese-interval geplaatst. Het gevolg hiervan is dat de aannameformules en de conclusie in hetzelfde hypothese-interval liggen. Dit wordt geïllustreerd in figuur 6.19. We wijzen erop dat deze notationele conventie beschouwd wordt als een *verkorte schrijfwijze*. Als we in formele zin over de hypothese-intervallen van een afleiding spreken waarin deze



Figuur 6.19: Bewijs van $A \rightarrow B, \neg B \vee C \vdash_{\mathcal{F}_{uit}} \neg C \rightarrow \neg A$.

verkorte schrijfwijze is toegepast, dan zullen we aannemen dat er voor iedere aannameformule een nieuw hypothese-interval is geopend.

In figuur 6.19 wordt ook de omkaderde afspraak betreffende de reïteratieregel toegepast. Zo staat bijvoorbeeld de formule $\neg B \vee C$ (regelnummer 2) voor de toepassing van de regel $\vee$-elim (regelnummer 12) eigenlijk in het verkeerde hypothese-interval. Het is echter eenvoudig in te zien dat door een eventuele toepassing van de reïteratieregel de genoemde formule in het juiste hypothese-interval, in dit geval $[7, 12]$, gezet kan worden.

Als we nu aan systeem $\mathcal{F}$ alle uitbreidingen die we tot nu toe hebben bekeken, toevoegen dan verkrijgen we het systeem $\mathcal{F}_{uit}$, het *uitgebreide systeem van Fitch*. Nog eens op een rijtje gezet, bevat het systeem $\mathcal{F}_{uit}$ de volgende uitbreidingen:

1. het gebruik van stellingen,
2. de afleidingsregels uit figuur 6.2,
3. de afspraak betreffende het achterwege laten van toepassingen van de reïteratieregel, en
4. de notationele uitbreiding betreffende afleidingen uit aannameformules.

Uit het voorafgaande blijkt dat de systemen $\mathcal{F}$ en $\mathcal{F}_{uit}$ equivalent zijn. Dit wil zeggen dat iedere formule F die in $\mathcal{F}_{uit}$ afleidbaar is uit een verzameling Γ van formules, ook in $\mathcal{F}$ afleidbaar is uit Γ, en omgekeerd. Dit resultaat is samengevat in de volgende stelling.

6.3.1 STELLING **Equivalentie van $\mathcal{F}$ en $\mathcal{F}_{uit}$**
Zij $\Gamma \subseteq PROP$ *en* $F \in PROP$. *Dan geldt:*

$$\Gamma \vdash_{\mathcal{F}} F \quad \Leftrightarrow \quad \Gamma \vdash_{\mathcal{F}_{uit}} F.$$

In de formulering van de bovenstaande stelling betekent $\vdash_{\mathcal{F}}$: *afleidbaar in systeem* $\mathcal{F}$. Een analogon geldt voor $\vdash_{\mathcal{F}_{uit}}$.

6.4 Opgaven

1. Zij $A, B, C \in PROP$. Bewijs in het niet-uitgebreide systeem van Fitch $\mathcal{F}$:
   (a) $\vdash (A \to (B \to C)) \to (B \to (A \to C))$.
   (b) $\vdash (B \to C) \to ((A \vee B) \to (A \vee C))$.
   (c) $\vdash A \vee (A \to B)$.
   (d) $\vdash ((A \vee (A \to B)) \to B) \to B$.
   (e) $\vdash (A \to B) \vee (B \to A)$.
2. Zij $A, B, C \in PROP$. Bewijs in het uitgebreide systeem van Fitch $\mathcal{F}_{uit}$:
   (a) $A \leftrightarrow B, (\neg A \wedge \neg B) \to \neg C, C \vdash B$.
   (b) $(A \wedge B) \to \neg D, \neg(D \to E), B \vee E \vdash \neg A$.
   (c) $A \to (B \vee (C \to D)), B \to (B \to \neg A), \neg(C \to \neg A) \vdash D$.
   (d) $\vdash ((A \to B) \to (A \to C)) \to (A \to (B \to C))$.
   (e) $\vdash ((A \to B) \to A) \to A$.
   (f) $\vdash (A \to (B \vee C)) \to ((A \to B) \vee (A \to C))$.
   (g) $\vdash A \to (B \to (A \leftrightarrow B))$.
   (h) $\vdash (A \vee (B \to C)) \to ((A \vee B) \to (A \vee C))$.
   (i) $\vdash ((A \vee B) \to (A \vee C)) \to (A \vee (B \to C))$.
3. Bewijs zelf de overige gevallen van stelling 6.2.2.

Hoofdstuk 7

Correctheid en Volledigheid van de Propositielogica

Tot nu hebben we de propositielogica semantisch en syntactisch beschouwd. In de semantiek speelde het begrip *logisch gevolg*, gesymboliseerd door de relatie $\models$, een belangrijke rol, terwijl het in de syntaxis voornamelijk draaide om het begrip *afleidbaarheid*, weergegeven door de relatie $\vdash$. In dit hoofdstuk zullen we bewijzen dat de relaties $\models$ en $\vdash$ samenvallen. Dit wil zeggen dat voor alle verzamelingen $\Gamma \subseteq PROP$ en formules $F \in PROP$ geldt dat:

$$\Gamma \models F \quad \Leftrightarrow \quad \Gamma \vdash F.$$

Men zegt ook wel dat de propositielogica *correct* en *volledig* is. De correctheid van de propositielogica, ofwel

$$\Gamma \vdash F \quad \Rightarrow \quad \Gamma \models F,$$

drukt uit dat het begrip afleidbaarheid correct is geformaliseerd door het natuurlijke deductiesysteem van Fitch ($\mathcal{F}$ of $\mathcal{F}_{uit}$). In dit systeem is het dus niet mogelijk om semantisch onjuiste beweringen af te leiden. De volledigheid ofwel

$$\Gamma \models F \quad \Rightarrow \quad \Gamma \vdash F,$$

brengt tot uitdrukking dat het systeem van Fitch krachtig genoeg is om alle semantisch juiste beweringen af te leiden.

Als Γ een eindige verzameling is, dan kan men de volledigheid en correctheid ook als volgt formuleren: een boom voor $\Gamma \cup \{\neg F\}$ geconstrueerd volgens de boommethode sluit af dan en slechts dan als er een afleiding bestaat voor $\Gamma \vdash F$ volgens de methode van Fitch.

Een andere belangrijke eigenschap van de propositielogica is het feit dat de relatie $\models$ *beslisbaar* is. Hiermee wordt bedoeld dat er een methode bestaat om in een eindig aantal stappen te bepalen of $\Gamma \models F$ voor willekeurige formules F

en eindige verzamelingen Γ van formules. Deze methode bestaat uit het construeren van een boom voor de verzameling $\Gamma \cup \{\neg F\}$ (zie eind paragraaf 5.1). Uit het feit dat de relaties $\models$ en $\vdash$ samenvallen, volgt dat $\vdash$ ook beslisbaar is. We zullen hierop niet verder ingaan.

Dit hoofdstuk is als volgt ingedeeld. In §7.1 bewijzen we een aantal meta-stellingen die gebruikt worden bij het bewijs van de correctheid en de volledigheid van de propositielogica. Vervolgens wordt in §7.2 de correctheid bewezen. Tenslotte bewijzen we in §7.3 de volledigheid.

7.1 Enkele stellingen ter voorbereiding

In deze paragraaf zullen we een aantal meta-stellingen bewijzen. De meeste daarvan zullen we gebruiken in het bewijs van de correctheid en de volledigheid van de propositielogica. De eerste stelling staat bekend als de *deductiestelling.*

7.1.1 STELLING **Deductiestelling**
Zij $A, B \in PROP$, dan geldt:

$$A \vdash B \quad \Leftrightarrow \quad \vdash A \rightarrow B.$$

BEWIJS

($\Rightarrow$) Uit $A \vdash B$ volgt dat er in $\mathcal{F}$ een afleiding van B bestaat uit de hypothese A. Door toepassing van de regel $\rightarrow$-intro op de premissen A en B verkrijgt men een afleiding van $A \rightarrow B$, zodat $\vdash A \rightarrow B$.

($\Leftarrow$) Begin in $\mathcal{F}$ als volgt een afleiding: open een hypothese-interval met A als hypothese. Leid in dit hypothese-interval de formule $A \rightarrow B$ af. Zo'n afleiding bestaat, want $\vdash A \rightarrow B$. Pas tenslotte in dit zelfde interval de regel $\rightarrow$-elim toe op de hypothese A en de zojuist afgeleide formule $A \rightarrow B$. Het resultaat is een afleiding van $A \vdash B$. ∎

Het is niet moeilijk om uit bovenstaande stelling de volgende generalisaties te bewijzen.

7.1.2 COROLLARIUM *Zij $\Gamma \subseteq PROP$ en $A, A_1, \ldots, A_n, B \in PROP$, dan geldt:*

1. $\Gamma \cup \{A\} \vdash B \quad \Leftrightarrow \quad \Gamma \vdash A \rightarrow B,$
2. $A_1, \ldots, A_n \vdash B \quad \Leftrightarrow \quad \vdash A_1 \rightarrow (\ldots \rightarrow (A_n \rightarrow B)),$
3. $A_1, \ldots, A_n \vdash B \quad \Leftrightarrow \quad \vdash (A_1 \wedge \ldots \wedge A_n) \rightarrow B.$

De deductiestelling en haar corollarium zijn ook geldig als men hierin $\vdash$ vervangt door $\models$. Met behulp van de definitie van de relatie $\models$ is dit op eenvoudige wijze te controleren.

In stelling 7.1.3 zijn een aantal eigenschappen bij elkaar gezet die verband houden met de afleidingsregels uit het natuurlijke deductiesysteem $\mathcal{F}$. De stelling brengt tot uitdrukking dat de afleidingsregels van systeem $\mathcal{F}$ (zie tabel 6.1) *waarheidsbehoudend* zijn. Dit wil zeggen dat men met deze afleidingsregels geen onware uitspraken kan afleiden uit ware uitspraken. We zullen deze stelling gebruiken in het bewijs van de correctheid. De beweringen uit de stelling corresponderen achtereenvolgens met de regels $\vee$-intro, $\vee$-elim, $\neg$-intro, $\neg$-elim, $\rightarrow$-intro, $\rightarrow$-elim, $\wedge$-intro, $\wedge$-elim, $\leftrightarrow$-intro, $\leftrightarrow$-elim, en rei.

7.1.3 STELLING *Zij* $\Gamma \subseteq \Delta \subseteq PROP$ *en* $A, B, C \in PROP$*, dan geldt:*

1. $\Gamma \models A \quad \Rightarrow \quad \Gamma \models A \vee B,$
2. $\Gamma \models B \quad \Rightarrow \quad \Gamma \models A \vee B,$
3. $\Gamma \models A \vee B \quad \& \quad \Gamma \cup \{A\} \models C \quad \& \quad \Gamma \cup \{B\} \models C \quad \Rightarrow \quad \Gamma \models C,$
4. $\Gamma \cup \{A\} \models B \quad \& \quad \Gamma \cup \{A\} \models \neg B \quad \Leftrightarrow \quad \Gamma \models \neg A,$
5. $\Gamma \models \neg\neg A \quad \Leftrightarrow \quad \Gamma \models A,$
6. $\Gamma \cup \{A\} \models B \quad \Leftrightarrow \quad \Gamma \models A \rightarrow B,$
7. $\Gamma \models A \quad \& \quad \Gamma \models A \rightarrow B \quad \Rightarrow \quad \Gamma \models B,$
8. $\Gamma \models A \quad \& \quad \Gamma \models B \quad \Rightarrow \quad \Gamma \models A \wedge B,$
9. $\Gamma \models A \wedge B \quad \Rightarrow \quad \Gamma \models A,$
10. $\Gamma \models A \wedge B \quad \Rightarrow \quad \Gamma \models B,$
11. $\Gamma \models A \rightarrow B \quad \& \quad \Gamma \models B \rightarrow A \quad \Rightarrow \quad \Gamma \models A \leftrightarrow B,$
12. $\Gamma \models A \leftrightarrow B \quad \Rightarrow \quad \Gamma \models A \rightarrow B,$
13. $\Gamma \models A \leftrightarrow B \quad \Rightarrow \quad \Gamma \models B \rightarrow A,$
14. $\Gamma \models A \quad \Rightarrow \quad \Delta \models A.$

BEWIJS Het bewijs van de beweringen 1 t/m 14 is een vrijwel direct gevolg van de definitie van de relatie $\models$ en de semantische definities van de connectieven. Bij wijze van voorbeeld zullen we voor enkele gevallen het bewijs leveren. De stelling geldt ook als we overal $\models$ door $\vdash$ vervangen. In dat geval komt bewering 6 overeen met bewering 1 van corollarium 7.1.2.

1. Bewering 3 ($\vee$-elim).

   Stel dat v een model is voor Γ. We moeten dan bewijzen dat v ook een model is voor C, ofwel $v(C) = 1$.

   Uit $\Gamma \models A \vee B$ volgt, met het gegeven dat v een model is voor Γ, dat $v(A \vee B) = 1$. Dan is $v(A) = 1$ of $v(B) = 1$.

   In geval $v(A) = 1$ levert $\Gamma \cup \{A\} \models C$ dat $v(C) = 1$. Evenzo levert $v(B) = 1$ gecombineerd met $\Gamma \cup \{B\} \models C$, dat $v(C) = 1$.

   Combineren we de twee bevindingen, dan concluderen we dat $v(C) = 1$, waarmee het gestelde is bewezen.

2. Bewering 8 ($\wedge$-intro).

   Stel dat v een model is voor Γ. We moeten dan bewijzen dat v tevens een model is voor $A \wedge B$, ofwel $v(A \wedge B) = 1$.

   Uit het gegeven dat $\Gamma \models A$ volgt, samen met de veronderstelling dat v een model is voor Γ, dat $v(A) = 1$. Evenzo volgt uit $\Gamma \models B$ dat $v(B) = 1$.

   Combineren we $v(A) = 1$ en $v(B) = 1$, dan verkrijgen we $v(A \wedge B) = 1$, waarmee het gestelde is bewezen. ■

De laatste stelling van deze paragraaf is van belang voor het bewijs van de volledigheidsstelling. Deze zogenaamde *compactheidsstelling* reduceert het probleem van het vervulbaar zijn van een *oneindige* verzameling Γ van formules tot het vervulbaar zijn van alle *eindige* deelverzamelingen van Γ.

7.1.4 DEFINITIE **Eindige vervulbaarheid**

Een verzameling $\Gamma \subseteq PROP$ *noemt men* eindig vervulbaar *indien alle eindige deelverzamelingen van* Γ *vervulbaar zijn.*

De compactheidsstelling stelt dat de begrippen vervulbaarheid en eindige vervulbaarheid samenvallen.

7.1.5 STELLING **Compactheidsstelling**

Zij $\Gamma \subseteq PROP$*, dan is* Γ *vervulbaar dan en slechts dan als* Γ *eindig vervulbaar is.*

BEWIJS ($\Rightarrow$)*: Triviaal.*

($\Leftarrow$)*: Neem aan dat* Γ *eindig vervulbaar is. Het idee van het bewijs is om een verzameling* $\Delta \supseteq \Gamma$ *te definiëren die 'maximaal' is in de zin dat voor iedere formule* $F \in PROP$ *geldt ofwel* $F \in \Delta$*, ofwel* $\neg F \in \Delta$*. Op basis van deze verzameling* Δ *kunnen we dan een valuatie* v *definiëren die een model is voor* Δ *en dus ook voor* Γ*. Deze valuatie* v *is zodanig gedefinieerd dat* $v(p) = 1$ *dan en slechts dan als* p *een propositiesymbool is waarvoor* $p \in \Delta$*.*

In de definitie van de verzameling Δ *maken we gebruik van een* opsomming *van de formules in PROP. Daarom laten we eerst zien dat zo'n opsomming bestaat. Daarna vervolgen we met de andere stappen van het bewijs.*

a. De verzameling *PROP* is aftelbaar.

Een verzameling V is aftelbaar *dan en slechts dan als er een surjectieve afbeelding van* $\mathbb{N}$ *op V bestaat. We bewijzen dat PROP aftelbaar is door aan te geven hoe een opsomming* $F_0, F_1, F_2, \ldots$ *van de formules in PROP kan worden verkregen.*

Zij $\Theta_{m,n}$ *de verzameling van alle proposities F zodanig dat* $comp(F) = m$ *en zodanig dat voor alle propositiesymbolen* p_i *die in F voorkomen, geldt* $i \leq n$*. Evident is voor alle* $m, n \in \mathbb{N}$ *de verzameling* $\Theta_{m,n}$ *eindig, zodat er een rijtje* $\overline{\theta_{m,n}}$ *bestaat van alle formules in* $\Theta_{m,n}$*. We verkrijgen nu een opsomming van alle formules in PROP door de rijtjes* $\theta_{m,n}$ *op de volgende wijze aan elkaar te plakken:*

$$\theta_{0,0}, \theta_{1,0}, \theta_{0,1}, \theta_{2,0}, \theta_{1,1}, \theta_{0,2}, \theta_{3,0}, \theta_{2,1}, \theta_{1,2}, \theta_{0,3}, \ldots.$$

b. De constructie van de verzameling Δ.

Zij $F_0, F_1, F_2, \ldots$ *een opsomming van de formules in PROP. Definieer de verzameling* Δ *als volgt:*

$$\Delta = \bigcup_{i \in \mathbb{N}} \Delta_i.$$

waarbij de verzamelingen Δ_i *inductief gedefinieerd zijn door:*

$$\begin{array}{rcl} \Delta_0 & = & \Gamma \\ \Delta_{n+1} & = & \left\{ \begin{array}{ll} \Delta_n \cup \{F_n\} & \textit{als dit eindig vervulbaar is,} \\ \Delta_n \cup \{\neg F_n\} & \textit{anders.} \end{array} \right. \end{array}$$

c. De verzamelingen Δ_i $(i \in \mathbb{N})$ zijn eindig vervulbaar.

Dit bewijzen we met behulp van volledige inductie.

- *Basisstap:* $i = 0$.
  De verzameling $\Delta_0 = \Gamma$ *is eindig vervulbaar krachtens de aanname.*
- *Inductiestap:* $i = n + 1$.
  De inductiehypothese luidt dat Δ_n *eindig vervulbaar is. We moeten laten zien dat* Δ_{n+1} *eindig vervulbaar is. Er zijn nu twee mogelijkheden.*
  - $\Delta_n \cup \{F_n\}$ *is eindig vervulbaar.*
    Maar dan is Δ_{n+1} *per definitie ook eindig vervulbaar.*
  - $\Delta_n \cup \{F_n\}$ *is niet eindig vervulbaar.*
    In dat geval bestaat er een eindige deelverzameling $\Lambda_0 \subseteq \Delta_n$ *zodanig dat* $\Lambda_0 \cup \{F_n\}$ *niet vervulbaar is. Uit stelling 5.1.1 volgt dan* $\Lambda_0 \models \neg F_n$*.*
    Om te bewijzen dat Δ_{n+1} *eindig vervulbaar is, kunnen we volstaan met aan te tonen dat* $\Lambda \cup \{\neg F_n\}$ *vervulbaar is voor een willekeurige, eindige deelverzameling* $\Lambda \subseteq \Delta_n$*. Als* Λ *zo'n eindige deelverzameling is, dan is* $\Lambda \cup \Lambda_0$ *dat ook. Deze verzameling is vervulbaar wegens de eindige vervulbaarheid van* Δ_n*. Laat v een model zijn voor* $\Lambda \cup \Lambda_0$*. Dan is v ook een model voor* Λ_0 *en dus ook voor* $\neg F_n$ *wegens* $\Lambda_0 \models \neg F_n$*. Maar dit betekent dat v de verzameling* $\Lambda \cup \{\neg F_n\}$ *vervult.*

d. De verzameling Δ is eindig vervulbaar.

Zij Λ *een eindige deelverzameling van* Δ *en zij m de hoogste index zodat* $F_m \in \Lambda$ *of* $\neg F_m \in \Lambda$*. Uit de definitie van de verzamelingen* Δ_i *volgt nu onmiddellijk dat* $\Lambda \subseteq \Delta_{m+1}$*. Aangezien* Δ_{m+1} *eindig vervulbaar is, moet* Λ *derhalve vervulbaar zijn. Hieruit volgt het gestelde.*

e. Voor iedere $F \in PROP$ geldt ofwel $F \in \Delta$, ofwel $\neg F \in \Delta$.

Wegens de constructie van Δ *geldt voor alle* $F \in PROP$ *dat* $F \in \Delta$ *of* $\neg F \in \Delta$*. We moeten dus nog aantonen dat voor geen enkele formule* $F \in PROP$ *geldt* $F \in \Delta$ *en* $\neg F \in \Delta$*. Als dit echter het geval zou zijn, dan zou* $\{F, \neg F\}$ *een onvervulbare eindige* *deelverzameling van* Δ *zijn, hetgeen onmogelijk is wegens de eindige vervulbaarheid van* Δ*.*

Deze eigenschap van Δ *zullen we verder in dit bewijs de* maximaliteit *van* Δ *noemen.*

f. De definitie van de valuatie v.

De zojuist bewezen bewering dat voor iedere $F \in PROP$ *ofwel* $F \in \Delta$*, ofwel* $\neg F \in \Delta$*, geldt in het bijzonder voor de propositiesymbolen* p*, zodat we een valuatie* v *kunnen definiëren die voldoet aan:*

$$v(p) = 1 \quad \Leftrightarrow \quad p \in \Delta.$$

Merk op dat dit betekent dat $v(p) = 0$ *voor alle propositiesymbolen* p *waarvoor* $p \notin \Delta$*.*

g. De valuatie v is een model voor Γ.

Met structurele inductie over de formules $F \in \Delta$ *zullen we de volgende uitspraak bewijzen:*

$$v(F) = 1 \quad \Leftrightarrow \quad F \in \Delta. \qquad (*)$$

Hieruit volgt dan dat v *een model is voor* Δ *en in het bijzonder voor* Γ*, aangezien* $\Gamma \subseteq \Delta$*.*

We hebben de volgende mogelijkheden voor F*:*

- $F = p$ *voor een propositiesymbool* p*. Dus voldoet* F *aan* $(*)$*.*
- $F = \neg A$ *voor* $A \in PROP$*.*

  $v(\neg A) = 1$ *dan en slechts dan als* $v(A) = 0$*. Wegens de inductieveronderstelling is dit dan en slechts dan het geval als* $A \notin \Delta$*. Maar wegens de maximaliteit van* Δ *is dit dan en slechts dan het geval als* $\neg A \in \Delta$ *. Dit betekent dat* F *aan* $(*)$ *voldoet.*
- $F = (A \vee B)$ *voor* $A, B \in PROP$*.*

  $v(A \vee B) = 1$ *dan en slechts dan als* $v(A) = 1$ *of* $v(B) = 1$*. Wegens de inductiehypothese is dit laatste equivalent met* $A \in \Delta$ *of* $B \in \Delta$*.*

  Als $A \in \Delta$ *of* $B \in \Delta$*, dan is* $A \vee B \in \Delta$*. Als dat namelijk niet het geval zou zijn, dan zou wegens de maximaliteit van* Δ *moeten gelden* $\neg(A \vee B) \in \Delta$*. Dit is echter onmogelijk vanwege de onvervulbaarheid van zowel* $\{A, \neg(A \vee B)\}$ *als* $\{B, \neg(A \vee B)\}$*.*

  Omgekeerd, als $A \vee B \in \Delta$*, dan is* $A \in \Delta$ *of* $B \in \Delta$*. Als dat namelijk niet het geval zou zijn, dan zou wegens de maximaliteit van* Δ *zowel* $\neg A \in \Delta$ *zijn als* $\neg B \in \Delta$*, wat onmogelijk is gezien de onvervulbaarheid van* $\{\neg A, \neg B, A \vee B\}$*.*

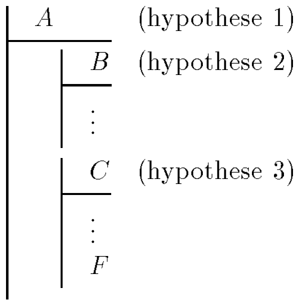
  Uit deze observaties volgt dat F *aan* $(*)$ *voldoet.*

De gevallen dat $F = (A \wedge B)$*,* $F = (A \rightarrow B)$ *of* $F = (A \leftrightarrow B)$ *gaan net zo, en worden aan de lezer overgelaten.*

Hiermee is het bewijs van de compactheidsstelling voltooid. ■

Uit het bewijs van de voorgaande stelling zou men de indruk kunnen krijgen dat iedere eindig vervulbare verzameling slechts één model v bezit. Dit is echter niet het geval. De definitie van v is afhankelijk van de constructie van Δ die gebaseerd is op de wijze van opsomming van de formules uit $PROP$. Aangezien deze ordening niet uniek is, zijn er meerdere mogelijkheden om Γ uit te breiden tot een verzameling Δ.

7.1.6 VOORBEELD De constructie van Δ's.
Zij $\Gamma = \{p_i \mid i \text{ is oneven}\}$ een verzameling van propositiesymbolen. Zij θ_1 een opsomming van $PROP$ waarin $F_0 = p_0$, en θ_2 een opsomming waarin $F_0 = \neg p_0$. Dan geven deze twee opsommingen aanleiding tot twee verschillende, maximale verzamelingen $\Delta_1 \supseteq \Gamma$ en $\Delta_2 \supseteq \Gamma$; namelijk zodanig dat respectievelijk $p_0 \in \Delta_1$ en $\neg p_0 \in \Delta_2$. Deze verzamelingen leveren verschillende modellen v_1 en v_2 voor Γ op: $v_1(p_0) = 1$ en $v_2(p_0) = 0$. ■



Figuur 7.1: Aktieve-hypothesenverzamelingen.

7.1.7 COROLLARIUM *Zij* $\Gamma \subseteq PROP$ *en* $F \in PROP$, *dan geldt dat* $\Gamma \models F$ *dan en slechts dan als er een eindige deelverzameling* $\Gamma_0 \subseteq \Gamma$ *bestaat zodanig dat* $\Gamma_0 \models F$.

BEWIJS ($\Leftarrow$): Triviaal.

($\Rightarrow$): Stel $\Gamma \models F$. Uit stelling 5.1.1 volgt dat $\Gamma \cup \{\neg F\}$ niet vervulbaar is. Passen we nu de compactheidsstelling toe, dan zien we dat er een eindige deelverzameling $\Gamma_0 \subseteq \Gamma \cup \{\neg F\}$ moet bestaan die niet vervulbaar is. Voor deze Γ_0 geldt dat $\Gamma_0 \cup \{\neg F\}$ ook niet vervulbaar is. Hieruit volgt, wederom met behulp van stelling 5.1.1, dat $\Gamma_0 \models F$, waarmee het corollarium is bewezen. ■

7.2 Correctheid

In deze paragraaf zullen wij de correctheidsstelling bewijzen. De inhoud van deze stelling is dat de relatie $\vdash$ correct is met betrekking tot de semantische relatie $\models$, met andere woorden, als met behulp van het (uitgebreide) systeem van Fitch $\Gamma \vdash F$ kan worden afgeleid, dan is de formule F ook inderdaad een logisch gevolg van de verzameling formules Γ, ofwel $\Gamma \models F$.

In het bewijs van deze stelling maken we gebruik van het begrip *aktieve-hypothesenverzameling*. De achtergrond hiervan is de volgende. In een afleiding in het systeem van Fitch worden formules afgeleid uit hypothesen. Bij iedere formule in een afleiding hoort een verzameling van de hypothesen waaruit die formule is afgeleid. Deze verzameling wordt de aktieve-hypothesenverzameling van die formule genoemd. De naamgeving komt voort uit het feit dat de hypothesen uit deze verzameling nog *aktief* zijn met betrekking tot het hypothese-interval waarin de betreffende formule ligt. In figuur 7.1 is de aktieve-hypothesenverzameling van formule F gelijk aan $\{A, C\}$ (merk op:

formule B hoort hier niet bij!). Verder is een hypothese zelf altijd een element van zijn hypothese-interval. Dus de aktieve-hypothesenverzameling van B is gelijk aan $\{A, B\}$.

7.2.1 DEFINITIE **Aktieve-hypothesenverzameling**

Zij $\mathbf{D}$ *een afleiding in het systeem* $\mathcal{F}$ *of* $\mathcal{F}_{uit}$ *bestaande uit het interval* $[1,n]$, *de formules* $F_1, \ldots, F_n$, *en zij* $\mathbf{H}$ *de verzameling van hypothese-intervallen van* $\mathbf{D}$. *Dan is de* aktieve-hypothesenverzameling *van een formule* F_i, *notatie* $\mathbf{A}(i)$, *gedefinieerd als:*

$$\mathbf{A}(i) = \{F_k \mid [k,l] \in \mathbf{H} \text{ en } i \in [k,l] \text{ voor zekere } l\}.$$

Ingeval in een afleiding in $\mathcal{F}_{uit}$ de hypothesen in het eerste hypothese-interval zijn ondergebracht, is er in *formele zin* sprake van even zoveel geneste hypothese-intervallen als hypothesen, zoals aan het eind van paragraaf 6.3 is afgesproken. Dit betekent dat bovenstaande definitie ook gebruikt kan worden voor dergelijke afleidingen.

7.2.2 STELLING **Correctheidsstelling**

Zij $\Gamma \subseteq PROP$ *en* $F \in PROP$, *dan geldt:*

$$\Gamma \vdash F \quad \Rightarrow \quad \Gamma \models F.$$

BEWIJS *Neem aan dat* $\Gamma \vdash F$. *Dan bestaat er volgens definitie 6.1.5 een eindige verzameling* $\Gamma_0 = \{P_1, \ldots, P_m\} \subseteq \Gamma$ $(m \in \mathbb{N})$ *zodanig dat* $\Gamma_0 \vdash F$. *Dit betekent dat er een afleiding* $\mathbf{D}$ *bestaat met interval* $D = [1,n]$, *formules* $F_1, \ldots, F_n$ *en hypothese-intervallen* $\mathbf{H}$ *zodanig dat* $F_i = P_i$ $(1 \leq i \leq m)$ *een hypothese is en* $F_n = F$. *Merk op dat* $[1,n]$ *een nulde interval is als* $m = 0$.

We gaan nu met volledige inductie naar i *bewijzen dat iedere formule* F_i *in afleiding* $\mathbf{D}$ *een logisch gevolg is van haar aktieve-hypothesenverzameling* $\mathbf{A}(i)$, *formeel:*

$$\mathbf{A}(i) \models F_i \quad \text{voor alle } i \in [1,n]. \qquad (\dagger)$$

Als dit bewezen is, volgt de stelling uit de volgende observaties. Indien $m = 0$, *dan is* $\mathbf{A}(n) = \emptyset$, $F_n = F$ *en* $gr(n) = 0$. *Nemen we nu* $i = n$ *in uitspraak* $(\dagger)$, *dan verkrijgen we* $\models F$, *hetgeen we moesten bewijzen.*

Is daarentegen $m \geq 1$, *dan geldt dat* $\mathbf{A}(n) = \Gamma_0$, $F_n = F$ *en* $gr(n) = m$. *Toepassing van* $(\dagger)$ *met* $i = n$ *levert dan* $\Gamma_0 \models F$. *Aangezien* $\Gamma_0 \subseteq \Gamma$ *volgt met stelling 7.1.3(14) dat* $\Gamma \models F$, *waarmee de stelling ook voor dit geval is bewezen.*

We gaan nu uitspraak $(\dagger)$ *bewijzen met inductie naar* i.

1. *Basisstap:* $i = 1$.

   We moeten laten zien dat $\mathbf{A}(1) \models F_1$. *Hieraan is altijd voldaan, aangezien* F_1 *een hypothese moet zijn (ook in het geval dat* $m = 0$*), zodat* $\mathbf{A}(1) = \{F_1\}$.

2. *Inductiestap:* $i = j + 1$.

   De inductieveronderstelling (IH) luidt dat $\mathbf{A}(k) \models F_k$ *voor alle* k *waarvoor* $1 \leq k \leq j$.

   Als F_{j+1} *een hypothese is, dan geldt* $F_{j+1} \in \mathbf{A}(j+1)$, *zodat* $\mathbf{A}(j+1) \models F_{j+1}$.

   Neem dus aan dat F_{j+1} *geen hypothese is. In dat geval is* F_{j+1} *afgeleid uit voorafgaande formules door toepassing van een afleidingsregel uit tabel 6.1. Er zijn de volgende mogelijkheden:*

   - F_{j+1} *is afgeleid door toepassing van de regel* $\vee$*-intro. Dit betekent dat* $F_{j+1} = (A \vee B)$ *voor zekere* $A, B \in PROP$. *Het interval waarin* F_{j+1} *ligt, bevat een formule* $F_k = A$ *(of* $F_k = B$*) waarbij* $k < j + 1$. *Volgens (IH) is* $\mathbf{A}(k) \models F_k$. *Bovendien geldt* $\mathbf{A}(k) = \mathbf{A}(j+1)$, *zodat ook* $\mathbf{A}(j+1) \models F_k$. *Passen we nu stelling 7.1.3 (geval 1 of 2) toe, dan volgt* $\mathbf{A}(j+1) \models F_{j+1}$.

- *F_{j+1} is afgeleid door toepassing van één van de afleidingsregels $\neg$-elim, $\rightarrow$-elim, $\wedge$-intro, $\wedge$-elim, $\leftrightarrow$-intro of $\leftrightarrow$-elim. In deze gevallen is het bewijs voor $\mathbf{A}(j+1) \models F_{j+1}$ analoog aan het bewijs voor het geval dat de regel $\vee$-intro is gebruikt.*
- *F_{j+1} is afgeleid door toepassing van de regel $\rightarrow$-intro. Dit betekent dat $F_{j+1} = (A \rightarrow B)$ voor zekere $A, B \in PROP$. Het interval waarin F_{j+1} ligt, bevat een hypothese-interval $[k, l]$ waarvoor $F_k = A$, $F_l = B$ en $k < l < j+1$. Volgens (IH) is $\mathbf{A}(l) \models B$. Bovendien geldt $\mathbf{A}(l) = \mathbf{A}(j+1) \cup \{A\}$, zodat $\mathbf{A}(j+1) \cup \{A\} \models B$. Uit stelling 7.1.3(6) volgt dan $\mathbf{A}(j+1) \models F_{j+1}$.*
- *F_{j+1} is afgeleid door toepassing van één van de regels $\vee$-elim of $\neg$-intro. In deze gevallen is het bewijs voor $\mathbf{A}(j+1) \models F_{j+1}$ analoog aan het bewijs voor het geval dat de regel $\rightarrow$-intro is gebruikt.*
- *F_{j+1} is afgeleid door toepassing van de regel rei. Stel dat de formule F_{j+1} in het interval I ligt, dan moet er een interval J en een formule $F_k = F_{j+1}$ bestaan zodanig dat $I \subseteq J$, $k < j+1$ en F_k ligt in J. Volgens (IH) is $\mathbf{A}(k) \models F_k$. Bovendien geldt $\mathbf{A}(k) \subseteq \mathbf{A}(j+1)$, zodat met stelling 7.1.3(14) volgt $\mathbf{A}(j+1) \models F_{j+1}$.*

Hiermee is de correctheidsstelling bewezen. ■

Met behulp van de correctheidsstelling kan men laten zien dat sommige formules niet afleidbaar zijn. Immers, als een formule F niet algemeen geldig is, dan kan F volgens de correctheidsstelling (neem $\Gamma = \emptyset$) niet afleidbaar zijn.

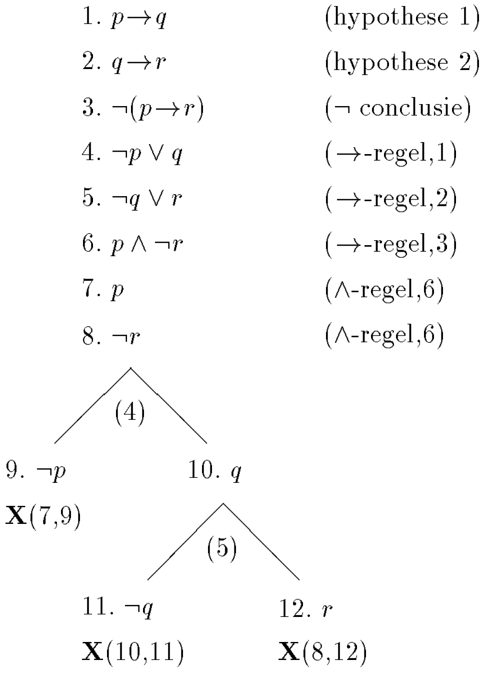
7.2.3 VOORBEELD Algemene geldigheid en afleidbaarheid.
De formule $A = p \rightarrow (p \rightarrow q)$ is niet algemeen geldig. Dit is eenvoudig met de boommethode aan te tonen. Volgens de correctheidsstelling mogen we concluderen dat A niet afleidbaar is, ofwel *niet* $\vdash A$. ■

Als F een algemeen geldige formule is, dan kunnen we ons afvragen of F ook afleidbaar is. De correctheidsstelling geeft geen antwoord op deze vraag. In de volgende paragraaf zullen we zien dat de volledigheidsstelling hierop wel een antwoord geeft, en wel een bevestigend.

7.3 Volledigheid

In deze paragraaf zullen we de volledigheid van de propositielogica bewijzen. De term *volledigheid* slaat in dit verband op het feit dat het systeem van Fitch voldoende krachtig is: als $\Gamma \models F$, dan is de formule F ook afleidbaar uit de verzameling formules Γ, ofwel $\Gamma \vdash F$.

Het hoofdidee van het bewijs is als volgt. Als $\Gamma \models F$ voor een eindige verzameling formules Γ en een formule F, dan sluit een boom B voor $\Gamma \cup \{\neg F\}$. Als we nu een methode zouden hebben om uit B een afleiding van $\Gamma \vdash F$ te construeren, dan zouden we klaar zijn. Zo'n methode bestaat: het is mogelijk om een boom B voor $\Gamma \cup \{\neg F\}$ te transformeren naar een afleiding van $\Gamma \vdash F$



Figuur 7.2: $p \to q, q \to r \models p \to r$.

in het uitgebreide systeem $\mathcal{F}_{uit}$ van Fitch. Hierbij komt het van pas dat de toegevoegde afleidingsregels van dit systeem (zie tabel 6.2) overeenkomen met die van de boommethode.

Als Γ echter geen eindige verzameling is, dan gaat deze methode niet op. De boommethode werkt immers alleen voor eindige verzamelingen. Nu komt het corollarium van de compactheidsstelling van pas: als $\Gamma \models F$, dan bestaat er een *eindige* deelverzameling Γ_0 van Γ zodanig dat $\Gamma_0 \models F$. Als we nu onze methode toepassen op een sluitende boom voor $\Gamma_0 \cup \{\neg F\}$, dan verkrijgen we een afleiding van $\Gamma_0 \vdash F$. Hieruit volgt direct dat $\Gamma \vdash F$ (zie definitie 6.1.5).

We beginnen met een voorbeeld van een transformatie van een boom naar een afleiding in het uitgebreide systeem van Fitch. In hoofdstuk 5 hebben we een boom gezien voor $p \to q, q \to r \models p \to r$. Voor de duidelijkheid herhalen we deze boom in figuur 7.2. De met de boom corresponderende afleiding van $p \to q, q \to r \vdash p \to r$ in $\mathcal{F}_{uit}$ staat in figuur 7.3.

De clou van de afleiding is dat de boom wordt 'geïmiteerd' waarbij vertak-

1.	$p \rightarrow q$	(hypothese 1)
2.	$q \rightarrow r$	(hypothese 2)
3.	$\neg(p \rightarrow r)$	($\neg$ conclusie)
4.	$\neg p \vee q$	($\rightarrow$-regel2,1)
5.	$\neg q \vee r$	($\rightarrow$-regel2,2)
6.	$p \wedge \neg r$	($\rightarrow$-regel1,3)
7.	p	($\wedge$-elim,6)
8.	$\neg r$	($\wedge$-elim,6)
9.	$\neg p$	(linker vertakking van 4)
9A.	X	(contra,7,9)
10.	q	(rechter vertakking van 4)
11.	$\neg q$	(linker vertakking van 5)
11A.	X	(contra,10,11)
12.	r	(rechter vertakking van 5)
12A.	X	(contra,8,12)
13.	X	($\vee$-elim,5,11A,12A)
14.	$X = p \rightarrow r$	($\vee$-elim,4,9A,13)
15.	$\neg\neg(p \rightarrow r)$	($\neg$-intro,3,14)
16.	$p \rightarrow r$	($\neg$-elim,15)

Figuur 7.3: De $\mathcal{F}_{uit}$-versie van de boom uit figuur 7.2.

kingen in de boom overeenkomen met paren van nieuwe hypothese-intervallen in de boom. De nummering van de formules in de afleiding is gelijk aan die in de boom. We maken nu de volgende observaties betreffende de afleiding.

1. De afleiding begint met een hypothese-interval met als hypothesen de formules uit Γ, in dit geval $p \rightarrow q$ en $q \rightarrow r$.
2. Het volgende hypothese-interval heeft als hypothese de ontkenning van de conclusie F, in dit geval $\neg(p \rightarrow r)$.
3. De afleiding volgt steeds letterlijk de boom tot aan een vertakking.
4. Voor iedere eerste formule van een vertakking wordt een nieuw hypothese-interval geopend met die formule als hypothese. De twee hypothese-intervallen worden onder elkaar geplaatst. In het voorbeeld worden bij vertakking (4) twee hypothese-intervallen onder elkaar geopend met respectievelijk de formules $\neg p$ en q als hypothesen. In het tweede interval wordt dit op recursieve wijze gedaan voor vertakking (5).
5. Als een tak van de boom sluit, wordt de contra-regel toegepast op de formules met het zelfde nummer als in de boom, met als resultaat een willekeurige formule X die later geïnstantieerd zal worden met de conclusie F, in dit geval $p \rightarrow r$.
6. Als alle formules in de boom aan de beurt zijn geweest, kan de afleiding compleet worden gemaakt door een aantal toepassingen van de regel $\vee$-elim, en tenslotte door achtereenvolgens de regels $\neg$-intro en $\neg$-elim toe te passen.

Hierboven hebben we een algemene methode gegeven om een boom voor $\Gamma \cup \{\neg F\}$ te transformeren tot een afleiding van $\Gamma \vdash F$ in het systeem $\mathcal{F}_{uit}$. Merk op dat de **X**-en in de boom géén formules zijn: zij dienen om aan te geven dat een tak afsluit. De X-en in de afleiding stellen echter wèl formules voor! Deze worden geïnstantieerd met de conclusie F.

Nu we de methode hebben uiteengezet om een boom te transformeren naar een afleiding in het uitgebreide systeem van Fitch, resteert ons nog om te laten zien dat deze methode altijd een correcte afleiding oplevert. Dit vormt een onderdeel van het bewijs van de volledigheidsstelling.

7.3.1 STELLING **Volledigheidsstelling**
Zij $\Gamma \subseteq PROP$ *en* $F \in PROP$*, dan geldt:*

$$\Gamma \models F \quad \Rightarrow \quad \Gamma \vdash F$$

BEWIJS *Neem aan dat* $\Gamma \models F$. *Wegens corollarium 7.1.7 bestaat er een eindige verzameling* $\Gamma_0 = \{P_1, \ldots, P_m\} \subseteq \Gamma$ $(m \in \mathbb{N})$ *zodanig dat* $\Gamma_0 \models F$. *Dan is wegens stelling 5.1.1 de verzameling* $\Gamma_0 \cup \{\neg F\}$ *niet vervulbaar. Dit betekent dat een boom* B *voor* $\Gamma_0 \cup \{\neg F\}$ *sluit. Deze boom* B *kan volgens de bovenstaande methode worden omgezet naar een afleiding* $\mathbf{D}$ *van* $\Gamma_0 \vdash F$ *in* $\mathcal{F}_{uit}$ *(deze afleiding kan eventueel geëxpandeerd worden tot een afleiding in het niet-uitgebreide systeem* $\mathcal{F}$*). Het bewijs kan dan worden besloten met de observatie dat uit* $\Gamma_0 \vdash F$ *volgt dat* $\Gamma \vdash F$.

We zullen nu met volledige inductie naar de lengte van de verkregen, niet-geëxpandeerde bewijsfiguur $\mathbf{D}$ *bewijzen dat de methode correct is, dat wil zeggen dat* $\mathbf{D}$ *een correcte afleiding is.*

Neem aan dat $\mathbf{D}$ *bestaat uit het interval* $D = [1, n]$, *de formules* $F_1, \ldots, F_n$ *en de hypothese-intervallen* $\mathbf{H}$. *Voor* Γ_0 *zijn er twee mogelijkheden:* $\Gamma_0 = \emptyset$ *en* $\Gamma_0 \neq \emptyset$. *In het eerste geval is* $[1, n]$ *een nulde interval. In het tweede geval bezit* $[1, n]$ m *hypothesen. In paragraaf 6.3 hebben we echter afgesproken dat intervallen met meer dan één hypothese eigenlijk geneste intervallen zijn met precies één hypothese. Derhalve is er in feite geen sprake van één interval* $[1, n]$ *maar van de intervallen* $[1, n], \ldots, [m, n]$, *waarbij* $F_i = P_i$ $(1 \leq i \leq m)$ *de hypothesen zijn.*

Men bewijst nu met inductie naar i *dat met betrekking tot bewijsfiguur* $\mathbf{D}$ *de volgende uitspraak geldt:*

> Voor alle $i \in [1, n]$ is F_i een hypothese, of een formule verkregen door toepassing van een afleidingsregel van het systeem $\mathcal{F}_{uit}$ op formules die aan F_i voorafgaan. (‡)

In het bewijs van deze uitspraak maakt men gebruik van de eigenschappen van de boom en die van de methode. Hierbij komt het eerder genoemde feit van pas dat het systeem $\mathcal{F}_{uit}$ *de syntactische versies van de regels van de boommethode bevat.*

Het bewijs van de correctheid van de methode kan dan worden besloten met de volgende observaties. Als $\Gamma_0 = \emptyset$, *dan is wegens de methode* $[1, n]$ *een nulde interval waarvoor* $F_n = F$ *en* $gr(n) = 0$. *Uit uitspraak* (‡) *volgt nu dat* $\mathbf{D}$ *een afleiding van* F *is in het uitgebreide systeem van Fitch* $\mathcal{F}_{uit}$. *In het geval dat* $\Gamma_0 \neq \emptyset$, *dan is eveneens op grond van de methode en uitspraak* (‡) *eenvoudig in te zien dat* $\mathbf{D}$ *een afleiding is van* F *uit de premissen* $P_1, \ldots, P_m$ *in het systeem* $\mathcal{F}_{uit}$.

Als we de nog niet ingevulde details van het bewijs aan de lezer overlaten, dan is hiermee de volledigheidsstelling bewezen. ■

De volgende, zogenaamde consistentiestelling is equivalent met de volledigheidsstelling (zie opgave 7). Hiertoe definiëren we eerst het begrip *consistentie.*

7.3.2 DEFINITIE **Consistentie**

Een verzameling $\Gamma \subseteq PROP$ *wordt* consistent *genoemd dan en slechts dan als er geen formule* $F \in PROP$ *bestaat zodanig dat geldt* $\Gamma \vdash F$ *en* $\Gamma \vdash \neg F$.

7.3.3 VOORBEELD Inconsistente verzamelingen.

De verzamelingen $\{p, \neg p\}$ en $\{p, p \to q, \neg q\}$ zijn inconsistent. ■

7.3.4 STELLING **Consistentiestelling**

Als $\Gamma \subseteq PROP$ *consistent is, dan is* Γ *vervulbaar.*

BEWIJS We bewijzen de stelling met contrapositie. Zij Γ onvervulbaar, dan moeten we bewijzen dat Γ niet consistent is.

Kies een willekeurige formule $F \in PROP$. Aangezien Γ onvervulbaar is, geldt zowel $\Gamma \models F$ als $\Gamma \models \neg F$ (dit is een rechtstreeks gevolg van de definitie van logisch gevolg!). Passen we nu de volledigheidsstelling toe, dan verkrijgen we $\Gamma \vdash F$ en $\Gamma \vdash \neg F$. Dit betekent dat Γ inconsistent is. ■

Het is niet moeilijk om te bewijzen dat het omgekeerde van deze stelling equivalent is met de correctheidsstelling (zie opgave 6).

7.4 Opgaven

1. Bewijs corollarium 7.1.2.
2. Bewijs de beweringen 1, 2, 4 t/m 7 en 9 t/m 14 uit stelling 7.1.3.
3. Zij $A, B, C \in PROP$. Bewijs de volgende metabeweringen met behulp van de boommethode, en transformeer de verkregen bomen naar afleidingen in het uitgebreide systeem van Fitch $\mathcal{F}_{uit}$:
   (a) $(B \wedge C) \to \neg A,\ A \wedge C \models \neg B$.
   (b) $\models ((A \vee (A \to B)) \to B) \to B$.
   (c) $A,\ (A \wedge \neg B) \to C,\ \neg(A \wedge C) \models B$.
4. Beschouw in het bewijs van de compactheidsstelling de bewering dat v een model is voor Δ. Voltooi de inductiestap in het bewijs van deze bewering door de gevallen te behandelen dat $F = (A \wedge B)$, $F = (A \to B)$ en $F = (A \leftrightarrow B)$.
5. Beschouw de inductiestap in het bewijs van de correctheidsstelling. Behandel het geval dat de formule F_{j+1} is afgeleid met behulp van de regel $\vee$-elim.
6. Bewijs uitspraak (‡) uit het bewijs van de volledigheidsstelling.
7. Laat zien dat de consistentiestelling equivalent is met de volledigheidsstelling en dat het omgekeerde van de consistentiestelling equivalent is met de correctheidsstelling.
8. Zij $\mathcal{F}^-$ het systeem van Fitch *zonder* de regel $\vee$-elim. Laat zien dat $\mathcal{F}^-$ correct, maar niet volledig is.
9. Zij $\mathcal{F}^+$ het systeem van Fitch waaraan de volgende afleidingsregel is toegevoegd: *uit* $A \to B$ *kan men* $B \to A$ *afleiden (in hetzelfde hypothese-interval)*. Laat zien dat $\mathcal{F}^+$ wel volledig maar, niet correct is.

Deel II

Predicatenlogica

Hoofdstuk 8

Syntaxis van de Predicatenlogica

In de voorgaande hoofdstukken hebben we de propositielogica geïntroduceerd als een formalisme waarmee redeneringen kunnen worden beschreven. Hierbij konden redeneringen worden geanalyseerd tot op het niveau van enkelvoudige beweringen. Een groot aantal redeneringen, ook wiskundige, kan niet met behulp van de propositielogica worden geanalyseerd. Immers, de interne struktuur van de enkelvoudige beweringen wordt buiten beschouwing gelaten. De volgende redenering, die we al in paragraaf 1.2 zijn tegengekomen, kan niet met behulp van de propositielogica worden geanalyseerd:

(8.1) Alle mensen zijn sterfelijk.
Socrates is een mens.
$\therefore$ Socrates is sterfelijk.

Van deze intuïtief geldige redenering zou de vertaling in de propositielogica er als volgt uitzien:

(8.2) p
q
$\therefore\ r$

Het is eenvoudig om na te gaan dat in de propositielogica *niet* geldt $p, q \models r$ (voor propositiesymbolen p, q en r). We kunnen dus concluderen dat de propositielogica niet rijk genoeg is om alle gewenste redeneringen te analyseren. Om deze reden introduceren wij in dit hoofdstuk een rijker logisch formalisme: de *predicatenlogica*. In deze logica worden ook enkelvoudige beweringen ontleed. Het zal blijken dat redenering (8.1) logisch geldig is in de predicatenlogica.

De indeling van dit hoofdstuk is als volgt. In §8.1 geven we een informele inleiding tot de predicatenlogica. Vervolgens wordt in §8.2 de syntaxis behandeld van *eerste-ordetalen*, zoals predicaatlogische talen worden genoemd. Paragraaf 8.3 heeft recursie en inductie als onderwerp. In §8.4 wordt de notie substitutie gedefinieerd en in §8.5, tenslotte, wordt gedemonstreerd hoe redeneringen in natuurlijke taal kunnen worden vertaald naar de predicatenlogica.

8.1 Predicaten, individuen en kwantoren

Bekijken we nog eens redenering (8.1), en proberen we daarbij de logische structuur ervan te begrijpen, dan lijkt het redelijk om deze structuur als volgt weer te geven (zie ook §1.2):

(8.3) Alle x die M zijn, zijn ook S.
a is M.
$\therefore$ a is S.

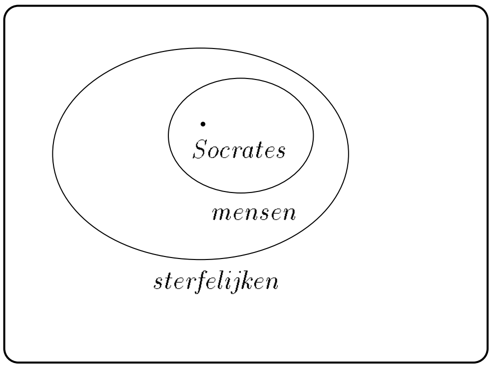
Daarbij dient men voor M het woord *mens*, voor S het woord *sterfelijk* en voor a de naam *Socrates* te lezen. In feite duiden de woorden *mens* en *sterfelijk* zogenaamde *eigenschappen* aan. Namen, daarentegen, duiden personen, dieren of dingen aan. We zeggen dat namen *individuen* aanduiden. Eigenschappen worden in de logica ook wel *predicaten* genoemd.

In de eerste regel van (8.3) komt de letter x voor. Deze x is een *variabele* en wordt in combinatie met het woord *alle* gebruikt om de eigenschappen M en S op een bepaalde wijze aan elkaar te koppelen. In termen van verzamelingen kan dit worden uitgedrukt door te zeggen dat de verzameling van individuen met de eigenschap *mens* een deelverzameling is van de verzameling individuen met de eigenschap *sterfelijk*. De tweede regel kan men dan lezen als de constatering dat het individu *Socrates* tot de verzameling individuen met de eigenschap *mens* behoort. Een soortgelijke lezing kan men aan de derde regel toekennen. Door deze wiskundige interpretatie te geven aan de redenering wordt het zonneklaar dat deze geldig is. In figuur 8.1 is een en ander in de vorm van een Venn-diagram weergegeven. De ovalen daarin stellen verzamelingen voor en punten elementen daarvan. Uit het diagram kan het volgende worden afgelezen:

(8.4) $\mathit{mensen} \subseteq \mathit{sterfelijken}$.
$\mathit{Socrates} \in \mathit{mensen}$.
$\therefore$ $\mathit{Socrates} \in \mathit{sterfelijken}$.

Deze wiskundige interpretatie is ook de gangbare, zoals we zullen zien in hoofdstuk 9.

We zijn al een eind op weg bij het bepalen welke ingrediënten de predicatenlogica zou moeten bezitten. Om nog meer greep te krijgen op de gewenste ingrediënten geven we nog een voorbeeld van een geldige redenering:



Figuur 8.1: De 'Socrates-redenering' als Venn-diagram.

(8.5) Alle informatici zijn mensen.

Sommige informatici vinden logica leuk.

$\therefore$ Sommige mensen vinden logica leuk.

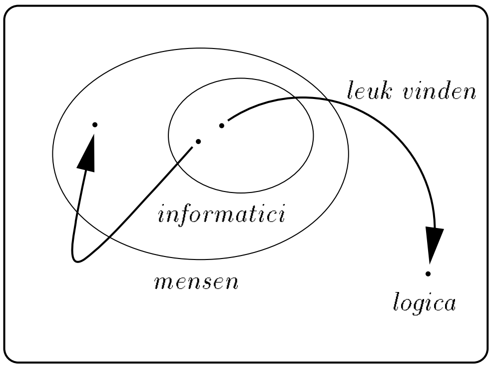
Deze redenering is een stuk lastiger om te doorzien. Met name de tweede en derde regel lijken ingewikkeld. Laten we eerst eens bekijken welke eigenschappen er in het spel zijn. We zien direct in dat *informaticus* en *mens* eigenschappen aanduiden. Hiervoor zullen we respectievelijk de letters I en M reserveren. In de tweede regel komen we *leuk vinden* tegen. Het is geen eigenschap die een individu kan bezitten, maar een *relatie* tussen individuen: de relatie die tussen de individuen x en y bestaat indien individu x individu y leuk vindt. We zullen deze relatie met de letter L aanduiden. Tenslotte is *logica* een naam voor het abstracte ding dat men 'logica' noemt. We zullen hiervoor de letter a gebruiken. De redenering kan dan als volgt worden weergegeven:

(8.6) Alle x die I zijn, zijn ook M.

Er is een x die I is en die in de relatie L tot a staat.

$\therefore$ Er is een x die M is en die in de relatie L tot a staat.

Net als voor de 'Socrates-redenering' kunnen we voor deze redenering een diagram tekenen. Behalve verzamelingen hebben we nu ook wiskundige relaties nodig. In figuur 8.2 hebben we de relatie *leuk vinden* weergegeven door een pijl te tekenen tussen de dingen waartussen deze relatie geldt. De oriëntatie van de pijl geeft daarbij de 'richting' van het leuk vinden aan: x vindt y leuk impliceert namelijk niet automatisch het omgekeerde. De rechter pijl in het diagram geeft dus aan dat een of andere informaticus *logica* leuk vindt, de linker pijl duidt aan dat een andere informaticus een of ander mens leuk vindt.



Figuur 8.2: Het diagram behorende bij de 'logica-redenering'.

In dit voorbeeld hebben we te maken met een relatie tussen twee individuen. Dit behoeft echter niet altijd het geval te zijn. Het is niet moeilijk om voorbeelden van relaties tussen drie of meer individuen te bedenken. Neem bijvoorbeeld de relatie die tussen drie individuen x, y en z bestaat, indien x van y naar z fietst. Al dit soort relaties zullen we in de predicatenlogica gebruiken en *predicaten* noemen. Een predicaat kan dus een eigenschap of een relatie zijn.

Een vraag die we nog moeten beantwoorden luidt: 'Wat is de precieze rol van *alle* en *er is een*?' Deze twee uitdrukkingen noemen we in de logica *kwantoren*. De eerste, *alle*, wordt de *universele kwantor* genoemd. Deze wordt gebruikt om uit te drukken dat voor alle individuen een bepaalde bewering geldig is. De eerste regel van (8.3) dient te worden gelezen als:

(8.7) Voor alle x geldt: als x een M is, dan is x ook S.

De tweede kwantor, *er is een*, noemt men de *existentiële kwantor*. Deze drukt uit dat er een individu bestaat dat een bepaalde eigenschap heeft. Men kan de tweede regel van (8.6) als volgt lezen:

(8.8) Er bestaat een x waarvoor geldt: x is een I en x staat in de relatie L tot a.

Merk op hoe in (8.7) en (8.8) de logische connectieven *als ... dan* en *en* te voorschijn zijn gekomen. We willen natuurlijk alle logische connectieven kunnen gebruiken in de predicatenlogica.

In het vervolg zullen we 'x is een M' en 'x staat in de relatie L tot a' noteren als $M(x)$, respectievelijk $L(x, a)$. We leggen er hierbij de nadruk op dat de volgorde van de argumenten in $L(x, a)$ van belang is: $L(a, x)$ betekent iets anders dan $L(x, a)$. Als individu a individu x leuk vindt, hoeft het nog niet zo te zijn dat x iets in a ziet. Ook voor de kwantoren zullen we een notatie gebruiken: $\forall x$ voor *voor alle x geldt*, en $\exists x$ voor *er bestaat een x waarvoor*. Nu kunnen we de redeneringen (8.1) en (8.5) weergeven in de predicatenlogica. De vertaling van (8.1) ziet er als volgt uit:

(8.9) $\forall x[M(x) \rightarrow S(x)]$
$M(a)$
$\therefore\ S(a)$

Redenering (8.5) kan als volgt in de predicatenlogica worden vertaald:

(8.10) $\forall x[I(x) \rightarrow M(x)]$
$\exists x[I(x) \wedge L(x, a)]$
$\therefore\ \exists x[M(x) \wedge L(x, a)]$

Een aspect van wiskundig redeneren dat we nog niet kunnen beschrijven met behulp van de tot nu toe geïntroduceerde begrippen, is het redeneren over *functies*. Wiskundigen doen uitspraken zoals:

(8.11) De functie $f : \mathbb{N} \rightarrow \mathbb{N}$ is monotoon stijgend.

Deze uitspraak bestaat eigenlijk uit twee delen:

1. f is een functie waarvan zowel het domein als het bereik gelijk is aan $\mathbb{N}$;
2. f is monotoon stijgend.

We kunnen deze uitspraken als volgt weergeven:

(8.12) Voor alle x geldt: als x een natuurlijk getal is, dan is $f(x)$ dat ook;

(8.13) en voor alle x en y geldt: als x en y natuurlijke getallen zijn en als $x \leq y$, dan $f(x) \leq f(y)$.

In deze omschrijving zijn x en y variabelen die voor individuen staan, en is *natuurlijk getal* een eigenschap. Ook de rol van $\leq$ zal duidelijk zijn: het is een relatie. Kenmerkend in (8.12) en (8.13) is het gebruik van de uitdrukking $f(x)$. Uitdrukkingen waarin functies voorkomen die een of meer argumenten hebben, mogen in de predicatenlogica worden gebruikt op dezelfde manier als namen. We kunnen (8.12) en(8.13) als volgt vertalen:

(8.14) $\forall x[N(x) \rightarrow N(f(x))]$,

(8.15) $\wedge\ \forall x \forall y[N(x) \wedge N(y) \wedge K(x, y) \rightarrow K(f(x), f(y))]$.

Hierin staat N voor de eigenschap *natuurlijk getal* en K voor de relatie $\leq$. In de wiskunde wordt deze onhandige notatie waarbij gebruik wordt gemaakt van N en K, niet gebruikt. Men schrijft:

$$\begin{aligned}(8.16)\quad & \forall x\,[x \in \mathbb{N} \rightarrow f(x) \in \mathbb{N}] \\ & \wedge\ \forall x \forall y[x \in \mathbb{N} \wedge y \in \mathbb{N} \wedge x \leq y \rightarrow f(x) \leq f(y)],\end{aligned}$$

of ook wel:

$$(8.17)\quad \forall x \in \mathbb{N}\,[f(x) \in \mathbb{N}] \wedge \forall x, y \in \mathbb{N}\,[x \leq y \rightarrow f(x) \leq f(y)].$$

De notatiewijze in (8.17) zullen we niet nader in beschouwing nemen. Over de notatie $x \in \mathbb{N}$ in (8.16) merken we op dat $\in$ hierin een relatie voorstelt en $\mathbb{N}$ een naam van een individu, namelijk de verzameling der natuurlijke getallen. In dat geval is $\mathbb{N}$ dus geen eigenschap.

Het laatste ingrediënt van de predicatenlogica dat we nog moeten beschrijven, is het *gelijkheidssymbool* $=$. In feite is $=$ een speciale relatie waarvan de betekenis op dezelfde wijze is gefixeerd als bijvoorbeeld die van de logische connectieven. Symbolen waarvan de betekenis is gefixeerd, noemt men *logische constanten.* Met behulp van het symbool $=$ kan de uitspraak:

(8.18) Functie $f : \mathbb{N} \rightarrow \mathbb{N}$ heeft een nulpunt.

worden vertaald als:

$$(8.19)\quad \forall x[N(x) \rightarrow N(f(x))] \wedge \exists x[N(x) \wedge f(x) = 0].$$

In de bovenstaande vertaling is het symbool 0 een naam van het natuurlijke getal passend bij het begrip nulpunt, nul dus. Merk op dat het consequenter zou zijn om te schrijven $=(f(x), 0)$ in plaats van $f(x) = 0$. Dit is echter hoogst ongebruikelijk en slecht leesbaar.

Als we alle nieuwe mogelijkheden die de predicatenlogica ons biedt, op een rijtje zetten, dan zien we dat we naast de logische connectieven kunnen beschikken over:

- aanduidingen voor predicaten (zowel eigenschappen als relaties), de zogenaamde *predicaatsymbolen*,
- *namen* van individuen, ook wel *individuele constanten* genaamd,
- aanduidingen voor functies, de zogenaamde *functiesymbolen*,
- *variabelen* waarmee individuen worden aangeduid,
- de *universele kwantor* $\forall$ en de *existentiële kwantor* $\exists$, en

- het *gelijkheidssymbool* of *identiteit* $=$.

Logische talen die hierover beschikken, noemt men *eerste-ordetalen.* Deze naamgeving heeft betrekking op het feit dat men kan kwantificeren (met behulp van $\forall$ of $\exists$) over individuen. Individuen worden namelijk beschouwd als eerste-orde-objecten. Voorbeelden van hogere-orde-objecten zijn predicaten en functies. Er bestaan ook hogere-ordetalen. Hierin kan men dus ook kwantificeren over hogere-orde-objecten. Deze talen vallen buiten het kader van dit boek.

8.2 Eerste-ordetalen

Het alfabet van een eerste-ordetaal valt uiteen in een *logisch alfabet* en een *niet-logisch alfabet.* Brengen we in herinnering datgene wat we in de vorige paragraaf hebben gezien, dan bevat de volgende definitie geen verrassingen.

8.2.1 DEFINITIE **Alfabet eerste-ordetaal**
Het alfabet *van een eerste-ordetaal bevat de volgende symbolen:*

1. logische symbolen *(ook* constanten *genaamd):*
    - *(a)* variabelen*:* $x_0, x_1, x_2, \ldots$ *.*
    - *(b)* connectieven*:* $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ *.*
    - *(c)* kwantoren*:* $\forall$, $\exists$ *.*
    - *(d)* gelijkheidssymbool *(optioneel):* $=$ *.*
    - *(e) haakjes:* (,) *.*
    - *(f) komma:* , *.*
2. niet-logische symbolen*:*
    - *(a)* predicaatsymbolen *(optioneel):* $P_0, P_1, \ldots$ *, waarbij* P_i *een* m_i*-plaatsig predicaatsymbool voorstelt* $(m_i > 0)$*.*
    - *(b)* functiesymbolen *(optioneel):* $f_0, f_1, \ldots$ *, waarbij* f_j *een* n_j*-plaatsig functiesymbool voorstelt* $(n_j > 0)$*.*
    - *(c)* namen *(optioneel):* $c_0, c_1, \ldots$ *.*

De bij de predicaatsymbolen en functiesymbolen gebruikte terminologie n-plaatsig *heeft betrekking op het aantal* argumenten *dat het predicaatsymbool of functiesymbool moet hebben.*

	$\mathcal{PL}$	$\mathcal{PL}^{=}$	$\mathcal{PL}^{f}$	$\mathcal{PL}^{f=}$
gelijkheid $=$	neen	ja	neen	ja
predicaatsymbolen $P_0, P_1, \ldots$	ja	ja	ja	ja
functiesymbolen $f_0, f_1, \ldots$	neen	neen	ja	ja
namen $c_0, c_1, \ldots$	ja	ja	ja	ja

Tabel 8.1: De verschillende verschijningsvormen van de predicatenlogica.

Zoals uit de definitie blijkt, behoeft het alfabet van een eerste-ordetaal niet alle soorten symbolen te bevatten. In het alfabet kunnen bijvoorbeeld predicaatsymbolen en namen voorkomen, terwijl functiesymbolen ontbreken. Het is zelfs mogelijk dat zowel predicaatsymbolen, als functiesymbolen en namen totaal ontbreken in het alfabet. Dit is echter een vreemde situatie, zodat we in het vervolg zullen aannemen dat een eerste-ordetaal minstens één predicaat- of functiesymbool bevat. Het alfabet van een eerste-ordetaal bevat echter in elk geval, op het gelijkheidssymbool na, alle logische constanten. Het gelijkheidssymbool $=$ kan voorkomen in het alfabet, maar dat is niet noodzakelijk.

Propositiesymbolen zou men kunnen opvatten als 0-plaatsige predicaatsymbolen, dus als predicaatsymbolen zonder argumenten. Volgens bovenstaande definitie zijn 0-plaatsige predicaatsymbolen echter niet toegestaan. Dit is geen wezenlijke beperking, zodat men de predicatenlogica kan beschouwen als een uitbreiding van de propositielogica. Anders gezegd, de propositielogica is een onderdeel van de predicatenlogica.

8.2.2 VOORBEELD De zuivere predicatenlogica.
De zogenaamde *zuivere predicatenlogica* bevat de volgende symbolen:

- de logische symbolen, behalve het gelijkheidssymbool,
- predicaatsymbolen: $P_0, P_1, \ldots$,
- namen: $c_0, c_1, \ldots$.

De zuivere predicatenlogica bevat dus geen functiesymbolen. In het vervolg zullen we de zuivere predicatenlogica kortweg *predicatenlogica* noemen en aanduiden met $\mathcal{PL}$. Een overzicht van de verschillende vormen van predicatenlogica treft men aan in tabel 8.1. ∎

8.2.3 VOORBEELD De taal van de rekenkunde.
Een eerste-ordetaal voor de rekenkunde, waarin optellen en vermenigvuldigen mogelijk is, bevat naast de logische constanten de volgende symbolen:

- het gelijkheidssymbool: $=$.
- de functiesymbolen: $\mathbf{s}$, $+$, $\cdot$.
- de naam: $\mathbf{0}$.

De bedoelde interpretatie van deze symbolen is (uiteraard) als volgt: $\mathbf{s}$ staat voor de opvolgerfunctie die van een natuurlijk getal de opvolger bepaalt, $+$ staat voor de operatie optelling, $\cdot$ voor de vermenigvuldiging en $\mathbf{0}$ voor het natuurlijke getal nul. Uit deze omschrijving volgt dat $\mathbf{s}$ een éénplaatsig functiesymbool is en dat $+$ en $\cdot$ tweeplaatsig zijn. ∎

In een eerste-ordetaal onderscheidt men twee syntactische categorieën: die van de *termen* en die van de *formules*. De verzameling van termen wordt genoteerd als *TERM* en de verzameling van alle formules als *FORM*. Beide categorieën worden inductief gedefinieerd in de volgende definities.

8.2.4 DEFINITIE **Term**
De verzameling TERM is de kleinste verzameling die voldoet aan:

1. $x_i \in TERM$ *voor alle* $i \in \mathbb{N}$.
2. $c \in TERM$ *voor alle namen* c.
3. *Als* f *een* n*-plaatsig functiesymbool is en als* $t_1, \ldots, t_n \in TERM$, *dan* $f(t_1, \ldots, t_n) \in TERM$.

De verzameling $\{x_i \mid i \in \mathbb{N}\}$ *van alle variabelen wordt aangeduid met VAR.*

8.2.5 VOORBEELD Termen.

1. Voorbeelden van termen uit de zuivere predicatenlogica zijn: c_0, x_3, c_{13}. Namen en variabelen zijn de enige termen, aangezien de zuivere predicatenlogica geen functiesymbolen heeft.
2. Voorbeelden van termen uit de taal van de rekenkunde zijn: $\mathbf{s}(\mathbf{0})$, $\mathbf{s}(\mathbf{s}(\mathbf{0}))$ (die respectievelijk voor de getallen één en twee staan), $+(x,y)$, $\cdot(x,y)$ en $\mathbf{s}(+(x, \cdot(\mathbf{0},y)))$. ∎

8.2.6 DEFINITIE **Formule**
De verzameling FORM is de kleinste verzameling die voldoet aan:

1. *Als* P *een* n*-plaatsig predicaatsymbool is en* $t_1, \ldots, t_n \in TERM$, *dan* $P(t_1, \ldots, t_n) \in FORM$.
2. *Als* $t_1, t_2 \in TERM$, *dan* $(t_1 = t_2) \in FORM$.

3. *Als* $A, B \in FORM$,
   dan $\neg A$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$, $(A \leftrightarrow B) \in FORM$.

4. *Als* $A \in FORM$ *en* $x \in VAR$, *dan* $\forall x A, \exists x A \in FORM$.

Formules van de vorm $P(t_1, \ldots, t_n)$ *of* $(t_1 = t_2)$ *worden* atomen *of* atomaire formules *genoemd. De verzameling atomen wordt genoteerd als* $ATOM$.

Literalen *zijn formules van de vorm* A *of* $\neg A$ *waarbij* $A \in ATOM$. *De verzameling literalen wordt genoteerd als* LIT.

8.2.7 VOORBEELD Formules.

1. Voorbeelden van formules uit de zuivere predicatenlogica zijn: $P_1(x_3, c_2)$, $(P_1(c_0, x_1) \rightarrow P_4(c_1, c_0, x_2, c_0))$ en $\forall x_0(\neg P_1(x_0, x_3) \wedge \exists x_1 P_4(x_0, x_1, x_3, x_1))$ (aangenomen dat P_1 tweeplaatsig en P_4 vierplaatsig is).

2. Voorbeelden van formules uit de taal van de rekenkunde zijn: $(\mathbf{0} = x_3)$, $\forall x_0 \neg(\mathsf{s}(x_0) = \mathbf{0})$ en $\forall x_0 \forall x_1(+(x_0, \mathsf{s}(x_1)) = \mathsf{s}(+(x_0, x_1)))$. ■

In de propositielogica hebben we de metavariabele $\star$ geïntroduceerd om tweeplaatsige connectieven mee aan te duiden. Deze metavariabele zullen we ook in de predicatenlogica gebruiken. Daarnaast zullen we gebruik maken van de metavariabele Q om kwantoren mee aan te duiden.

8.2.8 DEFINITIE **De metavariabele** Q

Het symbool Q *is een metavariabele waarvan de waarde altijd een kwantor is; met andere woorden* $\mathsf{Q} \in \{\forall, \exists\}$.

8.2.9 DEFINITIE **Subformule en echte subformule**

1. *Een formule* A *is een* subformule *van de formule* F *als één van de vier onderstaande voorwaarden geldt.*
   (a) $A = F$.
   (b) $F = \neg C$ *en* A *is een subformule van* C.
   (c) $F = (C \star D)$ *en* A *is een subformule van* C *of* D.
   (d) $F = \mathsf{Q}xC$ *en* A *is een subformule van* C.

2. *Een formule* A *is een* echte subformule *van de formule* F *als* A *een subformule is van* F *en* $A \neq F$.

Evenals in de propositielogica (zie §2.5), zullen we in de notatie van formules uit $FORM$ sommige haakjes weglaten. Verder noteren we namen, variabelen, functie- en predicaatsymbolen bij voorkeur zonder indices. In de taal van de rekenkunde zullen we de symbolen $+$ en $\cdot$ tussen hun argumenten plaatsen (de zogenaamde *infixnotatie*). Tenslotte zullen we ook de haakjes [,], {, en } gebruiken.

8.2.10 VOORBEELD Notatieconventies.

1. Namen: $a, b, c, \ldots$.
2. Variabelen: $x, y, z, u, \ldots$.
3. Functiesymbolen: $f, g, h, \ldots$.
4. Predicaatsymbolen: $P, Q, R, \ldots$.
5. $P(a, x) \rightarrow Q(b, a, y, a)$ staat bijvoorbeeld voor:
   $(P_1(c_0, x_1) \rightarrow P_4(c_1, c_0, x_2, c_0))$.
6. $\forall x[\neg P(x, y) \wedge \exists z Q(x, z, y, z)]$ staat bijvoorbeeld voor:
   $\forall x_0(\neg P_1(x_0, x_3) \wedge \exists x_1 P_4(x_0, x_1, x_3, x_1))$.
7. $\forall x \forall y[x + \mathrm{s}(y) = \mathrm{s}(x + y)]$ staat bijvoorbeeld voor:
   $\forall x_0 \forall x_1(+ (x_0, \mathrm{s}(x_1)) = \mathrm{s}(+ (x_0, x_1)))$.

De kwantoren $\forall$ en $\exists$ bezitten dezelfde prioriteit als het connectief $\neg$. ■

Tot slot willen we wijzen op de verschillende betekenissen van het =-teken. Als symbool van het alfabet van een eerste-ordetaal moet het louter als symbool worden gezien. Als zodanig kan het dan voorkomen in een formule zoals $(x_7 = x_9)$. Een ander gebruik dat we van het symbool = maken, is in metatalige uitspraken als $F = (P(x) \rightarrow Q(x))$. Hiermee wordt uitgedrukt dat de metavariabele F de waarde $(P(x) \rightarrow Q(x))$ heeft, dus dat F gelijk is aan de formule $(P(x) \rightarrow Q(x))$. Men zou voor deze verschillende betekenissen van = verschillende tekens kunnen gebruiken. Dit zullen we echter niet doen. De context maakt steeds duidelijk in welke betekenis het symbool = wordt gebruikt.

8.3 Inductie en recursie

In de propositielogica kan men door middel van structurele inductie over de formules in *PROP* bewijzen dat een eigenschap P geldt voor alle formules in *PROP*. Dit inductiebeginsel kan op vrijwel identieke wijze geformuleerd worden voor de verzamelingen *TERM* en *FORM*.

Evenals in de propositielogica kunnen er op recursieve wijze functies worden gedefinieerd met als domein *TERM* of *FORM* en als bereik een of andere verzameling W.

De volgende twee stellingen hebben deze recursieve definities als onderwerp. Zij garanderen het bestaan van een unieke functie $G : TERM \rightarrow W$ of $G : FORM \rightarrow W$ indien aan bepaalde recurrente betrekkingen met randvoorwaarden is voldaan. Het bewijs van deze stellingen laten we achterwege.

8.3.1 STELLING **Recursieve definities van functies over** *TERM*
Zij gegeven een niet-lege verzameling W en voor ieder n-plaatsig functiesymbool f een functie:

$$H_f : W^n \to W.$$

Dan heeft het stelsel recurrente betrekkingen

$$G(f(t_1,\ldots,t_n)) = H_f(G(t_1),\ldots,G(t_n)) \qquad (t_1,\ldots,t_n \in \mathit{TERM})$$

met als randvoorwaarde dat voor alle variabelen x en constanten c zowel $G(x)$ als $G(c)$ éénduidig bepaalde elementen van W zijn, precies één oplossing $G : \mathit{TERM} \to W$ die aan alle $t \in \mathit{TERM}$ een unieke functiewaarde $G(t)$ toekent.

8.3.2 STELLING **Recursieve definities van functies over** *FORM*
Zij gegeven een niet-lege verzameling W en een zevental functies:

$$\begin{array}{lcll} H_\neg & : & W \to W, & \\ H_\star & : & W \times W \to W, & \quad en \\ H_{\mathsf{Q}} & : & W \times \mathbb{N} \to W. & \end{array}$$

Dan heeft het stelsel recurrente betrekkingen

$$\begin{array}{rcl} G(\neg A) & = & H_\neg(G(A)), \\ G(A \star B) & = & H_\star(G(A), G(B)), \\ G(\mathsf{Q}x_iA) & = & H_{\mathsf{Q}}(G(A), i) \qquad (A, B \in \mathit{FORM} \text{ en } i \in \mathbb{N}), \end{array}$$

met als randvoorwaarde dat $G(P)$ voor alle atomen $P \in \mathit{ATOM}$ een éénduidig bepaald element van W is, precies één oplossing $G : \mathit{FORM} \to W$ die aan alle $F \in \mathit{FORM}$ een unieke functiewaarde $G(F)$ toekent.

In het onderstaande doen wij een beroep op het begrip *machtsverzameling* uit de verzamelingenleer. Voor de lezer die niet vertrouwd is met dit begrip, volgt hier de definitie: de machtsverzameling van een verzameling V, notatie $\mathcal{P}V$, is gedefinieerd als de verzameling van alle deelverzamelingen van V, ofwel $\{W \mid W \subseteq V\}$. De $\mathcal{P}$ staat hier voor de Engelse benaming *powerset* of het Duitse *Potenzmenge*.

Als voorbeeld van een recursief gedefinieerde functie beschouwen we in de onderstaande definitie de functie $VV : \mathit{TERM} \cup \mathit{FORM} \to \mathcal{P}\,\mathit{VAR}$, die van een gegeven term of formule de verzameling van *vrije variabelen* bepaalt. Dit zijn variabelen die niet bij een kwantor horen. In de formule $\forall x A(x,y) \to \exists z B(x,z)$, bijvoorbeeld, hoort de y in $A(x,y)$ bij geen enkele kwantor. Men zegt dat deze variabele y *vrij voorkomt* of een *vrije variabele* is. De x in $A(x,y)$ komt niet vrij voor, want deze hoort bij $\forall x$. Men noemt dit een *gebonden variabele*. Voor de variabele x in $B(x,z)$ geldt dat deze vrij voorkomt: deze x wordt niet gebonden door $\forall x$. Vanwege de prioriteitsregels hoort $\forall x$ alleen bij $A(x,y)$. Men noemt $A(x,y)$ het *bereik* van de kwantor $\forall x$. De variabele z wordt gebonden door $\forall z$ aangezien $B(x,z)$ in het bereik van $\forall z$ voorkomt. Dit alles betekent dat er in

$\forall x A(x,y) \rightarrow \exists z B(x,z)$ twee vrije variabelen voorkomen: y en x. De begrippen *vrij* en *gebonden* hebben dus betrekking op een *voorkomen* (optreden) van een variabele en niet op de variabele zelf. Merk op dat we $\forall x$ en $\forall z$ kwantoren hebben genoemd (in plaats van alleen de $\forall$). Dit is gebruikelijk aangezien een kwantor altijd optreedt in combinatie met een variabele.

8.3.3 DEFINITIE **Vrije variabelen**

1. *De verzameling van* vrije variabelen *van een term $t \in TERM$, notatie $VV(t)$, is recursief gedefinieerd door:*

$$\begin{array}{rcl} VV(x_i) & = & \{x_i\}, \\ VV(c) & = & \emptyset, \\ VV(f(t_1,\ldots,t_n)) & = & VV(t_1) \cup \ldots \cup VV(t_n). \end{array}$$

2. *De verzameling van* vrije variabelen *van een formule $F \in FORM$, notatie $VV(F)$, is recursief gedefinieerd door:*

$$\begin{array}{rcl} VV(P(t_1,\ldots,t_n)) & = & VV(t_1) \cup \ldots \cup VV(t_n), \\ VV(t_1 = t_2) & = & VV(t_1) \cup VV(t_2), \\ VV(\neg A) & = & VV(A), \\ VV(A \star B) & = & VV(A) \cup VV(B), \\ VV(\mathsf{Q} x_i A) & = & VV(A) \setminus \{x_i\}. \end{array}$$

Het is duidelijk dat door bovenstaande recurrente betrekkingen met randvoorwaarden een functie $VV : TERM \cup FORM \rightarrow \mathcal{P}\,VAR$ wordt gedefinieerd.

8.3.4 DEFINITIE **Gesloten termen en formules**
Een term $t \in TERM$ wordt een gesloten *term genoemd, indien $VV(t) = \emptyset$. Evenzo wordt een formule $F \in FORM$ een* gesloten *formule genoemd, indien $VV(F) = \emptyset$. Een gesloten formule wordt een* (vol)zin *genoemd. Termen of formules die niet gesloten zijn worden* open *genoemd. De verzameling van gesloten termen zullen we in het vervolg aanduiden met GTERM en die van volzinnen met ZIN.*

8.3.5 VOORBEELD Vrije en gebonden variabelen.

1. De term $t = f(x,a)$ bevat de vrije variabele x, dus $VV(t) = \{x\}$.

2. $VV(\forall x P(x,y)) = \{y\}$. De variabele x is gebonden.

3. $VV(\forall x \exists y P(x,y) \rightarrow P(x,z)) = \{x, z\}$. Let op: de variabele x in $P(x,z)$ is niet gebonden en dus vrij. ■

In het laatste voorbeeld hierboven kwam de variabele x zowel gebonden als vrij voor in één formule. Aangezien dit verwarring kan geven, verdient het aanbeveling om dit zoveel mogelijk te vermijden door in formules de gebonden variabelen anders te kiezen dan de vrije. Dit verandert de betekenis van de formule niet, zoals we in stelling 10.2.7 zullen zien. De intuïtie hierachter is dat gebonden variabelen slechts plaatsen in een formule aan een kwantor binden.

8.4 Substitutie

Vaak is het nodig termen te substitueren voor variabelen in termen of formules. Als s en t termen zijn en x een variabele, dan duidt $s[x:=t]$ het resultaat aan van de vervanging van alle voorkomens van x in de term s door de term t. Hierbij is het toegestaan dat $x \in VV(t)$. Op analoge wijze, als F een formule is, dan duidt $F[x:=t]$ het resultaat aan van de vervanging van alle vrije voorkomens van x door de term t. De resultaten van deze substituties zijn respectievelijk weer een term en een formule.

8.4.1 Definitie **Substitutie in termen**
Zij $t \in TERM$ en x een variabele. De substitutie-*operatie* $[x:=t]$ *is als volgt recursief gedefinieerd voor termen:*

$$\begin{array}{rcl} y[x:=t] & = & \begin{cases} t & \text{als } y = x, \\ y & \text{als } y \neq x, \end{cases} \\ c[x:=t] & = & c, \\ f(t_1,\ldots,t_n)[x:=t] & = & f(t_1[x:=t],\ldots,t_n[x:=t]). \end{array}$$

Stelling 8.3.1 garandeert dat voor iedere variabele x en term t de functie $[x:=t]$: $TERM \rightarrow TERM$ uniek is bepaald.

Substitutie in formules is een delicate aangelegenheid. Dit heeft met name betrekking op substituties in formules van de vorm $\mathsf{Q}yA$. Laten we de verschillende gevallen die zich kunnen voordoen, eens bekijken:

1. Wat zou men bedoelen met $(\forall x P(x,y))[y:=f(z)]$? Het antwoord ligt voor de hand: vervang in de formule $\forall x P(x,y)$ overal y door de term $f(z)$. Dit levert $\forall x P(x, f(z))$.
2. De substitutie $(\forall x P(x,y))[y:=f(x)]$ is problematisch. Directe substitutie als in het vorige geval zou opleveren $\forall x P(x, f(x))$. Het probleem schuilt

in het feit dat de x in $f(x)$ na substitutie gebonden wordt door $\forall x$. Dit is onwenselijk daar in de oorspronkelijke formule $\forall x P(x, y)$ een dergelijke binding tussen de x en de y niet bestond.

Om het probleem op te lossen doen we het volgende: eerst *herbenoemen* we de variabele x in $\forall x P(x, y)$, en daarna voeren we pas de substitutie uit. Als we x herbenoemen als w, dan is het resultaat van de substitutie $\forall w P(w, f(x))$.

Dit herbenoemen is volkomen legaal omdat de naamgeving van gebonden variabelen er niet toe doet. De formules $\forall x P(x, y)$ en $\forall w P(w, y)$ hebben dezelfde betekenis (zie ook hoofdstuk 10, stelling 10.2.7).

3. Het laatste geval is de substitutie $(\forall x P(x, y))[x := f(z)]$. Directe substitutie zou leveren $\forall f(z)\, P(f(z), y)$. Dit is echter geen formule. Ook $\forall x P(f(z), y)$ of $\forall z P(f(z), y)$ zijn geen geschikte kandidaten. Bij nader inzien is het eigenlijk vreemd om een gebonden variabele te willen laten bewerken door een substitutie! Het is immers 'toevallig' dat de gebonden variabele in de voorbeeldformule x heet. Als deze variabele w zou heten, dan zou het resultaat van de substitutie $\forall w P(w, y)$ zijn.

   Aangezien het redelijk is om te eisen dat het resultaat van een substitutie onafhankelijk is van de toevallige keuze van de gebonden variabelen, laten we een substitutie alleen op vrij voorkomende variabelen werken. Het resultaat van de substitutie is dus $\forall x P(x, y)$.

Een formalisering van deze drie gevallen vindt men in dezelfde volgorde terug in het laatste geval van de volgende definitie.

8.4.2 DEFINITIE **Substitutie in formules**
Zij $t \in TERM$ en x een variabele. De substitutie-*operatie* $[x := t]$ *is als volgt recursief gedefinieerd voor formules:*

Atomaire formules

$$\begin{aligned} P(t_1, \ldots, t_n)[x := t] &= P(t_1[x := t], \ldots, t_n[x := t]), \\ (t_1 = t_2)[x := t] &= (t_1[x := t] = t_2[x := t]). \end{aligned}$$

Niet-atomaire formules

$$\begin{aligned} (\neg A)[x := t] &= \neg(A[x := t]), \\ (A \star B)[x := t] &= (A[x := t] \star B[x := t]), \end{aligned}$$

$$(\mathsf{Q}yA)[x{:=}t] \quad = \quad \begin{cases} \mathsf{Q}y(A[x{:=}t]) & \textit{als } y \neq x \textit{ en } [\ y \notin VV(t) \\ & \quad \textit{of } x \notin VV(A)\ ], \\ \mathsf{Q}z(A[y{:=}z][x{:=}t]) & \textit{als } y \neq x,\ y \in VV(t), \\ & \quad \textit{en } x \in VV(A), \\ & \quad \textit{met } z \textit{ de eerste} \\ & \quad \textit{variabele waarvoor} \\ & \quad z \notin VV(A) \cup VV(t), \\ \mathsf{Q}yA & \textit{als } y = x. \end{cases}$$

De zinsnede 'eerste variabele ...' heeft betrekking op de ordening van de variabelen op hun index: $x_0, x_1, \ldots$.

Door de complexe vorm van de definitie van $(\mathsf{Q}yA)[x{:=}t]$ kan stelling 8.3.2 niet worden aangeroepen om te garanderen dat voor iedere variabele x en term t de functie $[x{:=}t] : \mathit{FORM} \to \mathit{FORM}$ uniek is bepaald. Hiertoe is een meer algemene formulering van deze stelling nodig. In de praktijk is echter in te zien dat de definitie 'werkt'.

8.4.3 VOORBEELD Substituties.

1. $(\exists x(x > y))[x{:=}y] = \exists x(x > y)$.

   Niet geldt dat $(\exists x(x > y))[x{:=}y] = \exists y(y > y)$ aangezien x een gebonden variabele is.

2. $(\exists x(x > y))[y{:=}x] = \exists z((x > y)[x{:=}z][y{:=}x]) = \exists z((z > y)[y{:=}x]) = \exists z(z > x)$.

   Niet geldt dat $(\exists x(x > y))[y{:=}x] = \exists x(x > x)$, aangezien de variabele x voorkomt in x. De variabele x wordt eerst herbenoemd tot z. De bovenstaande definitie schrijft voor dat z de eerste variabele is die niet vrij voorkomt in $x > y$ en x. Stelling 10.2.7 drukt echter uit dat de keuze van de variabele semantisch gezien niet uit maakt. De reden dat in de definitie toch een speciale variabele wordt gekozen, is gelegen in het feit dat we de uitkomst van de substitutie-operatie uniek bepaald willen hebben. Anders zou het geen functie zijn. In het vervolg zullen we impliciet aannemen dat de keuze van de nieuwe variabele in overeenstemming is met definitie 8.4.2. ■

Simultane substituties $[x_1{:=}t_1, \ldots, x_n{:=}t_n]$, waarbij n variabelen $(n \geq 2)$ tegelijkertijd vervangen worden, kunnen analoog gedefinieerd worden. Aan de hand van voorbeelden zullen we simultane substituties toelichten. Een nauwkeurige definitie zal worden gegeven in paragraaf 16.2.

8.4.4 Voorbeeld Simultane substituties.

1. $f(x,y)[x{:=}a, y{:=}b] = f(a,b)$.
2. Een simultane substitutie levert niet altijd hetzelfde resultaat op als de corresponderende herhaalde substitutie, zoals uit het volgende voorbeeld blijkt. Hierin is $[x{:=}y, y{:=}x]$ een simultane substitutie en $[x{:=}y][y{:=}x]$ de corresponderende herhaalde substitutie.
   (a) $(x = y)[x{:=}y, y{:=}x] = (y = x)$.
   (b) $(x = y)[x{:=}y][y{:=}x] = (y = y)[y{:=}x] = (x = x)$. ■

8.5 Vertalen van beweringen

In deze paragraaf zullen we een aantal voorbeelden geven van vertalingen van beweringen in de Nederlandse taal naar een eerste-ordetaal. Bij dit vertaalproces dient de lezer te bedenken dat de natuurlijke taal een veel groter uitdrukkingsvermogen heeft dan een eerste-ordetaal. Immers, een eerste-ordetaal is in de eerste plaats ontworpen voor het beschrijven van wiskundige redeneringen. Dit heeft tot gevolg dat niet-wiskundige beweringen vaak slechts moeizaam vertaald kunnen worden naar predicaatlogische formules.

8.5.1 Voorbeeld Eigenschappen van individuen en relaties tussen individuen.

1. Plato is een wijsgeer: $W(p)$.
2. Marie houdt van Jan: $H(m,j)$.
3. Marie zit tussen Klaas en Jan: $Z(k,m,j)$.
4. Marie houdt meer van Klaas dan van Jan: $M(m,k,j)$. ■

In de volgende voorbeelden wordt voor een aantal uitspraken het vertaalproces stapsgewijs weergegeven. Logische voegwoorden en kwantoren worden daarbij eerst omkaderd en daarna vervangen door de overeenkomstige logische connectieven en kwantoren.

8.5.2 Voorbeeld Vertalingen van beweringen en redeneringen.

1. $\boxed{\text{Alle}}$ schapen zijn wit,
   $\forall x[$ $\boxed{\text{als}}$ x is een schaap, $\boxed{\text{dan}}$ x is wit],
   $\forall x[$ x is een schaap $\rightarrow$ x is wit],
   $\forall x[S(x) \rightarrow W(x)]$.

2. $\boxed{\text{Er is een}}$ wit schaap,

   $\exists x[$ x is een schaap $\boxed{\text{en}}$ x is wit $]$,

   $\exists x[$ x is een schaap $\wedge$ x is wit $]$,

   $\exists x[S(x) \wedge W(x)]$.

   Bedenk dat de voor de hand liggende, maar foute vertaling

   $\exists x[S(x) \rightarrow W(x)]$

   minder uitdrukt dan bedoeld. Deze bewering is namelijk ook waar als er *geen* schapen bestaan. Dat is kennelijk niet de bedoeling. De eerste vertaling is echter alleen waar (en dit is bedoeld!) als er een individu a bestaat dat zowel een schaap, als wit is, dus als $S(a)$ en $W(a)$ beide waar zijn.

3. $\boxed{\text{Alles}}$ trekt alles aan,

   $\forall x[$ x trekt $\boxed{\text{alles}}$ aan $]$,

   $\forall x[\forall y($ x trekt y aan $)]$,

   $\forall x[\forall y T(x,y)]$,

   $\forall x \forall y T(x,y)$.

4. $\boxed{\text{Sommige}}$ docenten zijn kinderachtig,

   $\exists x[$ x is een docent $\boxed{\text{en}}$ x is kinderachtig $]$,

   $\exists x[$ x is een docent $\wedge$ x is kinderachtig $]$,

   $\exists x[D(x) \wedge K(x)]$.

5. $\boxed{\text{Iets}}$ doet me aan iets herinneren,

   $\exists x[$ x herinnert me aan $\boxed{\text{iets}}$ $]$,

   $\exists x[\exists y($ x herinnert me aan y $)]$,

   $\exists x[\exists y H(x,y)]$,

   $\exists x \exists y H(x,y)$.

6. $\boxed{\text{Sommige}}$ patienten vertrouwen alle doktoren,

   $\exists x[$ x is een patient $\boxed{\text{en}}$ x vertrouwt alle doktoren $]$,

   $\exists x[$ x is een patient $\wedge$ x vertrouwt $\boxed{\text{alle}}$ doktoren $]$,

   $\exists x[$ x is een patient $\wedge$ $\forall y($ $\boxed{\text{als}}$ y is een dokter $\boxed{\text{dan}}$ x vertrouwt y $)]$,

   $\exists x[$ x is een patient $\wedge$ $\forall y($ y is een dokter $\rightarrow$ x vertrouwt y $)]$,

   $\exists x[P(x) \wedge \forall y(D(y) \rightarrow V(x,y))]$.

7. Paardestaarten zijn dierestaarten,

   $\forall x\{$ $\boxed{\text{als}}$ x de staart van een paard is,
   $\boxed{\text{dan}}$ is x de staart van een dier $\}$,

   $\forall x\{$ x is de staart van $\boxed{\text{een}}$ paard $\to$
   x is de staart van $\boxed{\text{een}}$ dier $\}$,

   $\forall x\{$ $\exists y[$ y is een paard $\boxed{\text{en}}$ x is de staart van y $]$ $\to$
   $\exists y[$ y is een dier $\boxed{\text{en}}$ x is de staart van y $]\}$,

   $\forall x\{$ $\exists y[$ y is een paard $\wedge$ x is de staart van y $]$ $\to$
   $\exists y[$ y is een dier $\wedge$ x is de staart van y $]\}$,

   $\forall x\{\exists y[P(y) \wedge S(x,y)] \to \exists y[D(y) \wedge S(x,y)]\}$. $\blacksquare$

8.6 Opgaven

1. Geef een recursieve definitie van een functie BV die de verzameling van alle gebonden variabelen van een formule bepaalt.

2. Voer de volgende substituties uit:

   (a) $P(x, f(x))[x{:=}f(y)]$.

   (b) $P(x, f(x))[x{:=}f(x)]$.

   (c) $(\forall x P(x,y) \vee \exists y P(x,y))[y{:=}f(z)]$.

   (d) $(\forall x P(x,y) \vee \exists y P(x,y))[y{:=}g(x,y)]$.

3. Vertaal naar de predicatenlogica:

   (a) *Logici kunnen logisch redeneren.*

   (b) *Jan is geen logicus of hij kan logisch redeneren.*

   (c) *You (one) can fool all people sometimes, and you can fool some people all times, but nobody can fool all people all times.*

4. Vertaal de volgende redenering naar de predicatenlogica:

   People are prejudiced against anyone who is liked by someone they dislike.
   But nobody is prejudiced against himself.
   So people don't like anyone who dislikes them.

Hoofdstuk 9

Semantiek van de Predicatenlogica

In dit hoofdstuk behandelen we de semantiek van de predicatenlogica. Vergeleken met de semantiek voor de propositielogica is deze tamelijk gecompliceerd. In paragraaf 9.1 zullen we deze daarom op informele wijze introduceren. Daarna behandelen we in de paragrafen 9.2, 9.3 en 9.4 de semantische noties op formele wijze. Deze noties zijn *structuur*, *bedeling*, *interpretatie*, *model* en *logisch gevolg*.

9.1 Een informele introductie

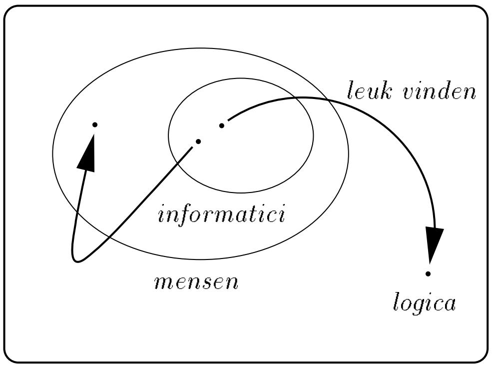
Evenals in de propositielogica kan men aan iedere formule van een eersteordetaal een waarheidswaarde toekennen, gegeven een zekere *interpretatie* van de symbolen in die taal. In het geval van de propositielogica volstonden valuaties die we op wiskundige wijze konden definiëren. Ook voor de predicatenlogica zoeken we een op wiskundige leest geschoeide semantiek. Hierbij zullen we eenvoudige verzamelingenleer nodig blijken te hebben. Aan de hand van een voorbeeld zullen we onderzoeken hoe een interpretatie voor de predicatenlogica eruit ziet. Beschouw de redenering:

(9.1) Alle informatici zijn mensen.
Sommige informatici vinden logica leuk.
$\therefore$ Sommige mensen vinden logica leuk.

en de bijbehorende vertaling (zie ook paragraaf 8.1):

(9.2) $\forall x[I(x) \rightarrow M(x)]$
$\exists x[I(x) \wedge L(x, a)]$
$\therefore\ \exists x[M(x) \wedge L(x, a)]$

De geldigheid van deze redenering kan worden ingezien met behulp van het bijbehorende diagram (zie figuur 9.1). Immers, als we aannemen dat



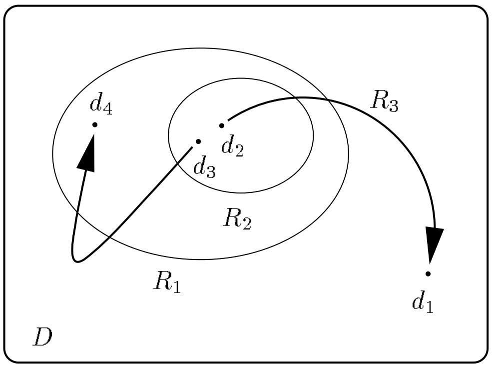
Figuur 9.1: Het diagram behorende bij de ‘logica-redenering’.

informatici $\subseteq$ *mensen* en dat er een object $o \in$ *informatici* bestaat zodanig dat o in de relatie *leuk vinden* staat tot het object *logica*, dan moet ook gelden dat er een object $o \in$ *mensen* bestaat zodanig dat o in de relatie *leuk vinden* staat tot het object *logica*.

Dit soort diagrammen waarin verzamelingen, elementen daarvan en relaties voorkomen, zullen dienen als basis voor de interpretatie van eerste-ordetalen. Zo'n diagram stelt een wiskundige *structuur* voor. Een structuur wordt gegeven door een verzameling elementen, het *domein* van de structuur genoemd, en relaties en functies daarop. Het domein van het diagram in figuur 9.1 is nog niet gespecificeerd. Kennelijk hadden we (misschien alle) concrete en abstracte objecten uit de wereld in onze gedachten, want mensen en het vak logica behoren tot dat domein. In dat domein zijn 4 elementen aangegeven door een punt. Als we aannemen dat het domein geen andere elementen bevat dan deze vier, dan is er nog steeds sprake van een structuur, zij het een eenvoudige. Voor dit moment zullen we gemakshalve deze aanname maken.

Verder zijn er speciale deelverzamelingen van het domein gegeven, namelijk de verzamelingen *mensen* en *informatici*, waarbij *informatici* $\subseteq$ *mensen*. De eerstgenoemde verzameling bevat 3 elementen, de tweede 2 elementen. Daarnaast is de relatie *leuk vinden* in het diagram getekend met behulp van pijlen. Deze relatie bestaat tussen 2 paren van objecten.

Als we nu abstraheren van onze gedachten aan mensen, informatici, leuk vinden en logica, dan kunnen we de essentie van de structuur weergeven zoals we hebben gedaan in figuur 9.2. Deze structuur heeft als domein de verzameling $D = \{d_1, d_2, d_3, d_4\}$, met daarop de twee deelverzamelingen $R_1 = \{d_2, d_3, d_4\}$ en $R_2 = \{d_2, d_3\}$, en de relatie $R_3 = \{(d_2, d_1), (d_3, d_4)\}$ bestaande uit de geordende paren (d_2, d_1) en (d_3, d_4) overeenkomend met de twee pijlen.



Figuur 9.2: De abstracte structuur voor het diagram uit figuur 9.1.

Om niet altijd plaatjes te hoeven tekenen, is er een notatiewijze voor structuren. De voorbeeldstructuur, die we $\mathcal{A}$ noemen, wordt genoteerd als een *tupel*:

$$\mathcal{A} = \langle D; R_1, R_2, R_3; d_1 \rangle.$$

Dit tupel bevat naast het domein net zoveel elementen als het niet-logisch alfabet van de betreffende taal, in dit geval M, I, L en a. De volgorde is hierbij van belang: R_1 dient als interpretatie van M, R_2 voor I, R_3 voor L, en d_1, tenslotte, voor a. Een structuur $\mathcal{A}$ legt dus een *verband* tussen de niet-logische symbolen van een eerste-ordetaal en bepaalde verzamelingen, relaties en elementen uit die structuur. Om dit verband expliciet te maken gebruiken we de volgende notatie:

$$M^{\mathcal{A}} = R_1, \ \ I^{\mathcal{A}} = R_2, \ \ L^{\mathcal{A}} = R_3, \ \ \text{en } a^{\mathcal{A}} = d_1.$$

Het is duidelijk dat beide premissen en ook de conclusie van de redenering waar zijn in $\mathcal{A}$. Immers,

$R_2 \subseteq R_1$ (alle informatici zijn mensen)

Er is een $o \in R_2$ (namelijk d_2) zodanig dat $(o, d_1) \in R_3$ (sommige logici vinden logica leuk)

$\therefore$ Er is een $o \in R_1$ (ook d_2) zodanig dat $(o, d_1) \in R_3$ (sommige mensen vinden logica leuk)

De structuur $\mathcal{A}$ is niet de enig mogelijke voor onze voorbeeldtaal. Als we uitgaan van ons oorspronkelijk idee dat het domein D van $\mathcal{A}$ uit alle concrete en abstracte objecten uit de wereld bestaat, krijgen we een structuur waarin R_1 de verzameling van alle mensen voorstelt, R_2 die van alle informatici, R_3 die van alle paren objecten (x, y) zodanig dat x vindt y leuk, en waarin d_1

het object logica voorstelt. Deze goed gedefinieerde structuur zou men, met betrekking tot de gegeven redenering, de *bedoelde structuur* kunnen noemen.

Er zijn ook hele andere structuren die op het eerste gezicht niets met de redenering te maken hebben. Beschouw bijvoorbeeld:

$$\mathcal{B} = \langle \mathbb{N}; S_1, S_2, \geq; 0 \rangle,$$

waarin S_1 de verzameling $\{0, 2, 4, \ldots\}$ van alle even natuurlijke getallen voorstelt, S_2 de verzameling $\{0, 4, 8, \ldots\}$ van alle viervouden, en $\geq$ de *groter dan of gelijk aan* ordening op $\mathbb{N}$. Zij vervolgens:

$$M^{\mathcal{B}} = S_1, \ \ I^{\mathcal{B}} = S_2, \ \ L^{\mathcal{A}} = \geq, \ \ \text{en } a^{\mathcal{A}} = 0.$$

Het is eenvoudig na te gaan dat de premissen en de conclusie van de voorbeeldredenering ook in deze structuur waar zijn. Dat brengt ons op de vraag of er voor de gegeven redenering structuren bestaan waarbij de premissen waar zijn, maar de conclusie onwaar. Het antwoord mag duidelijk zijn: neen. De redenering die we aan het begin van deze paragraaf gaven naar aanleiding van het diagram in figuur 9.1 laat zien dat dit onmogelijk is. Omdat zo'n structuur niet bestaat, is de redenering *geldig*. Een redenering is dus geldig indien iedere structuur die de premissen ervan waar maakt, ook de conclusie waar maakt.

Tot nu toe hebben we ons beperkt tot één- of tweeplaatsige predicaatsymbolen en namen. Indien we n-plaatsige predicaatsymbolen van een betekenis moeten voorzien, dan moeten we in de structuur een n-plaatsige relatie opnemen. Een n-plaatsige relatie is, wiskundig gezien, een verzameling van geordende n-tupels. Voor het geval $n = 2$ levert dat, zoals we hebben gezien, geordende paren op.

Als een eerste-ordetaal m-plaatsige functiesymbolen bevat, dan dienen structuren daarvoor m-plaatsige functies te bevatten. Wiskundig gezien zijn functies speciale verzamelingen. Zo kan men een éénplaatsige functie (bijvoorbeeld $f(x) = x + 1$ op het domein $\mathbb{N}$) opvatten als de verzameling van alle paren $(x, x + 1)$ waarbij $x \in \mathbb{N}$. In feite zijn functies dus speciale relaties: een m-plaatsige functie kan men opvatten als een speciale $(m + 1)$-plaatsige relatie.

Laten we nu onze bevindingen resumeren. Een *structuur* $\mathcal{A}$ voor een eerste-ordetaal L is tupel bestaande uit:

- een domein D van objecten,
- verzamelingen en relaties over D die als interpretatie dienen voor de predicaatsymbolen,
- functies op D die als interpretatie dienen voor de functiesymbolen, en

- elementen van D die als interpretatie dienen voor de namen van L.

We zullen een structuur $\mathcal{A}$ noteren als:

$$\mathcal{A} = \langle D; R_0, R_1, \ldots; g_0, g_1, \ldots; d_0, d_1, \ldots \rangle.$$

Hierin staat D voor het domein van $\mathcal{A}$, R_i voor een verzameling of relatie over D, g_j voor een functie over D, en d_k voor een element van D.

Met behulp van structuren kunnen we zinnen uit de predicatenlogica van een waarheidswaarde voorzien. Hierbij is de interpretatie van de logische connectieven gelijk aan die in de propositielogica. De betekenis van de kwantoren is in termen van het domein van een structuur uit te drukken. De universele kwantor drukt uit dat een bepaalde uitspraak waar is voor alle individuen uit het domein van de structuur, en de existentiële kwantor dat een uitspraak waar is voor tenminste één individu uit dat domein. In het geval van de formule $\forall x[I(x) \rightarrow M(x)]$ en voorbeeldstructuur $\mathcal{A}$ komt dit neer op: voor alle objecten $o \in D$ moet gelden als $o \in R_2$ dan $o \in R_1$. Maar dit is equivalent met $R_2 \subseteq R_1$, hetgeen ons vertrouwd voorkomt.

Als er formules met vrije variabelen in het spel zijn, moeten we over een methode beschikken om de vrije variabelen van een betekenis te voorzien. Hiervoor worden zogenaamde *bedelingen* gebruikt. Een bedeling bindt elementen uit het domein van een gegeven structuur aan variabelen. Zonder bedelingen zouden we niet in staat zijn om waarheidswaarden toe te kennen aan formules die vrije variabelen bevatten. Nemen we bijvoorbeeld de formule $M(x)$ en voorbeeldstructuur $\mathcal{A}$, dan kunnen we pas iets zeggen over de waarheid van $M(x)$ als we weten waarnaar de variabele x verwijst. Nemen we een bedeling die d_4 bindt aan x, dan is $M(x)$ waar met betrekking tot $\mathcal{A}$ en de gegeven bedeling, aangezien $d_4 \in R_1$. Nemen we echter een bedeling die d_1 bindt aan x, dan wordt $M(x)$ onwaar. De semantiek van de predicatenlogica maakt dus gebruik van structuren en bedelingen. Een combinatie van een structuur en een bedeling wordt een *interpretatie* genoemd. In paragraaf 9.3 zullen we uiteenzetten hoe in het algemeen de waarheidswaarde van een formule kan worden bepaald met betrekking tot een gegeven interpretatie.

Of een redenering waarin formules met vrije variabelen voorkomen, *logisch geldig* is wordt nu niet gedefinieerd in termen van structuren maar interpretaties. Een redenering is logisch geldig, indien iedere interpretatie die de premissen van die redenering waar maakt, ook de conclusie ervan waar maakt. Deze definitie is geheel analoog aan die voor redeneringen in de propositielogica. Het verschil is dat in het geval van de predicatenlogica interpretaties veel complexere wiskundige objecten zijn dan interpretaties (valuaties) in de propositielogica.

9.2 Structuren, bedelingen en interpretaties

In deze paragraaf geven we formele definities van de begrippen *structuur*, *bedeling* en *interpretatie*.

9.2.1 DEFINITIE **Structuur**
Laat L een eerste-ordetaal zijn die de predicaatsymbolen $P_0, P_1, \ldots$*, de functiesymbolen* $f_0, f_1, \ldots$ *en de namen* $c_0, c_1, \ldots$ *bevat. Een* structuur $\mathcal{A}$ *voor de taal L is dan gedefinieerd als een tupel:*

$$\mathcal{A} = \langle D; R_0, R_1, \ldots; g_0, g_1, \ldots; d_0, d_1, \ldots \rangle.$$

Hierin is:

1. *D een niet-lege verzameling, het* domein *van de structuur* $\mathcal{A}$ *genaamd; D wordt ook wel genoteerd als* $|\mathcal{A}|$.
2. R_i *een* m_i*-plaatsige relatie over D, dus* $R_i \subseteq D^{m_i}$*, indien* P_i *een* m_i*-plaatsig predicaatsymbool is* $(i \in \mathbb{N})$.
3. g_j *een* n_j*-plaatsige functie op D, dus* $g_j : D^{n_j} \to D$*, indien* f_j *een* n_j*-plaatsig functiesymbool is* $(j \in \mathbb{N})$.
4. d_k *een element van D, dus* $d_k \in D$ $(k \in \mathbb{N})$.

De relaties R_i*, de functies* g_j *en de elementen* d_k *noemt men respectievelijk de* relaties*, de* functies *en de* constanten *van* $\mathcal{A}$.

Aan de niet-logische symbolen van L worden als volgt betekenissen, ofwel denotaties*, toegekend in* $\mathcal{A}$*:*

1. *De denotatie van het predicaatsymbool* P_i *is de relatie* R_i $(i \in \mathbb{N})$. *Dit wordt genoteerd als* $P_i^{\mathcal{A}} = R_i$.
2. *De denotatie van het functiesymbool* f_j *is de functie* g_j $(j \in \mathbb{N})$. *Notatie:* $f_j^{\mathcal{A}} = g_j$.
3. *De denotatie van de naam* c_k *is de constante* d_k $(k \in \mathbb{N})$. *Notatie:* $c_k^{\mathcal{A}} = d_k$.

In het bovenstaande hebben we aangenomen dat de taal L aftelbaar oneindig veel predicaatsymbolen, functiesymbolen en namen bevat (dit is bijvoorbeeld het geval in de taal van de zuivere predicatenlogica). Al het gedefinieerde geldt echter ook in die gevallen waarin sprake is van eindig *veel niet-logische symbolen.*

De toevoeging van objecten uit $\mathcal{A}$ aan symbolen uit L is één op één. Dit wil zeggen dat met ieder predicaatsymbool, met ieder functiesymbool en met iedere naam uit het alfabet van L precies één relatie, één functie en één constante van $\mathcal{A}$ correspondeert en omgekeerd.

9.2.2 VOORBEELD Voorbeeld van een structuur.
Zij P de taal van de rekenkunde (zie voorbeeld 8.2.3). Beschouw de volgende structuur voor P:

$$\mathcal{N} = \langle \mathbb{N}; S, +, \cdot\, ; 0\rangle.$$

Deze heeft als domein de verzameling $\mathbb{N}$ der natuurlijke getallen. De functies van $\mathcal{N}$ zijn S (de opvolgerfunctie $S(n) = n + 1$), $+$ en $\cdot$, en de constante is 0. De relatie tussen P en $\mathcal{N}$ is zo dat: $\mathbf{s}^{\mathcal{N}} = S$, $\boldsymbol{+}^{\mathcal{N}} = +$, $\boldsymbol{\cdot}^{\mathcal{N}} = \cdot$ en $\mathbf{0}^{\mathcal{N}} = 0$.

Let op het verschil tussen de symbolen uit P en de symbolen die we voor hun denotaties in de metataal gebruiken! De structuur $\mathcal{N}$ is de *bedoelde structuur* voor P, omdat het domein van de structuur de verzameling der natuurlijke getallen is, en de functiesymbolen op de verwachte manier worden geïnterpreteerd. Een bedoelde structuur wordt vaak een *standaardinterpretatie* genoemd.

Met behulp van deze structuur kunnen we, op informele wijze, een waarheidswaarde toekennen aan zinnen uit P. Zo geldt bijvoorbeeld dat de zin $\mathbf{0} \boldsymbol{+} \mathbf{s}(\mathbf{0}) = \mathbf{s}(\mathbf{0})$ waar is in $\mathcal{N}$, aangezien hier, semantisch gezien, staat dat $0 + S(0) = S(0)$ ofwel $0 + 1 = 1$.

Beschouw nu de open formule $\mathbf{s}(x) = y$. We kunnen voor deze formule niet de waarheidswaarde in $\mathcal{N}$ afleiden. Immers, $\mathcal{N}$ vertelt ons niet hoe de variabelen x en y dienen te worden geïnterpreteerd. ∎

Om alle formules van een waarheidswaarde te kunnen voorzien, dus ook open formules, zijn *bedelingen* nodig. Een bedeling is een functie die aan alle variabelen van een eerste-ordetaal een element van het domein toekent.

9.2.3 DEFINITIE **Bedeling**
Zij $\mathcal{A}$ een structuur voor een eerste-ordetaal L, dan noemt men een functie $\beta : VAR \to |\mathcal{A}|$ een bedeling.

Zij verder $x \in VAR$ en $a \in |\mathcal{A}|$, dan is $\beta[x \mapsto a] : VAR \to |\mathcal{A}|$ de bedeling die als volgt is gedefinieerd:

$$\beta[x \mapsto a](y) = \begin{cases} \beta(y) & \text{als } y \neq x, \\ a & \text{als } y = x. \end{cases}$$

Dit betekent dat de bedeling $\beta[x \mapsto a]$ voor het argument x de waarde a heeft, maar verder gelijk is aan β.

9.2.4 VOORBEELD Voorbeeld van een bedeling.
Zij P de taal van de rekenkunde en $\mathcal{N}$ de bijbehorende structuur zoals gedefinieerd in voorbeeld 9.2.2. Zij β een bedeling die zodanig is dat $\beta(x) = 12$ en $\beta(y) = 13$. Op basis van $\mathcal{N}$ en β kunnen we een waarheidswaarde toekennen aan de formule $\mathbf{s}(x) = y$. Deze formule is voor deze keuze van β waar in $\mathcal{N}$, aangezien $S(12) = 13$ ofwel $12+1 = 13$. Nemen we bijvoorbeeld $\beta(x) = \beta(y) = 17$, dan is de formule onwaar. ■

Een structuur en een bedeling vormen samen een interpretatie. Met behulp van interpretaties kunnen we waarheidswaarde toekennen aan zowel gesloten als open formules. De formele definities hiervoor zullen in de volgende paragraaf worden gegeven.

9.2.5 DEFINITIE **Interpretatie**
Zij L een eerste-ordetaal, $\mathcal{A}$ een structuur voor L en β een bedeling. Dan wordt het geordende paar $\langle \mathcal{A}, \beta \rangle$ een interpretatie *van L genoemd.*

9.2.6 VOORBEELD Voorbeeld van een interpretatie.
Zij P de taal van de rekenkunde, $\mathcal{N}$ de standaardinterpretatie voor P en β een bedeling waarvoor $\beta(x) = 2$. Het geordende paar $\langle \mathcal{N}, \beta \rangle$ is dan een interpretatie voor P.

Beschouw de formule $\mathbf{s}(\mathbf{s}(\mathbf{0})) \cdot x = \mathbf{s}(\mathbf{s}(\mathbf{s}(\mathbf{s}(\mathbf{0}))))$. Deze formule heeft de waarheidswaarde *waar* in $\langle \mathcal{N}, \beta \rangle$, want $S(S(0)) \cdot 2 = S(S(S(S(0))))$, wat equivalent is met $2 \cdot 2 = 4$. ■

9.3 Interpretatie van termen en formules

In deze paragraaf zullen we de waarheidswaarde van een formule onder een gegeven interpretatie definiëren. Ter illustratie beginnen we met een voorbeeld waarin we de zaken vereenvoudigd voorstellen.

9.3.1 VOORBEELD Waarheidswaarde van een formule.
Zij L een eerste-ordetaal waarvan het niet-logische alfabet uitsluitend het tweeplaatsig predicaatsymbool P bevat. Laat verder $\mathcal{A} = \langle \mathbb{N}; < \rangle$ en $\mathcal{B} = \langle \mathbb{Z}; < \rangle$ structuren zijn voor L, zodanig dat $P^{\mathcal{A}} = <$ en $P^{\mathcal{B}} = <$ (daarbij stelt $<$ uiteraard de 'kleiner-dan'-relatie voor).

Beschouw nu de formule $F_1 = \forall y \exists x P(x, y)$. We zullen nu nagaan wat de waarheidswaarde is van de formule F_1 in zowel $\mathcal{A}$ als $\mathcal{B}$. Aangezien F_1 geen vrije variabelen bevat, is hiervoor geen bedeling nodig. Om de waarheidswaarde van F_1 in $\mathcal{A}$ te bepalen, 'vertalen' we de formule in termen van de structuur $\mathcal{A}$. De vertaling luidt dan: voor ieder natuurlijk getal y bestaat er een natuurlijk getal x zodanig dat $x < y$. Aangezien deze uitspraak niet geldig is in $\mathcal{A}$ —voor $y = 0$ bestaat er geen natuurlijke getal x met $x < y$—, concluderen we dat F_1 onwaar is in $\mathcal{A}$. Dezelfde procedure passen we toe om de waarheidswaarde van F_1 in $\mathcal{B}$ te bepalen. De transcriptie van F_1 luidt nu: voor ieder geheel getal y bestaat er een geheel getal x zodanig dat $x < y$. Deze uitspraak is geldig in $\mathcal{B}$, zodat F_1 waar is in $\mathcal{B}$.

Zij F_2 de formule $\forall x \neg P(x, y)$. Om de waarheidswaarde van deze formule te bepalen met betrekking tot $\mathcal{A}$ en $\mathcal{B}$ is nu een bedeling nodig die een waarde toekent aan de vrije variabele y. Zij β een bedeling zodanig dat $\beta(y) = 0$. We zullen nu de waarheidswaarde van F_2 bepalen in de interpretaties $\langle\mathcal{A}, \beta\rangle$ en $\langle\mathcal{B}, \beta\rangle$. Omdat y wordt geïnterpreteerd als het natuurlijke (of gehele) getal 0, kunnen we F_2 in interpretatie $\langle\mathcal{A}, \beta\rangle$ parafraseren als: voor ieder natuurlijk getal x geldt dat niet $x < 0$. Deze uitspraak is uiteraard geldig, zodat F_2 wordt vervuld door $\langle\mathcal{A}, \beta\rangle$. De parafrase van F_2 in $\langle\mathcal{B}, \beta\rangle$ luidt: voor elk geheel getal x geldt dat niet $x < 0$. Deze uitspraak is ongeldig aangezien er negatieve gehele getallen bestaan. Derhalve vervult $\langle\mathcal{B}, \beta\rangle$ de formule F_2 niet. ■

In het onderstaande veronderstellen we dat L een eerste-ordetaal is met predicaatsymbolen $P_0, P_1, \ldots$, functiesymbolen $f_0, f_1, \ldots$ en namen $c_0, c_1, \ldots$. We zullen definiëren wanneer een formule vervuld wordt door een interpretatie $\langle\mathcal{A}, \beta\rangle$. Hiertoe definiëren we eerst hoe de termen van L in $\mathcal{A}$ worden geïnterpreteerd.

9.3.2 DEFINITIE **Denotatie van termen**

Zij $\langle\mathcal{A}, \beta\rangle$ een interpretatie van L. Aan iedere term $t \in TERM$ van L wordt als volgt een denotatie *$t^{\mathcal{A},\beta}$ toegekend:*

$$\begin{aligned} x^{\mathcal{A},\beta} &= \beta(x), \\ c^{\mathcal{A},\beta} &= c^{\mathcal{A}}, \\ f(t_1, \ldots, t_n)^{\mathcal{A},\beta} &= f^{\mathcal{A}}({t_1}^{\mathcal{A},\beta}, \ldots, {t_n}^{\mathcal{A},\beta}). \end{aligned}$$

Voor de betekenis van $c^{\mathcal{A}}$ en $f^{\mathcal{A}}$ wordt naar definitie 9.2.1 verwezen. Wegens stelling 8.3.1 heeft het bovenstaande stelsel recurrente betrekkingen een unieke oplossing, zijnde de functie $\_^{\mathcal{A},\beta} : TERM \rightarrow |\mathcal{A}|$.

Nu zullen we definiëren wat de *waarheidswaarde* van een formule F is met betrekking tot een interpretatie $\langle\mathcal{A}, \beta\rangle$ van L. Het moeilijkst daarbij is de interpretatie van formules die kwantoren bezitten. Daarom eerst een voorbeeld.

9.3.3 VOORBEELD Interpretatie kwantoren.

Zij L de eerste-ordetaal die alleen het éénplaatsige predicaatsymbool P bevat, en zij $\mathcal{Q}$ de structuur $\langle \mathbb{Q}; T \rangle$ waarbij $\mathbb{Q}$ de verzameling der rationale getallen voorstelt en T de verzameling der rationale getallen die groter zijn dan 0. Evident geldt dat $T \subset \mathbb{Q}$. Stel $P^{\mathcal{Q}} = T$. Laat verder β een bedeling zijn.

Zij F de formule $\forall x P(x)$. We willen nu de waarheidswaarde van F recursief kunnen uitdrukken, gegeven de waarheidswaarde van $P(x)$ met betrekking tot $\langle \mathcal{Q}, \beta \rangle$ en het feit dat we weten dat x de vrije variabele is in $P(x)$.

We beginnen met op te merken dat de formule $P(x)$ waar is met betrekking tot $\langle \mathcal{Q}, \beta \rangle$ indien het individu dat x aanduidt (de denotatie van x), de eigenschap bezit die P aanduidt (de denotatie van P); formeel: indien $x^{\mathcal{Q},\beta} \in P^{\mathcal{Q}}$, hetgeen het geval is indien $\beta(x) \in T$. In analogie met de notatie $v(F) = 1$ waarbij v een valuatie is in de propositielogica en F een propositie, zullen we schrijven $v^{\mathcal{Q},\beta}(P(x)) = 1$ als $P(x)$ waar is met betrekking tot $\langle \mathcal{Q}, \beta \rangle$, dus indien $\beta(x) \in T$.

Intuïtief gezien is $\forall x P(x)$ waar indien voor alle q in het domein $\mathbb{Q}$ van onze voorbeeldstructuur $\langle \mathbb{Q}; T \rangle$ geldt dat q de door P aangeduide eigenschap T bezit. Dit kan als volgt worden uitgedrukt:

$$v^{\mathcal{Q},\beta}(\forall x P(x)) = 1 \quad \Leftrightarrow \quad \text{voor alle } q \in \mathbb{Q} \text{ geldt } v^{\mathcal{Q},\beta[x \mapsto q]}(P(x)) = 1.$$

Op deze wijze wordt de x in $P(x)$ 'gekoppeld' aan de q in $\beta[x \mapsto q]$. Bedenk namelijk dat $x^{\mathcal{Q},\beta[x \mapsto q]} = \beta[x \mapsto q](x) = q$, zodat $v^{\mathcal{Q},\beta[x \mapsto q]}(P(x)) = 1$ dan en slechts dan als $q \in T$. Aangezien niet voor alle $q \in \mathcal{Q}$ geldt $q \in T$, is $v^{\mathcal{Q},\beta}(\forall x P(x)) = 0$. Dit geldt trouwens voor alle β.

Op analoge wijze kunnen we analyseren wanneer $\exists x P(x)$ waar is met betrekking tot $\langle \mathcal{Q}, \beta \rangle$:

$$v^{\mathcal{Q},\beta}(\exists x P(x)) = 1 \quad \Leftrightarrow \quad \text{er is een } q \in \mathbb{Q} \text{ zodat } v^{\mathcal{Q},\beta[x \mapsto q]}(P(x)) = 1.$$

Omdat $T \neq \emptyset$, volgt hieruit dat $v^{\mathcal{Q},\beta}(\exists x P(x)) = 1$. Dit geldt wederom voor alle mogelijke β's. ■

Het bovenstaande voorbeeld motiveert de volgende definitie.

9.3.4 Definitie **Waarheidswaarde van formules**

Zij $\langle \mathcal{A}, \beta \rangle$ een interpretatie van L. Aan iedere formule $F \in FORM$ van L wordt door de functie $v^{\mathcal{A},\beta} : FORM \to \mathbb{B}$ een waarheidswaarde *$v^{\mathcal{A},\beta}(F) \in \mathbb{B}$ toegekend in overeenstemming met de volgende recurrente betrekkingen:*

Atomaire formules

$$v^{\mathcal{A},\beta}(P(t_1, \ldots, t_n)) = \begin{cases} 1 & \textit{als } ({t_1}^{\mathcal{A},\beta}, \ldots, {t_n}^{\mathcal{A},\beta}) \in P^{\mathcal{A}}, \\ 0 & \textit{in de overige gevallen.} \end{cases}$$

$$v^{\mathcal{A},\beta}(t_1 = t_2) = \begin{cases} 1 & \textit{als } {t_1}^{\mathcal{A},\beta} = {t_2}^{\mathcal{A},\beta}, \\ 0 & \textit{in de overige gevallen.} \end{cases}$$

Het symbool = zoals het voorkomt in ${t_1}^{\mathcal{A},\beta} = {t_2}^{\mathcal{A},\beta}$, *stelt de gelijkheidsrelatie op* $|\mathcal{A}|$ *voor. Voor de definitie van* $P^{\mathcal{A}}$ *zie definitie 9.2.1.*

Niet-atomaire formules

$$\begin{aligned} v^{\mathcal{A},\beta}(A \star B) &= f_\star(v^{\mathcal{A},\beta}(A), v^{\mathcal{A},\beta}(B)), \\ v^{\mathcal{A},\beta}(\neg A) &= f_\neg(v^{\mathcal{A},\beta}(A)), \\ v^{\mathcal{A},\beta}(\forall x A) &= \begin{cases} 1 & \textit{als voor alle } a \in |\mathcal{A}| \textit{ geldt } v^{\mathcal{A},\beta[x \mapsto a]}(A) = 1, \\ 0 & \textit{in de overige gevallen.} \end{cases} \\ v^{\mathcal{A},\beta}(\exists x A) &= \begin{cases} 1 & \textit{als er een } a \in |\mathcal{A}| \textit{ is zodat } v^{\mathcal{A},\beta[x \mapsto a]}(A) = 1, \\ 0 & \textit{in de overige gevallen.} \end{cases} \end{aligned}$$

De functies $f_\star$ *en* $f_\neg$ *zijn gedefinieerd in paragraaf 3.1.*

Met behulp van stelling 8.3.2 kan men bewijzen dat het hierboven staande stelsel recurrente betrekkingen een unieke oplossing bezit, zijnde een functie $v^{\mathcal{A},\beta} : FORM \to \mathbb{B}$.

Merk op dat de bovenstaande definitie gebruik maakt van definitie 9.3.2.

9.3.5 VOORBEELD Waarheidswaarde van formules.
Zij P de taal van de rekenkunde en $\mathcal{N}$ de bijbehorende standaardinterpretatie (zie voorbeeld 9.2.2). Dan geldt voor alle bedelingen β:

$$\begin{aligned} & v^{\mathcal{N},\beta}(\forall x \neg(\mathbf{s}(x) = \mathbf{0})) = 1 \\ \Leftrightarrow \quad & \text{voor alle } n \in \mathbb{N} \text{ geldt } v^{\mathcal{N},\beta[x \mapsto n]}(\neg(\mathbf{s}(x) = \mathbf{0})) = 1, \\ \Leftrightarrow \quad & \text{voor alle } n \in \mathbb{N} \text{ geldt } v^{\mathcal{N},\beta[x \mapsto n]}(\mathbf{s}(x) = \mathbf{0}) = 0, \\ \Leftrightarrow \quad & \text{voor alle } n \in \mathbb{N} \text{ geldt } \mathbf{s}(x)^{\mathcal{N},\beta[x \mapsto n]} \neq \mathbf{0}^{\mathcal{N},\beta[x \mapsto n]}, \\ \Leftrightarrow \quad & \text{voor alle } n \in \mathbb{N} \text{ geldt } \mathbf{s}^{\mathcal{N}}(x^{\mathcal{N},\beta[x \mapsto n]}) \neq \mathbf{0}^{\mathcal{N}}, \\ \Leftrightarrow \quad & \text{voor alle } n \in \mathbb{N} \text{ geldt } S(\beta[x \mapsto n](x)) \neq 0, \\ \Leftrightarrow \quad & \text{voor alle } n \in \mathbb{N} \text{ geldt } S(n) \neq 0. \end{aligned}$$

Aangezien voor geen enkel natuurlijk getal n de opvolger $S(n)$ van n gelijk is aan 0, volgt hieruit voor alle β dat $v^{\mathcal{N},\beta}(\forall x \neg(\mathbf{s}(x) = \mathbf{0})) = 1$. ∎

9.4 Model en logisch gevolg

De notatie $v^{\mathcal{A},\beta}(F)$ is niet altijd handig. We voeren daarom een andere notatie in.

9.4.1 DEFINITIE **Vervulbaarheidsrelatie**
Voor iedere interpretatie $\langle \mathcal{A}, \beta \rangle$ *en voor iedere formule* $F \in FORM$ *definiëren we:*

$$\langle \mathcal{A}, \beta \rangle \models F \quad \Leftrightarrow \quad v^{\mathcal{A},\beta}(F) = 1.$$

De relatie $\models$ *wordt de* vervulbaarheidsrelatie *genoemd.* $\langle \mathcal{A}, \beta \rangle \models F$ *betekent:* $\langle \mathcal{A}, \beta \rangle$ vervult F.

Bedelingen zijn bedoeld om de vrije variabelen in een term of formule van een denotatie te voorzien. Ze worden ook gebruikt bij de behandeling van de kwantoren. Dit is alleen een technische truc om de betekenis van de kwantoren te relateren aan het domein van een structuur.

Als een formule geen vrije variabelen bevat, is de bedeling irrelevant voor de waarheidswaarde van die formule. Als een formule wel vrije variabelen bevat, heeft de bedeling alleen effect op deze vrije variabelen. Anders gezegd, twee bedelingen die aan de vrije variabelen in een formule dezelfde denotaties toekennen, zijn uitwisselbaar met betrekking tot de waarheidswaarde van die formule. Iets dergelijks geldt natuurlijk ook voor termen.

Het bovenstaande is te generaliseren voor interpretaties. Als twee interpretaties aan alle variabelen en niet-logische symbolen in een term of formule dezelfde betekenis toekennen, dan zal die term of formule dezelfde denotatie, respectievelijk waarheidswaarde, bezitten in deze interpretaties. Dit is de inhoud van de volgende stelling.

9.4.2 STELLING **Interpretatiestelling**
Zij $t \in$ *TERM en* $F \in$ *FORM in een eerste-ordetaal* L*. Zij verder* $\langle \mathcal{A}_1, \beta_1 \rangle$ *en* $\langle \mathcal{A}_2, \beta_2 \rangle$ *interpretaties voor* L*, zodanig dat* $|\mathcal{A}_1| = |\mathcal{A}_2|$ *en* $\beta_1(x) = \beta_2(x)$ *voor alle* $x \in VV(t) \cup VV(F)$ *en zodanig dat* $S^{\mathcal{A}_1} = S^{\mathcal{A}_2}$ *voor alle niet-logische symbolen* S *in* F *en* t*, dan geldt:*

$$\begin{array}{ccc} t^{\mathcal{A}_1,\beta_1} & = & t^{\mathcal{A}_2,\beta_2}, \\ \langle \mathcal{A}_1, \beta_1 \rangle \models F & \Leftrightarrow & \langle \mathcal{A}_2, \beta_2 \rangle \models F. \end{array}$$

BEWIJS *Door middel van structurele inductie over TERM en FORM.* ■

9.4.3 COROLLARIUM *Zij* $F \in$ *ZIN een zin in een eerste-ordetaal* L *en* $\mathcal{A}$ *een structuur voor* L*, dan geldt ofwel voor alle bedelingen* β*, ofwel voor geen enkele bedeling* β *dat* $\langle \mathcal{A}, \beta \rangle \models F$*.*

De volgende definities komen vrijwel overeen met de definities van de corresponderende begrippen in de propositielogica (zie definitie 3.2.1). We noemen een interpretatie een *model* van een formule (of een verzameling van formules), indien die interpretatie de formule (respectievelijk alle formules in die verzameling) waar maakt. Verder noemt men een formule *waar* in een structuur als het er niet toe doet welke bedeling gekozen wordt. Dit geldt bijvoorbeeld voor de formule $(x = x)$. Hierin is x weliswaar een vrije variabele, maar de keuze van de bedeling is niet van belang. Sterker nog, het maakt zelfs niet uit welke structuur men neemt: de formule is *algemeen geldig*. Tenslotte zegt men dat een formule het *logisch gevolg* is van een gegeven verzameling formules, als ieder model van die verzameling tevens een model is voor de gegeven formule.

9.4.4 DEFINITIE **Model, algemene geldigheid en logisch gevolg**

1. *Een* model *voor een formule* $F \in FORM$ *is een interpretatie* $\langle \mathcal{A}, \beta \rangle$ *zodanig dat* $\langle \mathcal{A}, \beta \rangle \models F$. *Men noemt* F vervulbaar *indien* F *een model bezit. Als* $\langle \mathcal{A}, \beta \rangle$ *een model is voor* F, *dan zegt men ook dat* F vervuld *wordt door* $\langle \mathcal{A}, \beta \rangle$.

2. *Een* model *van een verzameling* $\Gamma \subseteq FORM$ *is een interpretatie* $\langle \mathcal{A}, \beta \rangle$ *zodanig dat* $\langle \mathcal{A}, \beta \rangle \models F$ *voor alle* $F \in \Gamma$. *Per definitie is iedere interpretatie een model van de lege verzameling. Men noemt* Γ vervulbaar *indien* Γ *een model bezit. Als* $\langle \mathcal{A}, \beta \rangle$ *een model is voor* Γ, *dan zegt men ook dat* Γ vervuld *wordt door* $\langle \mathcal{A}, \beta \rangle$. *Dit noteert men als* $\langle \mathcal{A}, \beta \rangle \models \Gamma$.

3. *Een formule* F *is* waar *in een structuur* $\mathcal{A}$, *notatie* $\mathcal{A} \models F$, *indien voor alle bedelingen* β *geldt dat* $\langle \mathcal{A}, \beta \rangle \models F$.

4. *Een formule* F *is* algemeen geldig, *notatie* $\models F$, *als voor iedere interpretatie* $\langle \mathcal{A}, \beta \rangle$ *geldt dat* $\langle \mathcal{A}, \beta \rangle \models F$.

5. *Een formule* F *is het* logisch gevolg *van een verzameling* $\Gamma \subseteq FORM$, *notatie* $\Gamma \models F$, *als ieder model van* Γ *tevens een model is voor* F. *Anders gezegd, indien voor alle interpretaties* $\langle \mathcal{A}, \beta \rangle$ *geldt:*

$$\langle \mathcal{A}, \beta \rangle \models \Gamma \quad \Rightarrow \quad \langle \mathcal{A}, \beta \rangle \models F.$$

Hebben we een redenering van de vorm Γ, $\therefore F$, dan noemen we deze *logisch geldig* indien F een logisch gevolg is van Γ. Niet alle redeneringen zijn logisch geldig. In zo'n geval is er een *tegenvoorbeeld*, een interpretatie die een model is van Γ, maar geen model van F.

9.4.5 DEFINITIE **Logisch geldige redenering en tegenvoorbeeld**

Zij $\Gamma \subset FORM$ *en* $F \in FORM$.

1. *De redenering* Γ, $\therefore F$ *is* logisch geldig, *dan en slechts dan als* $\Gamma \models F$.

2. *Een* tegenvoorbeeld *voor de redenering* Γ, $\therefore F$ *is een interpretatie* $\langle \mathcal{A}, \beta \rangle$ *zodanig dat* $\langle \mathcal{A}, \beta \rangle \models \Gamma$, *maar niet* $\langle \mathcal{A}, \beta \rangle \models F$.

Wegens corollarium 9.4.3 en definitie 9.4.4 kunnen we voor zinnen interpretaties identificeren met structuren: de bedelingen doen er niet toe. Dit betekent dat we ook een model voor een zin kunnen identificeren met de structuur van dat model. Een model voor een zin is dus een structuur die de zin waar maakt. Evenzo is een model voor een verzameling zinnen een structuur die alle zinnen uit die verzameling waar maakt.

Dezelfde identificatie is mogelijk bij tegenvoorbeelden voor ongeldige redeneringen waarvan de premissen en de conclusie zinnen zijn: voor de specificatie van een tegenvoorbeeld kan men dan volstaan met een structuur (zie ook hoofdstuk 11).

9.4.6 DEFINITIE **Universele afsluiting van formules**

Zij $F \in FORM$ *met* $VV(F) = \{x_{i_1}, \ldots, x_{i_n}\}$. *De* universele afsluiting *van* F, *notatie* $U(F)$, *is dan gedefinieerd door:*

$$U(F) = \forall x_{i_1} \ldots \forall x_{i_n} F.$$

De volgorde van de variabelen in de kwantoren $\forall x_{i_1} \ldots \forall x_{i_n}$ *mag willekeurig zijn (zie stelling 10.1.10). Voor een zin* $F \in ZIN$ *geldt per definitie* $U(F) = F$.

9.4.7 STELLING *Als* $F \in FORM$ *een formule is uit een eerste-ordetaal* L *en* $\mathcal{A}$ *een structuur voor* L, *dan geldt:*

$$\mathcal{A} \models F \quad \Leftrightarrow \quad \mathcal{A} \models U(F).$$

BEWIJS Het bewijs is met inductie naar het aantal vrije variabelen in F.

1. Basisstap.
   Als $VV(F) = \emptyset$, dan geldt de stelling aangezien in dat geval $U(F) = F$.
2. Inductiestap.
   De inductiehypothese luidt dat de stelling geldig is voor formules met n vrije variabelen.

   Stel dat het aantal vrije variabelen in F gelijk is aan $n + 1$, dat $x \in VV(F)$, en stel dat $\mathcal{A} \models F$. Dit betekent:

   voor alle β geldt $\langle \mathcal{A}, \beta \rangle \models F$.

   Dit is gelijkwaardig met:

   voor alle β en voor alle $a \in |\mathcal{A}|$ geldt $\langle \mathcal{A}, \beta[x \mapsto a] \rangle \models F$,

   hetgeen gelijkwaardig is met:

   voor alle β geldt $\langle \mathcal{A}, \beta \rangle \models \forall x F$.

   Maar dit betekent dat:

   $$\mathcal{A} \models \forall x F. \qquad (9.3)$$

   Hierop mogen we de inductiehypothese toepassen, aangezien $\forall x F$ precies n vrije variabelen bevat. Uitspraak 9.3 is dan gelijkwaardig met:

   $$\mathcal{A} \models U(\forall x F),$$

   hetgeen te schrijven is als $\mathcal{A} \models U(F)$, aangezien $U(\forall x F) = U(F)$. Hiermee is de stelling bewezen. ∎

9.5 Opgaven

1. Zij L de eerste-ordetaal bestaande uit het tweeplaatsige predicaatsymbool P, het éénplaatsige predicaatsymbool S, de namen c en d, en het gelijkheidssymbool $=$. Zij vervolgens $\mathcal{A}$ de structuur voor L die gedefinieerd is door:

   $$\mathcal{A} = \langle D; R_0, R_1; 1, 2\rangle$$

   waarbij:

   - $D = \{1, 2, 3, 4\}$,
   - $P^{\mathcal{A}} = R_0 = \{(1, 2), (2, 3), (2, 4)\}$,
   - $S^{\mathcal{A}} = R_1 = \{2\}$,
   - $c^{\mathcal{A}} = 1$, en
   - $d^{\mathcal{A}} = 2$.

   Zij verder gegeven een bedeling β die voldoet aan: $\beta(x) = 1$, $\beta(y) = 1$ en $\beta(z) = 2$.

   Ga na voor de volgende formules F of $\langle\mathcal{A}, \beta\rangle \models F$.

   (a) $x = y$.
   (b) $x = z$.
   (c) $\neg S(c) \rightarrow S(d)$.
   (d) $\exists x P(x, x)$.
   (e) $\forall x[\neg S(x) \vee \exists y \exists z[P(x, y) \wedge P(x, z) \wedge \neg(y = z)]]$.
   (f) $[\forall x S(x) \vee \forall x \forall y P(x, y)] \rightarrow \forall x \forall y[S(x) \vee P(x, y)]$.

2. Beschouw de formules (a), (e) en (f) uit de vorige opgave. Geef, indien mogelijk, voor ieder van deze formules een interpretatie $\langle\mathcal{A}, \beta\rangle$ zodanig dat:

   (a) formule (a) niet vervuld wordt door $\langle\mathcal{A}, \beta\rangle$.
   (b) formule (e) niet vervuld wordt door $\langle\mathcal{A}, \beta\rangle$.
   (c) formule (f) niet vervuld wordt door $\langle\mathcal{A}, \beta\rangle$.

3. Geef een bewijs van de interpretatiestelling (9.4.2).

Hoofdstuk 10

Stellingen over de Predicatenlogica

In dit hoofdstuk zullen we een aantal stellingen bewijzen over de predicatenlogica. Het betreft meta-stellingen die uiteraard geformuleerd zijn in de metataal. In paragraaf 10.1 introduceren we kwantoren voor de metataal, de equivalentierelatie $\approx$ op formules en bespreken we de distributie van de kwantoren over de logische connectieven. Paragraaf 10.2 gaat over de eigenschappen van substitutie en paragraaf 10.3, tenslotte, handelt over normaalvormen voor formules van de predicatenlogica.

10.1 Eigenschappen van kwantoren

In paragraaf 4.1 hebben we metatalige connectieven geïntroduceerd. Nu zullen we metatalige kwantoren invoeren: $\forall\!\!\!\forall$ met de betekenis *voor alle*, en $\exists\!\!\!\exists$ met de betekenis *er bestaat een*. Deze metakwantoren gebruiken we in de onderstaande stelling. Deze legt het verband tussen de connectieven en kwantoren uit de objecttaal en die uit de metataal.

10.1.1 Stelling *Zij $A, B \in FORM$, en zij $\langle \mathcal{A}, \beta \rangle$ een interpretatie, dan geldt:*

1. $\langle \mathcal{A}, \beta \rangle \models A \wedge B \quad \Leftrightarrow \quad \langle \mathcal{A}, \beta \rangle \models A \quad \& \quad \langle \mathcal{A}, \beta \rangle \models B,$
2. $\langle \mathcal{A}, \beta \rangle \models A \vee B \quad \Leftrightarrow \quad \langle \mathcal{A}, \beta \rangle \models A \quad \bigvee \quad \langle \mathcal{A}, \beta \rangle \models B,$
3. $\langle \mathcal{A}, \beta \rangle \models A \rightarrow B \quad \Leftrightarrow \quad [\, \langle \mathcal{A}, \beta \rangle \models A \quad \Rightarrow \quad \langle \mathcal{A}, \beta \rangle \models B \,],$
4. $\langle \mathcal{A}, \beta \rangle \models \neg A \quad \Leftrightarrow \quad \sim \langle \mathcal{A}, \beta \rangle \models A,$
5. $\langle \mathcal{A}, \beta \rangle \models A \leftrightarrow B \quad \Leftrightarrow \quad [\, \langle \mathcal{A}, \beta \rangle \models A \quad \Leftrightarrow \quad \langle \mathcal{A}, \beta \rangle \models B \,],$
6. $\langle \mathcal{A}, \beta \rangle \models \forall x A \quad \Leftrightarrow \quad \forall\!\!\!\forall a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x \mapsto a] \rangle \models A,$
7. $\langle \mathcal{A}, \beta \rangle \models \exists x A \quad \Leftrightarrow \quad \exists\!\!\!\exists a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x \mapsto a] \rangle \models A.$

BEWIJS De stelling volgt meteen uit definitie 9.3.4. ∎

De bovenstaande stelling geldt ook indien men voor A en B *zinnen* substitueert, en voor een interpretatie $\langle \mathcal{A}, \beta \rangle$ een structuur $\mathcal{A}$. Dat dit in het algemeen niet geldt voor formules die vrije variabelen bevatten, blijkt uit het volgende voorbeeld.

10.1.2 VOORBEELD Als $A, B \in \mathit{FORM}$ en $\mathcal{A}$ een structuur is, dan geldt in het algemeen niet:

$$\mathcal{A} \models A \to B \quad \Leftrightarrow \quad [\mathcal{A} \models A \quad \Rightarrow \quad \mathcal{A} \models B].$$

Dit is als volgt in te zien. Laat een eerste-ordetaal L alleen de éénplaatsige predicaatsymbolen P en Q bevatten. Beschouw nu de structuur $\mathcal{Z} = \langle \mathbb{Z}; E, O \rangle$ waarbij E de verzameling der even gehele getallen voorstelt, en O die der oneven. Stel dat $P^{\mathcal{Z}} = E$ en $Q^{\mathcal{Z}} = O$.

Enerzijds geldt nu noch $\mathcal{Z} \models P(x)$, noch $\mathcal{Z} \models Q(x)$. Dit betekent dat de metabewering:

$$\mathcal{Z} \models P(x) \quad \Rightarrow \quad \mathcal{Z} \models Q(x)$$

geldig is. Anderzijds geldt echter niet dat $\mathcal{Z} \models P(x) \to Q(x)$. Hiermee is een tegenvoorbeeld gegeven. ∎

Zoals we in definitie 4.1.5 hebben gezien, is een equivalentierelatie een reflexieve, symmetrische en transitieve relatie. Net als in de propositielogica kunnen we een equivalentierelatie op formules definiëren.

10.1.3 DEFINITIE **Equivalentie van formules**

Twee formules $A, B \in \mathit{FORM}$ *noemt men* equivalent, *notatie* $A \approx B$, *als geldt* $\models A \leftrightarrow B$.

10.1.4 STELLING *De relatie* $\approx$ *is een equivalentierelatie.*

BEWIJS Analoog aan het bewijs van stelling 4.1.7. ∎

Equivalentierelaties geven aanleiding tot zogenaamde *equivalentieklassen* die samen een *partitie* vormen. In het volgende hoofdstuk over de boommethode zullen we van het begrip equivalentieklasse gebruik maken.

10.1.5 DEFINITIE **Equivalentieklasse**

Zij R *een equivalentierelatie op* D. *Als* $a \in D$, *dan noemt men de verzameling:*

$$[a]_R = \{b \in D \mid aRb\}$$

de equivalentieklasse *van* a. *Men noemt* a *een* representant *van* $[a]_R$.

Ieder element $b \in [a]_R$ kan als representant van $[a]_R$ worden gekozen. Immers, $[b]_R = [a]_R$ dan en slechts dan als $b \in [a]_R$.

10.1.6 DEFINITIE **Partitie**
Een klasse $\mathcal{C}$ van deelverzamelingen van een gegeven verzameling D is een partitie *van D dan en slechts dan als aan de volgende voorwaarden is voldaan:*

1. $A \in \mathcal{C} \quad \Rightarrow \quad A \neq \emptyset$;
2. $A, B \in \mathcal{C} \quad \Rightarrow \quad [\ A = B \quad \bigvee \quad A \cap B = \emptyset\ ]$;
3. $\bigcup \mathcal{C} = D$.

Een partitie van een verzameling D is dus een klasse van niet-lege, disjuncte deelverzamelingen van D waarvan de vereniging gelijk is aan D.

10.1.7 VOORBEELD Partitie.
Beschouw de relatie $R \subseteq \mathbb{N} \times \mathbb{N}$ uit voorbeeld 4.1.6 die als volgt gedefinieerd is: xRy geldt dan en slechts dan als x en y na deling door 7 dezelfde rest hebben. Deze relatie resulteert in de partitie $\{K_0, K_1, \ldots, K_6\}$, waarin $K_i = \{n \in \mathbb{N} \mid n$ modulo 7 is gelijk aan $i\}$. Zo geldt bijvoorbeeld dat $K_3 = \{3, 10, 17, 24, \ldots\}$.

Het is niet moeilijk om de volgende stelling te bewijzen.

10.1.8 STELLING *Als R een equivalentierelatie is op D, dan is:*

$$\{\ [a]_R \mid a \in D\}$$

een partitie van D.

Als de formules A en B equivalent zijn, dus als $A \approx B$, dan geldt voor alle interpretaties $\langle \mathcal{A}, \beta \rangle$ dat $v^{\mathcal{A},\beta}(A) = v^{\mathcal{A},\beta}(B)$. Dit betekent dat de waarheidswaarde van een formule F met betrekking tot een interpretatie $\langle \mathcal{A}, \beta \rangle$ niet verandert, als men in A sommige voorkomens van A door B vervangt.

10.1.9 STELLING **Vervangingsstelling**
Zij $F, A, B \in FORM$ en zij A een subformule van F. Als $A \approx B$ en G een formule is verkregen uit F door sommige voorkomens van A door B te vervangen, dan $F \approx G$.

BEWIJS *Door middel van structurele inductie over F:*

1. *F is een atomaire formule.*
   Als A een subformule van F is, dan geldt $A = F$. Vervanging van A door B levert $G = B$. Evident geldt dan $F \approx G$. Wordt A niet door B vervangen, dan is $G = F$ en dus $F \approx G$.

2. *$F = \neg C$ voor een $C \in FORM$.*
   Als $F = A$, dan volgt het resultaat op precies dezelfde wijze als bij geval 1 hierboven. Als A een subformule is van C en C' is verkregen door in C sommige voorkomens van A door B te vervangen, dan volgt uit de inductiehypothese dat $C \approx C'$. Hieruit volgt voor alle interpretaties $\langle \mathcal{A}, \beta \rangle$:

$$v^{\mathcal{A},\beta}(F) = f_\neg(v^{\mathcal{A},\beta}(C)) = f_\neg(v^{\mathcal{A},\beta}(C')) = v^{\mathcal{A},\beta}(G).$$

De overige gevallen worden aan de lezer overgelaten. ∎

De volgende stelling drukt uit dat de volgorde van kwantoren die tot dezelfde soort behoren (met de daarbij behorende variabelen), er niets toe doet. Bovendien kunnen kwantoren over variabelen die niet voorkomen in een formule vervallen.

10.1.10 Stelling *Zij $A \in FORM$, dan geldt:*

1. $\forall x \forall y A \approx \forall y \forall x A$,
2. $\exists x \exists y A \approx \exists y \exists x A$,
3. $\forall x A \approx A$ *als* $x \notin VV(A)$,
4. $\exists x A \approx A$ *als* $x \notin VV(A)$.

Bewijs Triviaal. ∎

Er bestaat een mooi verband tussen de kwantoren $\forall$ en $\exists$.

10.1.11 Stelling *Zij $A \in FORM$, dan geldt:*

1. $\forall x A \approx \neg \exists x \neg A$,
2. $\exists x A \approx \neg \forall x \neg A$.

Bewijs Triviaal. ∎

Kwantoren kunnen soms *gedistribueerd* worden. In de volgende stelling staan deze mogelijkheden opgesomd.

10.1.12 Stelling **Distributie van kwantoren**
Zij $A, B \in FORM$, dan geldt:

1. $\forall x(A \wedge B) \approx (\forall x A \wedge \forall x B)$,
2. $\exists x(A \vee B) \approx (\exists x A \vee \exists x B)$,
3. $\forall x(A \vee B) \approx (\forall x A \vee B)$ *als* $x \notin VV(B)$,

4. $\exists x(A \wedge B) \;\approx\; (\exists x A \wedge B)$ *als* $x \notin VV(B)$.

BEWIJS *We bewijzen de beweringen 1 en 4, en laten de overige aan de lezer over.*

1. *Bewijs bewering 1.*

   Zij $\langle \mathcal{A}, \beta \rangle$ *een model voor* $\forall x(A \wedge B)$. *Dit betekent dat:*

   $$\forall a \in |\mathcal{A}| \cdot v^{\mathcal{A},\beta[x\mapsto a]}(A \wedge B) = 1,$$

   hetgeen equivalent is met:

   $$\forall a \in |\mathcal{A}| \cdot v^{\mathcal{A},\beta[x\mapsto a]}(A) = v^{\mathcal{A},\beta[x\mapsto a]}(B) = 1.$$

   Maar dit betekent dat:

   $$v^{\mathcal{A},\beta}(\forall x A) = v^{\mathcal{A},\beta}(\forall x B) = 1,$$

   ofwel:

   $$v^{\mathcal{A},\beta}(\forall x A \wedge \forall x B) = 1,$$

   zodat $\langle \mathcal{A}, \beta \rangle$ *een model is voor* $(\forall x A \wedge \forall x B)$.

2. *Bewijs bewering 4.*

   Zij $\langle \mathcal{A}, \beta \rangle$ *een model voor* $\exists x(A \wedge B)$. *Dit wil zeggen:*

   $$\exists a \in |\mathcal{A}| \cdot v^{\mathcal{A},\beta[x\mapsto a]}(A \wedge B) = 1,$$

   hetgeen equivalent is met:

   $$\exists a \in |\mathcal{A}| \cdot v^{\mathcal{A},\beta[x\mapsto a]}(A) = v^{\mathcal{A},\beta[x\mapsto a]}(B) = 1. \tag{10.1}$$

   Omdat $x \notin VV(B)$, *geldt dat* $v^{\mathcal{A},\beta[x\mapsto a]}(B) = v^{\mathcal{A},\beta}(B)$ *(zie stelling 9.4.2), zodat 10.1 gelijkwaardig is met:*

   $$[\exists a \in |\mathcal{A}| \cdot v^{\mathcal{A},\beta[x\mapsto a]}(A) = 1] \quad \& \quad v^{\mathcal{A},\beta}(B) = 1,$$

   Dit is equivalent met:

   $$v^{\mathcal{A},\beta}(\exists x A) = v^{\mathcal{A},\beta}(B) = 1,$$

   ofwel $v^{\mathcal{A},\beta}(\exists x A \wedge B) = 1$, *dus* $\langle \mathcal{A}, \beta \rangle$ *is een model voor* $(\exists x A \wedge B)$. ∎

Dat we met de distributie van kwantoren moeten oppassen wordt in het volgende voorbeeld duidelijk gemaakt.

10.1.13 VOORBEELD Distributie kwantoren.
De volgende beweringen die 'sprekend' lijken op de beweringen 3 en 4 van de vorige stelling zijn in het algemeen *niet* correct:

1. $\forall x(A \vee B) \;\approx\; (\forall x A \vee \forall x B)$,
2. $\exists x(A \wedge B) \;\approx\; (\exists x A \wedge \exists x B)$.

Als men in bewering 1 de A en B respectievelijk vervangt door de literalen $P(x)$ en $\neg P(x)$, en als $\mathcal{A}$ een structuur is waarin $P^{\mathcal{A}} \neq \emptyset$ en $P^{\mathcal{A}} \neq |\mathcal{A}|$, dan geldt voor alle β dat $v^{\mathcal{A},\beta}(\forall x[P(x) \vee \neg P(x)]) = 1$ en $v^{\mathcal{A},\beta}(\forall x P(x) \vee \forall x \neg P(x)) = 0$.

Als men in bewering 2 de A en B respectievelijk vervangt door de atomaire formules $P(x)$ en $Q(x)$, en als $\mathcal{A}$ een structuur is waarin $P^{\mathcal{A}} \neq \emptyset$, $Q^{\mathcal{A}} \neq \emptyset$ en $P^{\mathcal{A}} \cap Q^{\mathcal{A}} = \emptyset$, dan geldt voor alle β dat $v^{\mathcal{A},\beta}(\exists x[P(x) \wedge Q(x)]) = 0$ en $v^{\mathcal{A},\beta}(\exists x P(x) \wedge \exists x Q(x)) = 1$. ∎

10.2 Eigenschappen van substitutie

In de eerste stelling van deze paragraaf wordt tot uitdrukking gebracht dat de semantiek van de predicatenlogica zodanig is gedefinieerd dat de gelijkheid $=$ een equivalentierelatie voorstelt (zoals men van de gelijkheid gewend is in de wiskunde).

10.2.1 STELLING **Eigenschappen gelijkheid**
De gelijkheid is reflexief, symmetrisch en transitief. Dus voor alle termen $s, t, r \in \mathit{TERM}$ *van een eerste-ordetaal* L *geldt:*

$$\begin{array}{rcl} & \models & t = t, \\ s = t & \models & t = s, \\ s = t,\ t = r & \models & s = r. \end{array}$$

BEWIJS Zij $\mathcal{A}$ een structuur voor L. De gelijkheid op $\mathcal{A}$ is reflexief, symmetrisch en transitief (zie definitie 9.3.4). Hieruit kan vrijwel meteen de stelling worden afgeleid. ∎

De volgende twee stellingen hebben betrekking op het substitueren van termen in termen, respectievelijk van termen in formules. Deze stellingen drukken uit dat het resultaat van het eerst uitvoeren van de substitutie $[x:=t]$ op een term s of formule F en het vervolgens bepalen van de betekenis van $s[x:=t]$ of $F[x:=t]$ met betrekking tot een interpretatie $\langle \mathcal{A}, \beta \rangle$ overeenstemt met het bepalen van de betekenis van s of F met betrekking tot de interpretatie $\langle \mathcal{A}, \beta[x \mapsto t^{\mathcal{A},\beta}] \rangle$ ofwel de interpretatie $\langle \mathcal{A}, \beta \rangle$ die zodanig is aangepast dat aan de variable x de waarde $t^{\mathcal{A},\beta}$ wordt toegekend. Anders gezegd, substitutie op syntactisch niveau komt overeen met 'substitutie' op semantisch niveau.

10.2.2 STELLING **Substitutiestelling voor termen**
Zij $s, t \in \mathit{TERM}$ *in een eerste-ordetaal* L, *en* $\langle \mathcal{A}, \beta \rangle$ *een interpretatie voor* L, *dan:*

$$s[x:=t]^{\mathcal{A},\beta} = s^{\mathcal{A},\beta[x \mapsto t^{\mathcal{A},\beta}]}.$$

BEWIJS Het bewijs verloopt via structurele inductie over de term s.

1. $s = c$ waarbij c een naam is.
   Dan geldt de stelling omdat $c[x:=t] = c$ en $c^{\mathcal{A},\beta} = c^{\mathcal{A},\beta[x \mapsto t^{\mathcal{A},\beta}]} = c^{\mathcal{A}}$.
2. $s = y$ met $y \in \mathit{VAR}$ en $y \neq x$.
   Dit geval gaat op dezelfde wijze als het vorige.
3. $s = x$.
   Nu volgt de stelling aangezien $x[x:=t] = t$ en:
   $$x^{\mathcal{A},\beta[x \mapsto t^{\mathcal{A},\beta}]} = \beta[x \mapsto t^{\mathcal{A},\beta}](x) = t^{\mathcal{A},\beta}.$$
4. $s = f(s_1, \ldots, s_n)$ met $s_1, \ldots, s_n \in \mathit{TERM}$.
   De inductiehypothese luidt dat voor alle $1 \leq i \leq n$:
   $$s_i[x:=t]^{\mathcal{A},\beta} = s_i^{\mathcal{A},\beta[x \mapsto t^{\mathcal{A},\beta}]}.$$

Nu volgt:

$$\begin{array}{ll} \multicolumn{2}{l}{f(s_1,\ldots,s_n)[x:=t]^{\mathcal{A},\beta}} \\ = & f(s_1[x:=t],\ldots,s_n[x:=t])^{\mathcal{A},\beta}, \\ = & f^{\mathcal{A}}(s_1[x:=t]^{\mathcal{A},\beta},\ldots,s_n[x:=t]^{\mathcal{A},\beta}), \\ = & f^{\mathcal{A}}(s_1^{\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]},\ldots,s_n^{\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]}) \\ & \text{(vanwege de inductiehypothese),} \\ = & f(s_1,\ldots,s_n)^{\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]}. \end{array}$$

Hiermee is de stelling bewezen. ∎

10.2.3 Corollarium *Zij $s, t_1, t_2 \in$ TERM in een eerste-ordetaal L, dan geldt:*

$$t_1 = t_2 \models s[x:=t_1] = s[x:=t_2].$$

Bewijs Zij $\langle\mathcal{A},\beta\rangle$ een interpretatie voor L. Neem vervolgens aan dat

$$t_1{}^{\mathcal{A},\beta} = t_2{}^{\mathcal{A},\beta}. \tag{10.2}$$

Volgens de voorgaande stelling geldt voor $i = 1, 2$:

$$s[x:=t_i]^{\mathcal{A},\beta} = s^{\mathcal{A},\beta[x\mapsto t_i{}^{\mathcal{A},\beta}]}.$$

Hieruit en uit 10.2 volgt onmiddellijk het corollarium. ∎

Ten behoeve van het bewijs van de volgende stelling definiëren we de *complexiteit* van een formule (zie ook definitie 2.3.3). Dit is een maat voor het aantal syntactische regels dat is toegepast om de formule voort te brengen.

10.2.4 Definitie **Complexiteit van een formule**
De complexiteit *$comp(F)$ van een formule $F \in$ FORM is als volgt gedefinieerd:*

$$\begin{array}{rcl} comp(t_1 = t_2) & = & 0, \\ comp(P(t_1,\ldots,t_n)) & = & 0, \\ comp(\neg A) & = & comp(A) + 1, \\ comp(A \star B) & = & comp(A) + comp(B) + 1, \\ comp(\mathrm{Q}xA) & = & comp(A) + 1. \end{array}$$

Stelling 8.3.2 garandeert dat het bovenstaande stelsel recurrente betrekkingen een unieke oplossing $comp : FORM \to \mathbb{N}$ bezit.

10.2.5 Stelling **Substitutiestelling voor formules**
Zij $t \in$ TERM en $F \in$ FORM in een eerste-ordetaal L, dan geldt:

$$\langle\mathcal{A},\beta\rangle \models F[x:=t] \quad\Leftrightarrow\quad \langle\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]\rangle \models F.$$

Bewijs Het bewijs verloopt door middel van volledige inductie over de complexiteit $comp(F)$ van de formules $F \in FORM$.

1. Basisstap: $comp(F) = 0$, zodat F atomair is.

   Zij $F = (s_1 = s_2)$ met $s_1, s_2 \in TERM$.

   Nu geldt:

$$\begin{array}{ll} \langle\mathcal{A},\beta\rangle \models (s_1 = s_2)[x:=t] & \\ \quad\Leftrightarrow \quad \langle\mathcal{A},\beta\rangle \models s_1[x:=t] = s_2[x:=t], & \\ \quad\Leftrightarrow \quad s_1[x:=t]^{\mathcal{A},\beta} = s_2[x:=t]^{\mathcal{A},\beta}, & \\ \quad\Leftrightarrow \quad s_1^{\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]} = s_2^{\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]} & \text{(wegens stelling 10.2.2),} \\ \quad\Leftrightarrow \quad \langle\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]\rangle \models s_1 = s_2. & \end{array}$$

   Het geval dat $F = P(s_1, \ldots, s_n)$ met $s_1, \ldots, s_n \in TERM$ is op analoge wijze te behandelen.

2. Inductiestap: $comp(F) = n + 1$, zodat F niet-atomair is.

   De inductiehypothese (IH) is dat voor alle formules C met $comp(C) \leq n$ geldt:

$$\langle\mathcal{A},\beta\rangle \models C[x:=t] \quad\Leftrightarrow\quad \langle\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]\rangle \models C.$$

   (a) Zij $F = (A \wedge B)$ met $A, B \in FORM$. Hieruit volgt dat $comp(A) \leq n$ en $comp(B) \leq n$.

   Nu geldt dat:

$$\begin{array}{l} \langle\mathcal{A},\beta\rangle \models (A \wedge B)[x:=t] \\ \quad\Leftrightarrow \quad \langle\mathcal{A},\beta\rangle \models (A[x:=t] \wedge B[x:=t]), \\ \quad\Leftrightarrow \quad \langle\mathcal{A},\beta\rangle \models A[x:=t] \quad \& \quad \langle\mathcal{A},\beta\rangle \models B[x:=t] \\ \qquad\qquad \text{(wegens stelling 10.1.1),} \\ \quad\Leftrightarrow \quad \langle\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]\rangle \models A \quad \& \quad \langle\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]\rangle \models B \\ \qquad\qquad \text{(wegens de inductiehypothese),} \\ \quad\Leftrightarrow \quad \langle\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]\rangle \models A \wedge B. \end{array}$$

   De overige connectieven worden op analoge wijze behandeld.

   (b) Zij $F = \forall y A$ met $A \in FORM$. Hieruit volgt dat $comp(A) = n$. We beschouwen de vier gevallen die voortvloeien uit toepassing van definitie 8.4.2.

   Geval 1: $y \neq x$ en $y \notin VV(t)$. We leiden af:

$$\begin{array}{l} \langle\mathcal{A},\beta\rangle \models (\forall y A)[x:=t] \\ \quad\Leftrightarrow \quad \langle\mathcal{A},\beta\rangle \models \forall y(A[x:=t]), \\ \quad\Leftrightarrow \quad \forall a \in |\mathcal{A}| \cdot \langle\mathcal{A},\beta[y\mapsto a]\rangle \models A[x:=t], \\ \quad\Leftrightarrow \quad \forall a \in |\mathcal{A}| \cdot \langle\mathcal{A},\beta[y\mapsto a][x\mapsto t^{\mathcal{A},\beta[y\mapsto a]}]\rangle \models A \\ \qquad\qquad \text{(wegens } comp(A) = n \text{ en (IH)),} \\ \quad\Leftrightarrow \quad \forall a \in |\mathcal{A}| \cdot \langle\mathcal{A},\beta[y\mapsto a][x\mapsto t^{\mathcal{A},\beta}]\rangle \models A \\ \qquad\qquad \text{(aangezien } y \notin VV(t) \text{ en stelling 9.4.2),} \\ \quad\Leftrightarrow \quad \forall a \in |\mathcal{A}| \cdot \langle\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}][y\mapsto a]\rangle \models A \\ \qquad\qquad \text{(omdat } x \neq y\text{),} \\ \quad\Leftrightarrow \quad \langle\mathcal{A},\beta[x\mapsto t^{\mathcal{A},\beta}]\rangle \models \forall y A. \end{array}$$

   Geval 2: $y \neq x$ en $x \notin VV(A)$. Wordt aan de lezer overgelaten.

Geval 3: $y \neq x$, $y \in VV(t)$ en $x \in VV(A)$. Zij z de eerste variabele waarvoor geldt dat $t \notin VV(A) \cup VV(t)$, dan leiden we af:

$$
\begin{array}{ll}
\multicolumn{2}{l}{\langle \mathcal{A}, \beta \rangle \models (\forall y A)[x := t]} \\
\Leftrightarrow & \langle \mathcal{A}, \beta \rangle \models \forall z (A[y := z][x := t]), \\
\Leftrightarrow & \forall a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[z \mapsto a] \rangle \models A[y := z][x := t], \\
\Leftrightarrow & \forall a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[z \mapsto a][x \mapsto t^{\mathcal{A}, \beta[z \mapsto a]}] \rangle \models A[y := z] \\
 & \text{(wegens } comp(A[y := z]) = n \text{ en (IH)),} \\
\Leftrightarrow & \forall a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[z \mapsto a][x \mapsto t^{\mathcal{A}, \beta}] \rangle \models A[y := z] \\
 & \text{(aangezien } z \notin VV(t) \text{ en stelling 9.4.2),} \\
\Leftrightarrow & \forall a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x \mapsto t^{\mathcal{A}, \beta}][z \mapsto a] \rangle \models A[y := z] \\
 & \text{(daar } x \neq z \text{, omdat } x \in VV(A) \text{ en } z \notin VV(A)), \\
\Leftrightarrow & \forall a \in |\mathcal{A}| \cdot \\
 & \langle \mathcal{A}, \beta[x \mapsto t^{\mathcal{A}, \beta}][z \mapsto a][y \mapsto z^{\mathcal{A}, \beta[x \mapsto t^{\mathcal{A}, \beta}][z \mapsto a]}] \rangle \models A \\
 & \text{(nogmaals (IH)),} \\
\Leftrightarrow & \forall a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x \mapsto t^{\mathcal{A}, \beta}][z \mapsto a][y \mapsto a] \rangle \models A, \\
\Leftrightarrow & \forall a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x \mapsto t^{\mathcal{A}, \beta}][y \mapsto a][z \mapsto a] \rangle \models A, \\
\Leftrightarrow & \forall a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x \mapsto t^{\mathcal{A}, \beta}][y \mapsto a] \rangle \models A \\
 & \text{(aangezien } z \notin VV(A) \text{ en stelling 9.4.2),} \\
\Leftrightarrow & \langle \mathcal{A}, \beta[x \mapsto t^{\mathcal{A}, \beta}] \rangle \models \forall y A.
\end{array}
$$

Geval 4: $y = x$. Wordt aan de lezer overgelaten.

Het bewijs voor het geval dat $F = \exists y A$ gaat op precies dezelfde wijze als dat voor $F = \forall y A$. ■

De reden dat we deze stelling hebben aangetoond met behulp van volledige inductie naar de complexiteit van de formules, is gelegen in het feit dat we een geschikte inductiehypothese nodig hadden in het derde geval van $F = \mathrm{Q} y A$. We willen immers de inductiehypothese kunnen toepassen op $A[y := z]$, in plaats van op A. Hadden we structurele inductie over *FORM* toegepast, dan hadden we een inductiehypothese ter beschikking gehad die alleen op de formule A van toepassing was.

10.2.6 COROLLARIUM *Zij $s, t \in TERM$ en $F \in FORM$ in een eerste-ordetaal L, dan geldt:*

$$s = t \models F[x := s] \leftrightarrow F[x := t].$$

BEWIJS Het bewijs is eenvoudig en lijkt op dat van corollarium 10.2.3. ■

De laatste stelling van deze paragraaf drukt uit dat de semantiek van de predicatenlogica adequaat omgaat met gebonden variabelen: de naam van een gebonden variabele is niet van belang onder de voorwaarde dat deze verschillend is van de vrije variabelen in de betreffende formule. Dit betekent dat gebonden variabelen mogen worden herbenoemd zonder dat de betekenis van de formule verandert.

10.2.7 STELLING **Herbenoemen gebonden variabelen**

Zij $A \in FORM$ en $y \notin VV(A)$, dan geldt:

1. $\exists x A \;\approx\; \exists y(A[x:=y])$,

2. $\forall x A \;\approx\; \forall y(A[x:=y])$.

BEWIJS *We bewijzen alleen het eerste geval; het tweede gaat op dezelfde wijze. Neem aan dat $y \notin VV(A)$, we kunnen dan afleiden:*

$$\begin{array}{ll}
\multicolumn{2}{l}{\langle \mathcal{A}, \beta \rangle \models \exists y(A[x:=y])} \\
\Leftrightarrow & \exists a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[y \mapsto a] \rangle \models A[x:=y], \\
\Leftrightarrow & \exists a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[y \mapsto a][x \mapsto y^{\mathcal{A}, \beta[y \mapsto a]}] \rangle \models A \\
 & \textit{(wegens stelling 10.2.5)}, \\
\Leftrightarrow & \exists a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[y \mapsto a][x \mapsto a] \rangle \models A, \\
\Leftrightarrow & \exists a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x \mapsto a][y \mapsto a] \rangle \models A, \\
\Leftrightarrow & \exists a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x \mapsto a] \rangle \models A \\
 & \textit{(aangezien } y \notin VV(A) \textit{ en stelling 9.4.2)}, \\
\Leftrightarrow & \langle \mathcal{A}, \beta \rangle \models \exists x A. \quad \blacksquare
\end{array}$$

10.2.8 VOORBEELD Herbenoemen van variabelen.

1. Een direkte uitvoering van de substitutie $[y:=f(x)]$ op de formule $F = \forall x P(x,y)$ geeft een onbedoelde binding van $f(x)$ aan $\forall x$. Door herbenoeming van de gebonden variabele x gaat F over in de variant $G = \forall z P(z,y)$ waarvoor $G \approx F$. Op G mag de substitutie wel uitgevoerd worden.

2. Om stelling 10.1.12 over het distribueren van kwantoren over $\wedge$ en $\vee$ te kunnen toepassen moet men soms eerst de variabelen herbenoemen. In het volgende voorbeeld verloopt de distributie van de $\exists$-kwantoren over het connectief $\wedge$ als volgt:

$$\begin{array}{rcl}
\exists x P(x) \wedge \exists x \neg Q(x) & \approx & \exists x[P(x) \wedge \exists x \neg Q(x)], \\
 & \approx & \exists x[P(x) \wedge \exists y \neg Q(y)], \\
 & \approx & \exists x \exists y[P(x) \wedge \neg Q(y)].
\end{array}$$

In dit voorbeeld is bewering 4 uit stelling 10.1.12 tweemaal toegepast, nadat de gebonden variabele x in $\neg Q(x)$ is herbenoemd tot y. Daarnaast is de vervangingsstelling (10.1.9) gebruikt. ∎

10.3 Normaalvormen

Ook in de predicatenlogica kent men normaalvormen. In de volgende definitie introduceren we de zogenaamde *prenexnormaalvorm*. Bij een formule in deze normaalvorm staan alle kwantoren vooraan de formule.

10.3.1 DEFINITIE **Prenexnormaalvorm**
Een formule $F \in FORM$ staat in prenexnormaalvorm *als F de volgende vorm bezit:*

$$\mathsf{Q}_1 x_{i_1} \ldots \mathsf{Q}_n x_{i_n} M.$$

Hierin zijn $\mathsf{Q}_1, \ldots, \mathsf{Q}_n$ universele of existentiële kwantoren en bevat M geen kwantoren. M noemt men de matrix *van de formule F en de rij kwantoren $\mathsf{Q}_1 x_{i_1} \ldots \mathsf{Q}_n x_{i_n}$ de* prenex *van F. De prenex mag eventueel leeg zijn, dus $n = 0$.*

10.3.2 VOORBEELD Prenexnormaalvorm.
De formule $\forall x \exists y[(P(x) \vee Q(x,y)) \rightarrow R(x)]$ staat in prenexnormaalvorm. ■

10.3.3 STELLING *Voor iedere formule $F \in FORM$ bestaat er een formule F^p in prenexnormaalvorm waarvoor geldt dat $F \approx F^p$.*

BEWIJS Het in prenexnormaalvorm brengen van een formule komt in principe neer op het herhaaldelijk toepassen van de stellingen 10.2.7 en 10.1.12 (ga na).

Prenexnormaalvorm-algoritme

1. Elimineer eerst alle voorkomens van $\rightarrow$ en $\leftrightarrow$ door de volgende herschrijvingen toe te passen (van links naar rechts) op subformules van F:

$$\begin{array}{rcl} A \rightarrow B & \approx & \neg A \vee B, \\ A \leftrightarrow B & \approx & (\neg A \vee B) \wedge (\neg B \vee A). \end{array}$$

2. Distribueer vervolgens $\neg$ over de kwantoren door de volgende herschrijvingen toe te passen (van links naar rechts):

$$\begin{array}{rcl} \neg \forall x A & \approx & \exists x \neg A, \\ \neg \exists x A & \approx & \forall x \neg A \end{array}$$

3. Breng tenslotte alle kwantoren naar buiten door de volgende herschrijvingen toe te passen (van links naar rechts):

$$\begin{array}{rcll} \forall x A \wedge \forall x B & \approx & \forall x(A \wedge B), & \\ \exists x A \vee \exists x B & \approx & \exists x(A \vee B), & \\ \mathsf{Q} x A \star B & \approx & \mathsf{Q} x(A \star B), & \text{als } x \notin VV(B), \\ A \star \mathsf{Q} x B & \approx & \mathsf{Q} x(A \star B), & \text{als } x \notin VV(A). \end{array}$$

Waarbij $\mathsf{Q} \in \{\forall, \exists\}$ en $\star \in \{\wedge, \vee\}$. Om deze herschrijfregels te kunnen toepassen mogen gebonden variabelen worden herbenoemd. ■

10.3.4 VOORBEELD Prenexnormaalvorm.
We bepalen een prenexnormaalvorm van $F = \forall x P(x) \rightarrow \exists z \forall y R(z, y)$.

$$\begin{aligned} F &\approx \forall x P(x) \rightarrow \exists z \forall y R(z, y), \\ &\approx \neg \forall x P(x) \vee \exists z \forall y R(z, y), \\ &\approx \exists x \neg P(x) \vee \exists z \forall y R(z, y), \\ &\approx \exists x \neg P(x) \vee \exists x \forall y R(x, y), \\ &\approx \exists x [\neg P(x) \vee \forall y R(x, y)], \\ &\approx \exists x \forall y [\neg P(x) \vee R(x, y)]. \end{aligned}$$

Let op hoe in de vierde regel de variabele z wordt herbenoemd tot x om de tweede herschrijvingsregel te kunnen toepassen. Als we het herbenoemen achterwege laten, krijgen we:

$$\begin{aligned} F &\approx \exists x \neg P(x) \vee \exists z \forall y R(z, y), \\ &\approx \exists x [\neg P(x) \vee \exists z \forall y R(z, y)], \\ &\approx \exists x \exists z [\neg P(x) \vee \forall y R(z, y)], \\ &\approx \exists x \exists z \forall y [\neg P(x) \vee R(z, y)]. \end{aligned}$$

De eerste normaalvorm is eenvoudiger dan de tweede, aangezien zij minder kwantoren bevat. Het voorbeeld laat tevens zien dat prenexnormaalvormen niet uniek behoeven te zijn. ∎

10.4 Opgaven

1. Bewijs stelling 10.1.8
2. Bewijs stelling 10.1.11.
3. Bewijs de gevallen 2 en 3 van stelling 10.1.12.
4. Voltooi het bewijs van stelling 10.2.5.
5. Breng de volgende formules in prenexnormaalvorm:
   (a) $\forall x [P(x) \vee Q(x)] \wedge \exists x R(x)$.
   (b) $\forall x P(x) \rightarrow [\exists x Q(x) \rightarrow \forall x R(x)]$.
   (c) $\forall x \exists y [R(x, y) \rightarrow \exists z Q(z, y)]$.
   (d) $\exists z [\forall x \exists y R(x, y, z) \leftrightarrow \exists x \forall y S(x, y, z)]$.

Hoofdstuk 11

De Boommethode voor de Predicatenlogica

In hoofdstuk 5 hebben we de boommethode voor de propositielogica geïntroduceerd. Deze methode kan met een aantal regels worden uitgebreid zodanig dat zij bruikbaar wordt voor de predicatenlogica. We spreken over 'uitbreiden' aangezien de regels van de boommethode voor de propositielogica ook mogen worden gebruikt bij de boommethode voor de predicatenlogica. Men kan namelijk voor ieder van de stellingen over vervulbare verzamelingen uit §5.1 een analogon bewijzen voor de predicatenlogica. Dat is niet verwonderlijk als men zich realiseert dat de predicatenlogica een uitbreiding is van de propositielogica. Men verkrijgt deze overeenkomstige stellingen voor de predicatenlogica door in de stellingen in §5.1 overal $PROP$ te vervangen door $FORM$. De bewijzen van deze stellingen laten we aan de lezer over.

De indeling van dit hoofdstuk is als volgt. In §11.1 bewijzen we een aantal stellingen over vervulbare verzamelingen voor de predicatenlogica, terwijl in §11.2 de nieuwe regels voor de boommethode worden geïntroduceerd, waarbij ook voorbeelden van afleidingen worden gegeven. In §11.3 bewijzen we tenslotte de zogenaamde *boomstelling*. Deze stelling zegt dat de boommethode volledig is, dat wil zeggen dat er een sluitende boom voor Γ bestaat als de eindige verzameling Γ onvervulbaar is.

11.1 Theoretische onderbouwing

De volgende stelling is het analogon van stelling 5.1.1 voor de predicatenlogica.

11.1.1 Stelling *Zij $\Gamma \subseteq FORM$ en $F \in FORM$, dan geldt de volgende equivalentie:*

$$\Gamma \models F \quad \Leftrightarrow \quad \Gamma \cup \{\neg F\} \textit{ is niet vervulbaar.}$$

BEWIJS

($\Rightarrow$) Zij $\langle \mathcal{A}, \beta \rangle$ een interpretatie. Dan zijn er twee mogelijkheden:

- Er is een $A \in \Gamma$ met $v^{\mathcal{A},\beta}(A) = 0$. Maar dan is $\langle \mathcal{A}, \beta \rangle$ zeker geen model voor $\Gamma \cup \{\neg F\}$.
- Voor alle $A \in \Gamma$ is $v^{\mathcal{A},\beta}(A) = 1$, ofwel $\langle \mathcal{A}, \beta \rangle$ is een model voor Γ. Uit $\Gamma \models F$ volgt nu dat $\langle \mathcal{A}, \beta \rangle$ ook een model is voor F. Dit betekent dat $v^{\mathcal{A},\beta}(\neg F) = 0$ en dat wederom $\langle \mathcal{A}, \beta \rangle$ geen model is voor $\Gamma \cup \{\neg F\}$.

Hieruit volgt dat geen enkele interpretatie $\langle \mathcal{A}, \beta \rangle$ een model is voor $\Gamma \cup \{\neg F\}$, zodat deze verzameling onvervulbaar is.

($\Leftarrow$) Zij $\langle \mathcal{A}, \beta \rangle$ een model voor Γ. In combinatie met het gegeven dat $\Gamma \cup \{\neg F\}$ onvervulbaar is, levert dit $v^{\mathcal{A},\beta}(\neg F) = 0$. Dit betekent dat $v^{\mathcal{A},\beta}(F) = 1$, zodat $\langle \mathcal{A}, \beta \rangle$ een model is voor F. We concluderen dat $\Gamma \models F$. ■

Merk op dat het bewijs van deze stelling bijna identiek is aan dat van stelling 5.1.1. De volgende stellingen geven aan wanneer stellingen vervulbaar zijn die formules bevatten die kwantoren of het gelijkheidssymbool bevatten.

11.1.2 STELLING *Zij $\Gamma \subseteq FORM$, $A \in FORM$ en $t \in TERM$. Zij verder $\langle \mathcal{A}, \beta \rangle$ een interpretatie. Dan geldt:*

$$\langle \mathcal{A}, \beta \rangle \models \Gamma \cup \{\forall x A\} \quad \Leftrightarrow \quad \langle \mathcal{A}, \beta \rangle \models \Gamma \cup \{\forall x A, A[x:=t]\}.$$

BEWIJS

($\Rightarrow$) *Als $\langle \mathcal{A}, \beta \rangle \models \Gamma \cup \{\forall x A\}$, dan ook $\langle \mathcal{A}, \beta \rangle \models \forall x A$, zodat $\langle \mathcal{A}, \beta[x \mapsto a] \rangle \models A$ voor alle $a \in |\mathcal{A}|$. In het bijzonder is $\langle \mathcal{A}, \beta[x \mapsto t^{\mathcal{A},\beta}] \rangle \models A$, aangezien $t^{\mathcal{A},\beta} \in |\mathcal{A}|$. Hieruit volgt met stelling 10.2.5 dat $\langle \mathcal{A}, \beta \rangle \models A[x:=t]$, zodat $\langle \mathcal{A}, \beta \rangle \models \Gamma \cup \{\forall x A, A[x:=t]\}$.*

($\Leftarrow$) *Triviaal.* ■

11.1.3 STELLING *Zij $\Gamma \subseteq FORM$, $A \in FORM$ en zij c een naam of een variabele die niet voorkomt, respectievelijk niet vrij voorkomt, in de formules in Γ of in A. Dan geldt:*

$$\Gamma \cup \{\exists x A\} \textit{ is vervulbaar} \quad \Leftrightarrow \quad \Gamma \cup \{A[x:=c]\} \textit{ is vervulbaar.}$$

BEWIJS *We bewijzen de stelling alleen voor het geval dat c een naam is. De lezer wordt uitgenodigd zelf het onderstaande bewijs aan te passen voor het geval dat c een variabele is.*

($\Rightarrow$) *Stel dat $\Gamma \cup \{\exists x A\}$ vervuld wordt door de interpretatie $\langle \mathcal{A}, \beta \rangle$. Hieruit volgt dat er een $a \in |\mathcal{A}|$ is zodanig dat $\langle \mathcal{A}, \beta[x \mapsto a] \rangle \models A$. Zij $\langle \mathcal{B}, \beta \rangle$ de interpretatie die gelijk is aan $\langle \mathcal{A}, \beta \rangle$ behalve dat $c^{\mathcal{B}} = a$. Over $c^{\mathcal{A}}$ weten we niets; misschien is $c^{\mathcal{A}} = a$, misschien niet. Uit de stellingen 9.4.2 en 10.2.5 volgt dat $\langle \mathcal{B}, \beta \rangle$ een model voor $\Gamma \cup \{A[x:=c]\}$.*

($\Leftarrow$) *Omgekeerd, als* $\Gamma \cup \{A[x:=c]\}$ *vervuld wordt door, laten we zeggen* $\langle \mathcal{A}, \beta \rangle$*, dan volgt uit stelling 10.2.5 dat* $\langle \mathcal{A}, \beta[x \mapsto c^{\mathcal{A},\beta}] \rangle \models A$*. Aangezien* $c^{\mathcal{A},\beta} \in |\mathcal{A}|$*, betekent dit dat* $\langle \mathcal{A}, \beta \rangle \models \exists x A$*. Hieruit volgt dat* $\Gamma \cup \{\exists x A\}$ *vervuld wordt door* $\langle \mathcal{A}, \beta \rangle$*.* ■

11.1.4 STELLING *Zij* $\Gamma \subseteq FORM$ *en* $t \in TERM$*. Zij verder* $\langle \mathcal{A}, \beta \rangle$ *een interpretatie. Dan geldt:*

$$\langle \mathcal{A}, \beta \rangle \models \Gamma \quad \Leftrightarrow \quad \langle \mathcal{A}, \beta \rangle \models \Gamma \cup \{t = t\}.$$

BEWIJS *Deze eigenschap volgt meteen uit de definities.* ■

11.1.5 STELLING *Zij* $\Gamma \subseteq FORM$*,* $A \in FORM$ *en* $s, t \in TERM$*. Zij verder* $\langle \mathcal{A}, \beta \rangle$ *een interpretatie. Dan geldt:*

$$\langle \mathcal{A}, \beta \rangle \models \Gamma \cup \{s = t, A[x:=s]\} \quad \Leftrightarrow \quad \langle \mathcal{A}, \beta \rangle \models \Gamma \cup \{s = t, A[x:=t]\}.$$

BEWIJS *Dit volgt uit corollarium 10.2.6 en de definities.* ■

Met behulp van de stellingen uit paragraaf 5.1 die ook, zoals eerder is opgemerkt, voor de predicatenlogica gelden, en de stellingen hierboven kunnen we de formules uit een verzameling $\Gamma \subset FORM$ afbreken tot literalen. Net als in het propositionele geval kunnen we nagaan of Γ vervulbaar is. Hierbij tekenen wij aan dat uit stelling 11.1.2 blijkt, dat een formule van de vorm $\forall x A$ steeds opnieuw gebruikt kan worden voor het genereren van formules door het invullen van steeds andere termen t voor de variabele x. Dit betekent dat niet altijd alle formules uit Γ 'definitief' kunnen worden afgebroken tot literalen.

11.1.6 VOORBEELD Vervulbaarheid en onvervulbaarheid.

1. $\models \forall x P(x) \to \neg \exists x \neg P(x)$.
   Dit volgt uit het feit dat de verzameling $\Gamma = \{\neg[\forall x P(x) \to \neg \exists x \neg P(x)]\}$ niet vervulbaar is (zie figuur 11.1). Aangezien de verzameling Γ_5 niet vervulbaar is, volgt door toepassing van de relevante stellingen dat Γ ook niet vervulbaar is.

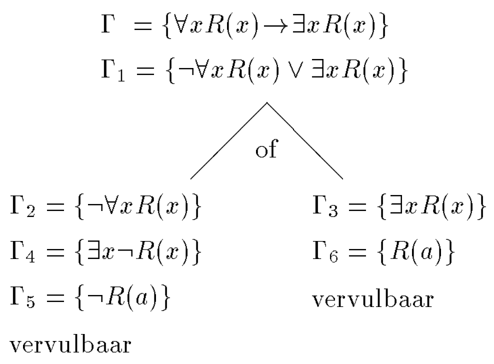
   Ten aanzien van de a die geïntroduceerd wordt in Γ_4, geldt dat deze niet mag voorkomen in de verzameling Γ_3. Of deze a nu een naam of een variabele van de objecttaal is, doet er eigenlijk niet toe (stelling 11.1.3). In het vervolg zullen we in ieder geval de letters $a, b, c, \ldots$ gebruiken en deze als namen behandelen.

2. $\Gamma = \{\forall x R(x) \to \exists x R(x)\}$ is vervulbaar.
   Uit figuur 11.2 blijkt dat de verzamelingen Γ_5 en Γ_6 vervulbaar zijn. Hieruit volgt dat ook Γ vervulbaar is.

$$\begin{aligned}
\Gamma\ &= \{\neg[\forall x P(x) \rightarrow \neg\exists x \neg P(x)]\}\\
\Gamma_1 &= \{\forall x P(x) \wedge \neg\neg\exists x \neg P(x)\}\\
\Gamma_2 &= \{\forall x P(x), \neg\neg\exists x \neg P(x)\}\\
\Gamma_3 &= \{\forall x P(x), \exists x \neg P(x)\}\\
\Gamma_4 &= \{\forall x P(x), \neg P(a)\}\\
\Gamma_5 &= \{\forall x P(x), P(a), \neg P(a)\}
\end{aligned}$$

niet vervulbaar

Figuur 11.1: $\Gamma = \{\neg[\forall x P(x) \rightarrow \neg\exists x \neg P(x)]\}$ is niet vervulbaar.



Figuur 11.2: $\Gamma = \{\forall x R(x) \rightarrow \exists x R(x)\}$ is vervulbaar.

$\forall$-regels		**$\exists$-regels**		**$=$-regel**	**sub-regel**
$\neg\forall xA$	$*\ \forall xA$	$\neg\exists xA$	$\exists xA$		$s=t$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\boldsymbol{\exists x\neg A}$	$\boldsymbol{A[x:=t]}$	$\boldsymbol{\forall x\neg A}$	$\boldsymbol{A[x:=c]}$	$\boldsymbol{t=t}$	$A[x:=s]$
					$\vdots$
					$\boldsymbol{A[x:=t]}$

Tabel 11.1: Aanvullende regels voor de *Boommethode*.

De structuren $\mathcal{A} = \langle D; P; d\rangle$ en $\mathcal{B} = \langle D; Q; d\rangle$ waarbij $D = \{d\}$, $P = \emptyset$ en $Q = \{d\}$, zijn modellen voor de zin $\forall xR(x)\rightarrow\exists xR(x)$.

Immers voor alle bedelingen β geldt $v^{\mathcal{A},\beta}(\forall xR(x)) = 0$, aangezien $R^{\mathcal{A}} = P = \emptyset$ en $D \neq \emptyset$. Hieruit volgt voor alle β dat $\langle\mathcal{A},\beta\rangle \models \forall xR(x)\rightarrow\exists xR(x)$.

Verder geldt voor alle bedelingen β dat $v^{\mathcal{B},\beta}(\exists xR(x)) = 1$, aangezien $d \in R^{\mathcal{B}} = \{d\}$. Hieruit volgt dat $\langle\mathcal{B},\beta\rangle \models \forall xR(x)\rightarrow\exists xR(x)$ voor alle bedelingen β. ■

11.2 Bomen voor de predicatenlogica

In deze paragraaf zullen we de uitbreiding van de *boommethode voor de predicatenlogica* bespreken. De regels voor deze boommethode zijn die uit tabel 5.1 (zie §5.2) aangevuld met die uit tabel 11.1. De regels in tabel 5.1 moeten nu gelezen worden met $A, B \in \mathit{FORM}$. Voor de regels in tabel 11.1 geldt het volgende:

1. De eerste formule(s) is (zijn) de premissen, terwijl de laatste formule, die vet gedrukt is, de conclusie is. Verder geldt dat $A \in \mathit{FORM}$.

2. De ster ($*$) bij de tweede $\forall$-regel duidt aan dat men uit de formule $\forall xA$ voor iedere mogelijke keuze van een term $t \in \mathit{TERM}$ de formule $A[x{:=}t]$ mag afleiden. Dit betekent dat de formule $\forall xA$ meerdere keren mag worden toegepast in een boom (cf. stelling 11.1.2).

   De rechtvaardiging hiervoor is dat als $\forall xA$ waar is, de formule A blijkbaar waar moet zijn voor *iedere keuze* van x, zodat alle mogelijke termen op de plaats van x mogen worden gesubstitueerd in A. In de praktijk dient men zinvolle, dus bruikbare termen te kiezen: het doel is namelijk om een afsluitende boom te construeren.

3. Bij toepassing van de tweede $\exists$-regel dient een tot dan toe nog niet gebruikte naam (of niet gebruikte vrij voorkomende variabele) c te worden geïntroduceerd. Dus c mag niet voorkomen (of vrij voorkomen) in de formules die boven de formule $A[x{:=}c]$ in dezelfde tak van de boom staan (cf. stelling 11.1.3).

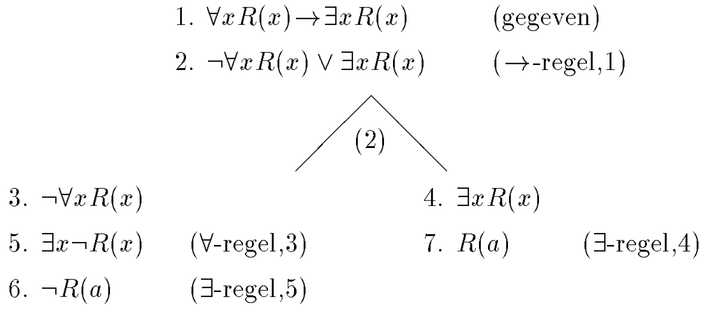
   Het idee achter deze regel is als volgt. Als men weet dat de formule $\exists x A$ waar is, dan moet er een object o bestaan zodanig dat A waar is als de variabele x wordt gebonden aan object o. Omdat we niet weten wat het object o is, maar er wel over willen kunnen redeneren, geven we het een voorlopige naam. We kiezen dan een naam c die nog niet gebruikt is in de tak. We weten immers niets van object o, behalve dat het de door A uitgedrukte eigenschap bezit. Zouden we een reeds gebruikte naam kiezen voor o, dan zouden we het ons onbekende object o identificeren met het object dat de reeds gebruikte naam draagt. Dat is onwenselijk, omdat we, nogmaals gezegd, hoegenaamd niets weten van o.

   We zullen bij toepassing van de boommethode er steeds vanuit gaan dat c een naam is, omdat dit beter bij de intuïtie aansluit: c is een naam voor een individu met de eigenschap A. We spreken dan over een *nieuwe naam* en we nemen aan dat we bij het construeren van bomen over een voldoende hoeveelheid nieuwe namen kunnen beschikken. Het is echter toegestaan om *nieuwe variabelen* te gebruiken.

4. De $=$-regel heeft geen premissen. Zij geeft aan dat de atomaire formule $t = t$ (voor elke $t \in \mathit{TERM}$) op iedere plaats in de boom mag worden toegevoegd (cf. stelling 11.1.4).

5. Ten aanzien van de sub-regel (cf. stelling 11.1.5) merken we het volgende op. Als F een formule is waarin mogelijk de term s voorkomt, en als G uit F is verkregen door sommige voorkomens van s door t te vervangen, dan is dat een geldige toepassing van deze regel. Men kan namelijk in al deze gevallen een formule A vinden zodanig dat $A[x{:=}s] = F$ en $A[x{:=}t] = G$ voor een geschikte keuze van een variabele x. Laat namelijk x een variabele zijn die niet vrij voorkomt in A, s of t, en kies voor A de formule F waarin de voorkomens van s die door t moeten worden vervangen om G te krijgen, zijn vervangen door x.

   Uit bijvoorbeeld $R(a,a)$ en $(a = b)$, mag niet alleen $R(b,b)$, maar ook $R(b,a)$ en $R(a,b)$ worden afgeleid. In het geval dat $R(b,a)$ wordt afgeleid, kan men bijvoorbeeld voor A de formule $R(x,a)$ kiezen, want $R(x,a)[x{:=}a] = R(a,a)$ en $R(x,a)[x{:=}b] = R(b,a)$.



Figuur 11.3: $\Gamma = \{\forall x R(x) \rightarrow \exists x R(x)\}$ is vervulbaar.

6. In $\mathcal{PL}$, $\mathcal{PL}^=$, $\mathcal{PL}^f$ en $\mathcal{PL}^{f=}$ kunnen alle regels voor de kwantoren $\forall$ en $\exists$ uit figuur 11.1 worden toegepast. In $\mathcal{PL}^=$ en $\mathcal{PL}^{f=}$ mogen bovendien de =-regel en de sub-regel worden gebruikt.

Zoals reeds gezegd, nemen we aan dat we over een voldoende grote hoeveelheid nieuwe namen kunnen beschikken voor het gebruik van de tweede $\exists$-regel. Deze zijn niet altijd in de eerste-ordetaal onder beschouwing aanwezig. Derhalve impliceert het gebruik van de boommethode dat we soms een taal moeten uitbreiden met nieuwe namen, of moeten overgaan op het gebruik van variabelen waarvan er altijd aftelbaar oneindig veel zijn. Het uitbreiden van een taal met nieuwe namen heeft wegens de interpretatiestelling (9.4.2) geen gevolgen voor het vervulbaar of onvervulbaar zijn van de oorspronkelijke verzameling Γ, aangezien deze nieuwe namen niet kunnen voorkomen in Γ.

Indien men uitsluitend met nieuwe variabelen werkt, behoeft men de beschouwde taal nooit uit te breiden bij gebruik van de boommethode. Theoretisch gezien is dit eleganter dan het werken met nieuwe namen, maar het is minder intuïtief. Vandaar onze keuze om met nieuwe namen te werken.

11.2.1 VOORBEELD Toepassingen van de boommethode.

1. $\Gamma = \{\forall x R(x) \rightarrow \exists x R(x)\}$ is vervulbaar.
   In figuur 11.3 vindt men de boom, geconstrueerd volgens de boommethode en overeenkomende met de 'boom' uit figuur 11.2 behorende bij voorbeeld 11.1.6.

   Er is een verschil tussen beide typen bomen. Bij bomen die uit *verzamelingen* van formules zijn opgebouwd, worden in die verzamelingen

1.	$\neg[\exists x \forall y P(x,y) \rightarrow \forall y \exists x P(x,y)]$	($\neg$ conclusie)
2.	$\exists x \forall y P(x,y) \wedge \neg \forall y \exists x P(x,y)$	($\rightarrow$-regel,1)
3.	$\exists x \forall y P(x,y)$	($\wedge$-regel,2)
4.	$\neg \forall y \exists x P(x,y)$	($\wedge$-regel,2)
5.	$\exists y \neg \exists x P(x,y)$	($\forall$-regel,4)
6. $*$	$\forall y P(a,y)$	($\exists$-regel,3)
7.	$\neg \exists x P(x,b)$	($\exists$-regel,5)
8. $*$	$\forall x \neg P(x,b)$	($\exists$-regel,7)
9.	$\neg P(a,b)$	($\forall$-regel,8)
10.	$P(a,b)$	($\forall$-regel,6)
X(9,10)		

Figuur 11.4: $\models \exists x \forall y P(x,y) \rightarrow \forall y \exists x P(x,y)$.

steeds alle ongewijzigde formules doorgegeven. Dit in tegenstelling tot bomen die volgens de boomregels zijn geconstrueerd, waarbij alleen de afgebroken formules worden doorgegeven.

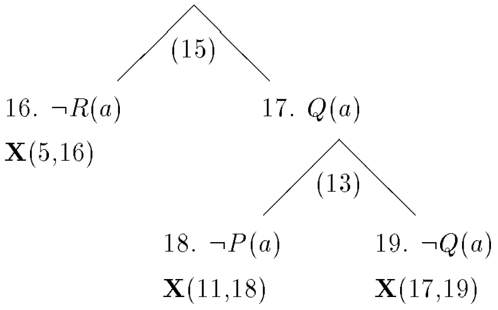
De structuren $\mathcal{A}$ en $\mathcal{B}$ genoemd in voorbeeld 11.1.6(2) zijn modellen voor Γ. Merk op dat de naam a in beide takken is geïntroduceerd. Dit is toegestaan, immers voor de linker tak geldt dat a niet voorkomt in de formules 1, 2, 3 en 5, en voor de rechter tak geldt dat a niet voorkomt in de formules 1, 2 en 4.

2. $\models \exists x \forall y P(x,y) \rightarrow \forall y \exists x P(x,y)$.
   Dit wordt bewezen door de gesloten boom in figuur 11.4.

3. $\forall x[P(x) \rightarrow \neg Q(x)], a = b \models R(a) \rightarrow [\forall x(R(x) \rightarrow Q(x)) \rightarrow \neg P(b)]$.
   De boom in figuur 11.5 laat zien dat de bewering geldig is.

4. *niet* $\forall x \exists y R(x,y) \models \exists y \forall x R(x,y)$.
   De boom voor dit voorbeeld staat in figuur 11.6. Na formule 3 in deze boom, is het alleen mogelijk om de tweede $\forall$-regel toe te passen op formule 1 of 3 (de formules met een $*$). Aangezien bij het gebruik van deze regel iedere term mag worden gesubstitueerd voor x, respectievelijk y, kiezen

1. *	$\forall x[P(x)\rightarrow\neg Q(x)]$	(hypothese 1)
2.	$a=b$	(hypothese 2)
3.	$\neg\{R(a)\rightarrow[\forall x(R(x)\rightarrow Q(x))\rightarrow\neg P(b)]\}$	($\neg$ conclusie)
4.	$R(a)\wedge\neg[\forall x(R(x)\rightarrow Q(x))\rightarrow\neg P(b)]$	($\rightarrow$-regel,3)
5.	$R(a)$	($\wedge$-regel,4)
6.	$\neg[\forall x(R(x)\rightarrow Q(x))\rightarrow\neg P(b)]$	($\wedge$-regel,4)
7.	$\forall x(R(x)\rightarrow Q(x))\wedge\neg\neg P(b)$	($\rightarrow$-regel,6)
8. *	$\forall x(R(x)\rightarrow Q(x))$	($\wedge$-regel,7)
9.	$\neg\neg P(b)$	($\wedge$-regel,7)
10.	$P(b)$	($\neg$-regel,9)
11.	$P(a)$	(sub,2,10)
12.	$P(a)\rightarrow\neg Q(a)$	($\forall$-regel,1)
13.	$\neg P(a)\vee\neg Q(a)$	($\rightarrow$-regel,12)
14.	$R(a)\rightarrow Q(a)$	($\forall$-regel,8)
15.	$\neg R(a)\vee Q(a)$	($\rightarrow$-regel,14)



Figuur 11.5: $\forall x[P(x)\rightarrow\neg Q(x)], a=b\models R(a)\rightarrow[\forall x(R(x)\rightarrow Q(x))\rightarrow\neg P(b)]$.

1. $*$	$\forall x \exists y R(x,y)$	(hypothese)
2.	$\neg\exists y \forall x R(x,y)$	($\neg$ conclusie)
3. $*$	$\forall y \neg\forall x R(x,y)$	($\exists$-regel,2)
4.	$\exists y R(a,y)$	($\forall$-regel,1)
5.	$R(a,b)$	($\exists$-regel,4)
6.	$\neg\forall x R(x,b)$	($\forall$-regel,3)
7.	$\exists x \neg R(x,b)$	($\forall$-regel,6)
8.	$\neg R(c,b)$	($\exists$-regel,7)
	$\vdots$	

Figuur 11.6: *niet* $\forall x \exists y R(x,y) \models \exists y \forall x R(x,y)$.

we er maar één: we substitueren a voor x in formule 1. Deze a kan zowel een naam of een variabele zijn: dat maakt niet uit, zoals we reeds hebben opgemerkt. Omdat vóór formule 4 nog geen nieuwe namen of variabelen zijn geïntroduceerd, is het toevallig dat a nieuw is. Het gebruik van de tweede $\forall$-regel vereist dat niet.

Het is eenvoudig in te zien dat de boom niet sluit. We kunnen weliswaar proberen om de met $*$ gemerkte formules nogmaals te gebruiken, bijvoorbeeld voor de naam c, maar de formules die met $\exists$ beginnen, forceren dan de introductie van weer andere nieuwe namen. Dit voorkomt een afsluiting van de boom.

Om tot een structuur $\mathcal{A}$ te komen zodanig dat $\mathcal{A} \models \forall x \exists y R(x,y)$, maar *niet* $\mathcal{A} \models \exists y \forall x R(x,y)$ 'lezen' we de boom 'af'. Zo'n structuur, wordt een *tegenvoorbeeld* genoemd.

Het 'aflezen' levert op: $R(a,b)$ is waar, en $R(c,b)$ is onwaar. Verder moeten de formules 1 en 3 waar zijn. Dit zijn juist de met $*$ gelabelde formules. Dit levert het volgende tegenvoorbeeld. Zij $\mathcal{A} = \langle D; Q; d_1, d_2, d_3 \rangle$ met d_1, d_2 en d_3 verschillend en $D = \{d_1, d_2, d_3\}$, en zij $Q = \{(d_1, d_2), (d_2, d_2), (d_3, d_3)\}$. Laat nu $R^{\mathcal{A}} = Q$, $a^{\mathcal{A}} = d_1$, $b^{\mathcal{A}} = d_2$ en $c^{\mathcal{A}} = d_3$, dan voldoet $\mathcal{A}$ aan de gestelde eisen: $R(a,b)$ is waar in $\mathcal{A}$ omdat $(d_1, d_2) \in R^{\mathcal{A}}$, en $R(c,b)$ is onwaar in $\mathcal{A}$ omdat $(d_3, d_2) \notin R^{\mathcal{A}}$. Ook moeten we verifiëren dat de formules 1 en 3 waar zijn in $\mathcal{A}$. Formule 1 is waar aangezien voor elk element $d \in D = \{d_1, d_2, d_3\}$ er een element $e \in D$ bestaat zodanig dat $(d, e) \in R^{\mathcal{A}}$: voor d_1 voldoet d_2 omdat $(d_1, d_2) \in R^{\mathcal{A}}$, voor d_2 voldoet d_2 zelf aangezien $(d_2, d_2) \in R^{\mathcal{A}}$, en voor d_3 voldoet op dezelfde wijze d_3 ook. Formule 3 is waar omdat het voor alle elementen $e \in D$ niet zo is dat voor alle elementen $d \in D$ geldt dat $(d, e) \in R^{\mathcal{A}}$: voor $e = d_1$ bevat $R^{\mathcal{A}}$ geen enkel paar van de vorm (d, d_1), voor $e = d_2$ geldt dat $(d_3, d_2) \notin R^{\mathcal{A}}$, en voor $e = d_3$ is een dergelijk paar ook gemakkelijk te vinden. Derhalve is $\mathcal{A}$ een tegenvoorbeeld.

Structuur $\mathcal{A}$ is overigens niet het eenvoudigste tegenvoorbeeld. Definieer $\mathcal{B} = \langle B; S; d_1, d_2 \rangle$ met $d_1 \neq d_2$, $B = \{d_1, d_2\}$, $S = \{(d_1, d_1), (d_2, d_2)\}$, en $R^{\mathcal{B}} = S$, dan is het niet moeilijk te verifiëren dat $\mathcal{B}$ ook een tegenvoorbeeld is. ∎

In het laatste voorbeeld hebben we laten zien hoe uit een boom die niet afsluit een tegenvoorbeeld kan worden ‘afgelezen’. Het construeren van een tegenvoorbeeld in de predicatenlogica is veel lastiger dan in de propositielogica. Dit wordt veroorzaakt door de formules in de boom die met een universele kwantor beginnen. Bij het construeren van een tegenvoorbeeld moeten we namelijk rekening houden met alle formules die met de tweede $\forall$-regel daaruit afleidbaar zijn. Samengevat:

1. Het is vaak niet op het eerste gezicht duidelijk dat een boom *niet* afsluit. Dit geval doet zich in zekere mate voor met betrekking tot de boom in figuur 11.6. Hierbij is het inzicht nodig dat het geen zin meer heeft om de boomregels nog verder toe te passen.

2. Bij het construeren van een tegenvoorbeeld is *creativiteit* nodig. Kon men bij de propositielogica tegenvoorbeelden volledig aflezen uit de boom, in de predicatenlogica kan men slechts de *randvoorwaarden* voor een tegenvoorbeeld uit de boom aflezen. Met name dient bij de constructie van een tegenvoorbeeld rekening te worden gehouden met de universeel gekwantificeerde formules in de openblijvende tak: deze met $*$ gelabelde formules moeten alle waar zijn in het tegenvoorbeeld!

De predicatenlogica is dan ook *niet algoritmisch beslisbaar*. Dit houdt in dat er geen computerprogramma kan worden geschreven dat in eindige tijd voor een willekeurige eindige $\Gamma \subset \mathit{FORM}$ en $F \in \mathit{FORM}$ kan beslissen of $\Gamma \models F$ geldig, dan wel ongeldig is. Dit staat in contrast met de propositielogica die wel algoritmisch beslisbaar is.

De bomen in deze paragraaf hadden alle betrekking op gesloten formules (zinnen). Daarom was het voor het construeren van een tegenvoorbeeld uit een open blijvende tak voldoende om een structuur te geven in plaats van een interpretatie. Immers, in het geval van zinnen mochten interpretaties worden geïdentificeerd met structuren. Werkt men echter met formules, dan bestaat een tegenvoorbeeld uit een structuur en een bedeling voor de vrije variabelen in de formules. Een tegenvoorbeeld is dan een interpretatie.

11.3 De boomstelling

Evenals in de propositielogica geldt in de predicatenlogica de stelling dat een eindige verzameling Γ van formules onvervulbaar is dan en slechts dan als er een boom voor Γ bestaat die sluit. Men zegt ook wel dat de boommethode *volledig* is. In feite zullen we bewijzen dat de boomstelling zelfs geldt voor oneindige verzamelingen van formules Γ. We moeten dan eerst definiëren wat we onder een boom voor een oneindige verzameling formules verstaan.

Het bewijs van de boomstelling is niet eenvoudig vanwege de complicaties die aan het eind van de vorige paragraaf zijn genoemd. In de propositielogica zijn bomen altijd eindige objecten, in de predicatenlogica behoeft dit niet het geval te zijn. Het is mogelijk dat men bij het zoeken naar een sluitende boom in een oneindig lange tak terecht komt, doordat men de regels (met name die voor de universele kwantor) in de verkeerde volgorde toepast, of doordat de boom eenvoudigweg niet sluit.

11.3.1 DEFINITIE **Boom voor oneindige verzameling**
Een boom $B(\Gamma)$ voor een oneindige verzameling van formules Γ ontstaat door een boom voor een eindige deelverzameling $\Gamma_0 \subseteq \Gamma$ te construeren waarbij het op ieder moment in de constructie van de boom toegestaan is om nog niet gebruikte formules uit Γ in een tak op te nemen.

De eindige verzameling Γ_0 uit de bovenstaande definitie, die gebruikt wordt voor de constructie van van een boom $B(\Gamma)$, dient om een uitgangspunt voor de boom te hebben. Gaandeweg mogen andere formules uit Γ bij de constructie worden betrokken. Op deze wijze wordt voorkomen dat we alle formules uit Γ bovenaan in de boom moeten plaatsen, hetgeen al een oneindige tak zou opleveren indien Γ oneindig is. Deze situatie is onwenselijk omdat we graag willen dat op ieder moment van de constructie van $B(\Gamma)$ het tussenresultaat eindig is. Merk op dat het toegestaan is dat $\Gamma_0 = \emptyset$.

Om de boomstelling te bewijzen gaan we uit van zogenaamde *canonieke bomen*. De intuïtie achter dit begrip is dat in een open blijvende tak van een canonieke boom alle regels van de boommethode die kunnen worden toegepast, eens worden toegepast. Zo'n open blijvende tak bevat dan alle informatie die nodig is om een model voor Γ te construeren. Men zou kunnen zeggen dat een open blijvende tak van een canonieke boom 'verstandig' is geconstrueerd, waarmee wordt bedoeld dat de regels in een verstandige volgorde zijn toegepast.

11.3.2 DEFINITIE **Canonieke boom**
Zij $\Gamma \subset FORM$ een verzameling formules en zij $B(\Gamma)$ een boom voor Γ. Dan noemt men $B(\Gamma)$ een canonieke boom, *dan en slechts dan als voor iedere tak van deze boom geldt:*

- *De tak sluit doordat deze een paar complementaire formules F en $\neg F$ bevat, óf*
- *de tak sluit niet en voldoet aan de volgende eigenschap:*

Iedere formule in Γ komt voor in de tak.

Zij $F_0, F_1, F_2, \ldots$ de rij van opeenvolgende formules in deze tak, gerekend vanaf de wortel van $B(\Gamma)$.

Als F_m een formule is waarop een van de regels van de boommethode, met uitzondering van de tweede $\forall$-regel en de $=$-regel, van toepassing is, dan bestaat er een formule F_n $(n > m)$ die het resultaat is van de toepassing van die regel.

Als $F_m = \forall x A$ voor zekere $A \in FORM$ en t een term is die gemaakt kan worden met behulp van de variabelen, functiesymbolen en namen (dus ook de geïntroduceerde nieuwe namen of variabelen) die voorkomen in de formules in de tak, dan moet de formule $A[x:=t]$ voorkomen in de tak.

Als t een term is die gemaakt kan worden met behulp van de variabelen, functiesymbolen en namen die voorkomen in de formules in de tak, dan moet de formule $t = t$ voorkomen in de tak.

Het is niet moeilijk in te zien dat men altijd een canonieke boom kan construeren. Dit wordt aan de lezer overgelaten. Merk op dat het mogelijk is dat een open blijvende tak in een canonieke boom aftelbaar oneindig veel formules bevat.

Aan de andere kant, is iedere sluitende boom eindig. Deze eigenschap volgt uit het feit dat in een sluitende boom iedere tak een eindig aantal formules bevat en een tak zich hooguit kan vertakken in twee subtakken (door toepassing van de $\vee$-vertakkingsregel), en uit een toepassing van *König's Lemma*:

11.3.3 STELLING **König's Lemma**
Iedere zich eindig-vertakkende boom die een oneindig aantal knopen bevat, bezit een tak met een oneindig aantal knopen.

BEWIJS We laten zien hoe in een zich eindig-vertakkende boom B met een oneindig aantal knopen een tak k_0, k_1, ..., k_n, k_{n+1}, ... kan worden verkregen met de eigenschap dat voor iedere $n \in \mathbb{N}$ de knoop k_n de wortel is van een subboom van B met een oneindig aantal knopen. Deze reeks knopen representeert dan een oneindige tak van B.

Neem voor k_0 de wortel van de boom B. Wegens het feit dat B een oneindig aantal knopen bevat, voldoet k_0 aan de gestelde eigenschap.

De inductiehypothese is dat k_n aan de genoemde eigenschap voldoet. Aangezien de subboom van B met wortel k_n zich ook eindig vertakt en wegens de inductiehypothese een oneindig aantal knopen bevat, moet minstens één van de opvolgers van k_n de wortel van een oneindige subboom van B zijn. Noem één van deze knopen k_{n+1}. Deze k_{n+1} voldoet derhalve aan de gestelde eigenschap. ■

11.3.4 STELLING Boomstelling

Voor iedere niet-lege verzameling $\Gamma \subset FORM$ geldt:

$$\Gamma \textit{ is onvervulbaar} \quad \Leftrightarrow \quad \textit{er bestaat een boom } B(\Gamma) \textit{ die sluit.}$$

BEWIJS *($\Leftarrow$): Volgt uit de stellingen uit de paragrafen 5.1 en 11.1.*

($\Rightarrow$): Dit bewijzen we met behulp van contrapositie. Stel dus dat er geen sluitende boom voor Γ bestaat. We zullen laten zien hoe we dan een model $\langle \mathcal{A}, \beta \rangle$ voor Γ kunnen construeren.

Uit het gegeven volgt dat iedere boom voor Γ open blijft. Dit betekent dat ook een canonieke boom voor Γ een niet-sluitende tak T bezit. Formeel zullen we T beschouwen als de verzameling van alle formules in T. We laten zien hoe we een model voor Γ uit T kunnen 'aflezen'. Eerst definiëren we hiertoe een equivalentierelatie R op de verzameling van alle termen die in de formules in T voorkomen. De verzameling van alle equivalentieklassen geïnduceerd door R, wordt het domein $|\mathcal{A}|$ van $\mathcal{A}$. Daarna definiëren we de denotaties van de niet-logische symbolen uit T in overeenstemming met wat we in T 'lezen'. We delen de constructie van het model $\langle \mathcal{A}, \beta \rangle$ op in verschillende stappen.

a. De definitie van de relatie R.

Zij L_T *de eerste-ordetaal met als niet-logische symbolen alle niet-logische symbolen die in de formules in* T *voorkomen en zij* BT *de verzameling van alle termen die in de formules in* T *voorkomen (de boomtermen).*

Definieer de relatie R *als volgt:*

$$R = \{(s,t) \in BT \times BT \mid (s = t) \in T\}$$

b. R is een equivalentierelatie.

We laten zien dat R *reflexief, symmetrisch en transitief is. Zij* $s \in BT$*. Dan is* $(s,s) \in R$*, omdat de formule* $s = s$ *in* T *voorkomt vanwege de canoniciteit van* $B(\Gamma)$*. Dit betekent dat* R *reflexief is.*

Zij $s,t \in BT$ *en* $(s,t) \in R$*. Dit betekent dat de formule* $s = t$ *in* T *voorkomt. Verder is* $(s = s) \in T$ *vanwege de canoniciteit van* $B(\Gamma)$*. Maar dan is ook* $(t = s) \in T$ *omdat de sub-regel vanwege de canoniciteit moet zijn toegepast op* $s = s$ *en* $s = t$*. Hieruit volgt dat* $(t,s) \in R$*, zodat* R *symmetrisch is.*

De transitiviteit van R *kan op analoge wijze worden afgeleid.*

c. De definitie van een interpretatie $\langle \mathcal{A}, \beta \rangle$ voor de taal L_T.

Definieer het domein van $\mathcal{A}$ *als volgt:*

$$|\mathcal{A}| = \{\ [s]_R \mid s \in BT\}.$$

De denotaties van de niet-logische symbolen van L_T *kiezen we als volgt:*

- *Als* P *een* m*-plaatsig predicaatsymbool is, dan*

$$\begin{array}{rcl} P^{\mathcal{A}} & = & \{([s_1]_R, \ldots, [s_m]_R) \mid \\ & & P(s_1, \ldots, s_m) \in T \text{ voor } s_1, \ldots, s_m \in BT\}. \end{array}$$

- *Als* f *een* n*-plaatsig functiesymbool is, dan is* $f^{\mathcal{A}}$ *gedefinieerd door:*

$$f^{\mathcal{A}}([s_1]_R, \ldots, [s_n]_R) = [t]_R,$$

  als $(f(s_1, \ldots, s_n) = t) \in T$ *voor* $s_1, \ldots, s_n, t \in BT$*.*

- *Als* $a \in BT$ *een naam is, dan*

$$a^{\mathcal{A}} \quad = \quad [a]_R.$$

Zij $x \in BT$ *een variabele. Definieer* β *zodanig dat:*

$$\beta(x) \quad = \quad [x]_R.$$

d. $\langle \mathcal{A}, \beta \rangle$ is een interpretatie voor de taal L_T.

Het is niet zonder meer duidelijk dat $\mathcal{A}$ *een structuur is. Met name moeten we laten zien dat* $|\mathcal{A}| \neq \emptyset$ *en dat* $f^{\mathcal{A}}$ *voor alle functiesymbolen* f *een functie op* $|\mathcal{A}|$ *is. Aan de overige vereisten is wegens de definitie van* $\mathcal{A}$ *voldaan.*

Om te beginnen is $|\mathcal{A}| \neq \emptyset$*, omdat* $BT \neq \emptyset$*, hetgeen volgt uit* $\Gamma \neq \emptyset$*. Het zal nu duidelijk zijn waarom we als elementen van* $|\mathcal{A}|$ *equivalentieklassen geïnduceerd door* R *nemen: termen* $s,t \in BT$ *waarvoor in* T *is afgeleid dat* $s = t$ *moeten dezelfde denotatie in* $\langle \mathcal{A}, \beta \rangle$ *krijgen.*

$f^{\mathcal{A}}$ *is een functie als* $f^{\mathcal{A}}$ *overal gedefinieerd is, en als* $f^{\mathcal{A}}$ *functioneel is. We laten eerst zien dat* $f^{\mathcal{A}}$ *functioneel is. Dat wil zeggen dat* $[t_1]_R = [t_2]_R$ *als:*

$$\begin{array}{rcl} f^{\mathcal{A}}([s_1]_R, \ldots, [s_n]_R) & = & [t_1]_R \quad \text{en} \\ f^{\mathcal{A}}([s_1]_R, \ldots, [s_n]_R) & = & [t_2]_R. \end{array}$$

Stel dus dat dit laatste het geval is. Uit de definitie van $f^{\mathcal{A}}$ *volgt dan dat er* $q_i, r_i \in [s_i]_R$ $(1 \leq i \leq n)$ *bestaan zodanig dat:*

$$\begin{array}{lcl} (f(q_1, \ldots, q_n) = t_1) & \in & T \quad en \\ (f(r_1, \ldots, r_n) = t_2) & \in & T. \end{array}$$

Omdat $q_i, r_i \in [s_i]_R$*, moet vanwege de definitie van* R *en de canoniciteit van* $B(\Gamma)$ *gelden dat* $(q_i = r_i) \in T$ $(1 \leq i \leq n)$*. Nogmaals gebruik makend van de canoniciteit van* $B(\Gamma)$ *volgt* $(t_1 = t_2) \in T$ *door een aantal toepassingen van de sub-regel. Dit betekent dat* $[t_1]_R = [t_2]_R$*.*

Er blijft over om te laten zien dat $f^{\mathcal{A}}$ *overal gedefinieerd, dus* totaal *is. Wegens de canoniciteit van* $B(\Gamma)$*, is* $(f(s_1, \ldots, s_n) = f(s_1, \ldots, s_n)) \in T$ *voor alle termen* $s_1, \ldots, s_n \in BT$*. Uit de definitie van* $f^{\mathcal{A}}$ *volgt dan dat* $f^{\mathcal{A}}$ *gedefinieerd is voor alle* $[s_i]_R \in |\mathcal{A}|$ $(1 \leq i \leq n)$*.*

e. Voor alle termen $t \in BT$ geldt $t^{\mathcal{A},\beta} = [t]_R$.

Dit bewijzen we met structurele inductie over de term t*:*

- $t = x$ *voor een variabele* $x \in VAR$*.*

  Dan geldt dat $x^{\mathcal{A},\beta} = \beta(x) = [x]_R$*.*

- $t = c$ *voor een naam* c*.*

  Dan geldt per definitie $c^{\mathcal{A},\beta} = c^{\mathcal{A}} = [c]_R$*.*

- $t = f(s_1, \ldots, s_n)$ *voor* $s_1, \ldots, s_n \in BT$*.*

  Volgens de inductieveronderstelling is ${s_i}^{\mathcal{A},\beta} = [s_i]_R$ $(1 \leq i \leq n)$*. Uit de canoniciteit van* $B(\Gamma)$ *volgt dat:*

  $$(f(s_1, \ldots, s_n) = f(s_1, \ldots, s_n)) \quad \in \quad T.$$

  Dit betekent dat:

  $$f^{\mathcal{A}}([s_1]_R, \ldots, [s_n]_R) \quad = \quad [f(s_1, \ldots, s_n)]_R.$$

  Hieruit en uit de inductiehypothese volgt:

  $$\begin{array}{rcl} f(s_1, \ldots, s_n)^{\mathcal{A},\beta} & = & f^{\mathcal{A}}({s_1}^{\mathcal{A},\beta}, \ldots, {s_n}^{\mathcal{A},\beta}), \\ & = & f^{\mathcal{A}}([s_1]_R, \ldots, [s_n]_R), \\ & = & [f(s_1, \ldots, s_n)]_R. \end{array}$$

f. Voor alle atomen F geldt: $\quad F \in T \quad \Leftrightarrow \quad \langle \mathcal{A}, \beta \rangle \models F$.

Er zijn twee mogelijkheden voor F*:*

- $F = (s = t)$ *voor* $s, t \in BT$*.*

  Als $(s = t) \in T$*, dan is* $[s]_R = [t]_R$*. Verder geldt dat* $s^{\mathcal{A},\beta} = [s]_R$ *en* $t^{\mathcal{A},\beta} = [t]_R$ *wegens bewering (e) hierboven. Nu volgt meteen dat* $\langle \mathcal{A}, \beta \rangle \models s = t$*.*

  Omgekeerd, als $\langle \mathcal{A}, \beta \rangle \models s = t$*, dan* $s^{\mathcal{A},\beta} = t^{\mathcal{A},\beta}$*, zodat* $[s]_R = [t]_R$*. Dit laatste is alleen mogelijk als* $(s = t) \in T$*.*

- $F = P(s_1, \ldots, s_m)$ *voor* $s_1, \ldots, s_m \in BT$*.*

  Als $P(s_1, \ldots, s_m) \in T$*, dan is* $([s_1]_R, \ldots, [s_m]_R) \in P^{\mathcal{A}}$ *wegens de definitie van* $\mathcal{A}$*. Uit bewering (e) volgt* $[s_i]_R = {s_i}^{\mathcal{A},\beta}$ $(1 \leq i \leq m)$*, zodat* $\langle \mathcal{A}, \beta \rangle \models P(s_1, \ldots, s_m)$*.*

  Omgekeerd, als gegeven is dat$\langle \mathcal{A}, \beta \rangle \models P(s_1, \ldots, s_m)$*, dan leiden we af dat* $({s_1}^{\mathcal{A},\beta}, \ldots, {s_m}^{\mathcal{A},\beta}) \in P^{\mathcal{A}}$*. Daar* ${s_i}^{\mathcal{A},\beta} = [s_i]_R$ $(1 \leq i \leq m)$*, moet derhalve* $([s_1]_R, \ldots, [s_m]_R) \in P^{\mathcal{A}}$*. Hieruit volgt dat* $P(s_1, \ldots, s_m) \in T$ *aangezien* $B(\Gamma)$ *canoniek is (ga na).*

g. $\langle \mathcal{A}, \beta \rangle$ is een model voor Γ.
We zullen nu bewijzen dat alle formules $F \in FORM$ de volgende eigenschap bezitten:

$$F \in T \quad \Rightarrow \quad \langle \mathcal{A}, \beta \rangle \models F. \tag{$*$}$$

Als dit is bewezen volgt $\langle \mathcal{A}, \beta \rangle \models \Gamma$, aangezien $\Gamma \subseteq T$. Het bewijs verloopt via volledige inductie naar de complexiteit $comp(F)$ van de formules $F \in FORM$. Neem aan dat $F \in T$.

- *Basisstap: $comp(F) = 0$.*

  Dan is F een atoom en geldt eigenschap $(*)$ wegens bewering (f) hierboven.

- *Inductiestap: $comp(F) = n + 1$.*

  De inductiehypothese luidt dat voor alle formules $F \in T$ met complexiteit $comp(F) \leq n$ geldt $\langle \mathcal{A}, \beta \rangle \models F$.

  We behandelen alleen het geval dat $F = \neg A$. Dit geval is het meest gecompliceerd. Om de inductiestap te voltooien voor de andere gevallen, behoeft men slechts de verschillende stappen voor het onderhavige geval aan te passen.

  Zij dus $F = \neg A$ voor $A \in FORM$. We onderscheiden de volgende gevallen voor A:

  - *$A \in ATOM$.*
    Als $\neg A \in T$, dan $A \notin T$ omdat T niet sluit. Uit bewering (f) volgt nu niet *$\langle \mathcal{A}, \beta \rangle \models A$, zodat $\langle \mathcal{A}, \beta \rangle \models \neg A$.*
  - *$A = \neg C$ voor $C \in FORM$.*
    Uit de canoniciteit van $B(\Gamma)$ volgt dat $C \in T$. Aangezien $comp(C) \leq n$ kunnen we de inductiehypothese op C toepassen, hetgeen levert $\langle \mathcal{A}, \beta \rangle \models C$. Hieruit volgt $\langle \mathcal{A}, \beta \rangle \models \neg\neg C$.
  - *$A = (C \vee D)$ voor $C, D \in FORM$.*
    Vanwege de canoniciteit van $B(\Gamma)$ moet $(\neg C \wedge \neg D) \in T$, en ook $\neg C \in T$ en $\neg D \in T$. Omdat $comp(\neg C) \leq n$ en $comp(\neg D) \leq n$, volgt uit de inductiehypothese $\langle \mathcal{A}, \beta \rangle \models \neg C$ en $\langle \mathcal{A}, \beta \rangle \models \neg D$. Hieruit kan worden afgeleid $\langle \mathcal{A}, \beta \rangle \models \neg(C \vee D)$.
  - *$A = (C \wedge D)$ voor $C, D \in FORM$.*
    Uit de canoniciteit van $B(\Gamma)$ volgt dat $(\neg C \vee \neg D) \in T$. Dit betekent dat ofwel $\neg C \in T$, ofwel $\neg D \in T$ (na splitsing van de tak). Hieruit leiden we met behulp van de inductiehypothese af dat ofwel $\langle \mathcal{A}, \beta \rangle \models \neg C$, ofwel $\langle \mathcal{A}, \beta \rangle \models \neg D$. In beide gevallen volgt $\langle \mathcal{A}, \beta \rangle \models \neg(C \wedge D)$.
  - *$A = (C \rightarrow D)$ of $A = (C \leftrightarrow D)$ voor $C, D \in FORM$.*
    Analoog aan de vorige gevallen.
  - *$A = \exists x C$ voor $C \in FORM$.*
    Wederom vanwege de canoniciteit van $B(\Gamma)$ geldt $\forall x \neg C \in T$. Dit betekent dat voor alle termen $t \in BT$ geldt $\neg C[x := t] \in T$. Met behulp van de inductiehypothese leiden we hieruit af dat $\langle \mathcal{A}, \beta \rangle \models \neg C[x := t]$ voor alle termen $t \in BT$.
    Passen we op dit laatste de substitutiestelling (10.2.5) toe, dan verkrijgen we:

    $$\forall t \in BT \cdot \langle \mathcal{A}, \beta[x := t^{\mathcal{A}, \beta}] \rangle \models \neg C.$$

    Aangezien $t^{\mathcal{A}, \beta} = [t]_R$ voor alle $t \in BT$, en $[t]_R \in |\mathcal{A}|$ dan en slechts dan als $t \in BT$, volgt:

    $$\forall d \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x := d] \rangle \models \neg C,$$

    zodat $\langle \mathcal{A}, \beta \rangle \models \forall x \neg C$, waaruit weer volgt $\langle \mathcal{A}, \beta \rangle \models \neg \exists x C$.

– *$A = \forall x C$ voor $C \in FORM$.*
Analoog aan het vorige geval.

In alle gevallen geldt dus eigenschap $(*)$ voor $F = \neg A$.

Hiermee is aangetoond dat $\langle \mathcal{A}, \beta \rangle \models \Gamma$. ∎

11.4 Opgaven

1. Zij A, B en C éénplaatsige predicaatsymbolen en R een tweeplaatsig predicaatsymbool. Ga met behulp van de Boommethode voor elk van de volgende metabeweringen na of deze juist dan wel onjuist is. Construeer voor elke onjuiste metabewering een tegenvoorbeeld.
   (a) $\models \exists x \forall y R(x,y) \rightarrow \forall y \exists x R(x,y)$.
   (b) $\models \forall x[C(x) \rightarrow \exists y C(y)]$.
   (c) $\forall x[A(x) \rightarrow B(x)] \models \forall x\{\exists y[A(x) \wedge R(x,y)] \rightarrow \exists y[B(x) \wedge R(x,y)]\}$.
   (d) $\models \forall x[\exists y C(y) \rightarrow C(x)]$.
   (e) $\models \exists x[A(x) \wedge B(x)] \leftrightarrow [\exists x A(x) \wedge \exists x B(x)]$.
   (f) $\models [\forall x A(x) \rightarrow \exists x B(x)] \rightarrow \exists x[A(x) \rightarrow B(x)]$.
   (g) $\models \exists x[A(x) \rightarrow B(x)] \rightarrow [\exists x A(x) \rightarrow \forall x B(x)]$.
   (h) $\models [\forall x A(x) \rightarrow \forall y B(y)] \rightarrow \exists x \forall y[A(x) \rightarrow B(y)]$.
   (i) $\forall x \forall y[R(x,y) \rightarrow A(x)] \models \forall x[\forall y R(x,y) \rightarrow A(x)]$.
   (j) $\forall x \exists y[R(x,y) \rightarrow A(x)] \models \forall x[\exists y R(x,y) \rightarrow A(x)]$.
   (k) $\forall x \forall y[A(x) \vee R(x,y)] \models \forall x A(x) \vee \forall x \forall y R(x,y)$.

2. Beschouw de volgende redenering (zie ook de opgaven van hoofdstuk 8).

   > *People are prejudiced against anyone who is liked by someone they dislike.*
   > *But nobody is prejudiced against himself.*
   > *So people don't like anyone who dislikes them.*

   Ga met behulp van de boommethode na of deze redenering logisch geldig is. Construeer een tegenvoorbeeld als de redenering niet logisch geldig is.

3. Voltooi het bewijs van stelling 11.1.3.

4. Beschrijf een methode voor het produceren van canonieke bomen.

5. Bewijs dat de relatie R uit het bewijs van de boomstelling transitief is.

6. Voltooi het inductiebewijs van de bewering dat $\langle \mathcal{A}, \beta \rangle \models \Gamma$ in het bewijs van de boomstelling.

Hoofdstuk 12

Natuurlijke Deductie voor de Predicatenlogica

Het onderwerp van dit hoofdstuk is de natuurlijke deductiemethode van Fitch voor de predicatenlogica. In paragraaf 12.1 introduceren we de afleidingsregels van deze methode, in paragraaf 12.2 geven we voorbeelden van afleidingen, en tenslotte behandelen we in paragraaf 12.3 het *uitgebreide systeem van Fitch.*

12.1 Afleidingsregels

Het systeem van Fitch $\mathcal{F}$ voor de predicatenlogica is een uitbreiding van dat voor de propositielogica. De afleidingsregels uit tabel 6.1 behoren dus ook tot het systeem voor de predicatenlogica met dien verstande dat men nu voor A en B formules uit $FORM$ dient te lezen. Daarnaast is er een aantal nieuwe regels voor de kwantoren en de gelijkheid. Deze staan in de tabellen 12.1 en 12.2. Hierin duiden de vet gedrukte formules de conclusies aan, terwijl de overige formules de premissen zijn. Verder geldt dat $A, B \in FORM$, $s, t \in TERM$, en dat c een naam (of variabele) is. Alvorens in definitie 12.1.3 formeel te specificeren wanneer de regels toepasbaar zijn, geven we een informele introductie.

In deze definitie gebruiken we de noties *nieuwe naam* en *nieuwe variabele* (deze begrippen werden ook gedefinieerd, zij het anders, ten behoeve van de boommethode, in hoofdstuk 11). Men noemt c een nieuwe naam in de formule $A[x{:=}c]$ met betrekking tot A, als c een naam is die niet voorkomt in A of in de aktieve hypothesen voorafgaand aan $A[x{:=}c]$ (voor definities van de begrippen *voorafgaand aan* en *aktieve-hypothesenverzameling* wordt respectievelijk verwezen naar de definities 6.1.1 en 7.2.1, en de informele introducties van deze begrippen voorafgaand aan deze definities).

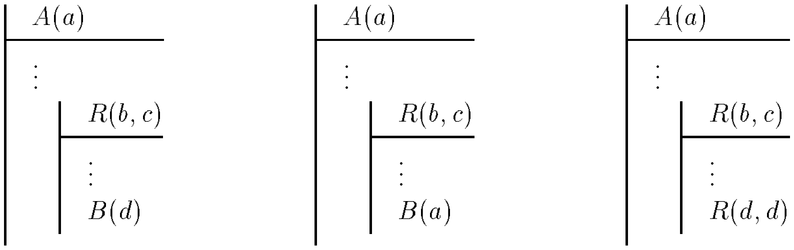
Als c een variabele is, dan noemt men c een nieuwe variabele in de formule $A[x{:=}c]$ met betrekking tot A, indien c niet vrij voorkomt in A of in de aktieve hypothesen voorafgaand aan $A[x{:=}c]$, meer precies geformuleerd:

∀-intro	**∀-elim**	**∃-intro**	**∃-elim**
$A[x{:=}c]$ $\vdots$ $\boldsymbol{\forall x A}$	$\forall x A$ $\vdots$ $\boldsymbol{A[x{:=}t]}$	$A[x{:=}t]$ $\vdots$ $\boldsymbol{\exists x A}$	$\exists x A$ $\vdots$ $c : A[x{:=}c]$ $\vdots$ B $\vdots$ $\boldsymbol{B}$

Tabel 12.1: De regels voor de kwantoren.

=-intro	**sub**
$\vdots$ $\boldsymbol{t = t}$	$s = t$ $\vdots$ $A[x{:=}s]$ $\vdots$ $\boldsymbol{A[x{:=}t]}$

Tabel 12.2: De regels voor de identiteit.



Figuur 12.1: Bewijsfiguren bij voorbeeld 12.1.2.

12.1.1 DEFINITIE **Nieuwe naam en nieuwe variabele**
Zij gegeven een bewijsfiguur $\mathbf{D}$ *met interval* D *en laat verder gegeven zijn de formule* $F_i = A[x:=c]$ $(i \in D)$.

1. *Een naam* c *is een* nieuwe naam *in* $A[x:=c]$ *met betrekking tot* A*, indien* c *niet voorkomt in* A *of in een formule van* $\mathbf{A}(i) \cap \{F_j \mid 0 \leq j < i\}$.
2. *Een variabele* c *is een* nieuwe variabele *in* $A[x:=c]$ *met betrekking tot* A*, indien* c *niet vrij voorkomt in* A *of in een formule van* $\mathbf{A}(i) \cap \{F_j \mid 0 \leq j < i\}$.

Merk op dat het feit of c een nieuwe naam of variabele is in de formule $A[x:=c]$, afhankelijk is van een bewijsfiguur, van de plaats van deze formule in die bewijsfiguur en van de formule A. Merk ook op dat de verzameling $\mathbf{A}(i) \cap \{F_j \mid 0 \leq j < i\}$ precies die aktieve hypothesen van formule F_i bevat die aan F_i *voorafgaan*.

12.1.2 VOORBEELD Nieuwe namen.
Beschouw de bewijsfiguren in figuur 12.1.

- In de linker bewijsfiguur is d een nieuwe naam in de formule $B(d)$ met betrekking tot $B(x)$. De formule $B(d)$ kan namelijk geschreven worden als $B(x)[x:=d]$, en verder komt de naam d niet voor in de aktieve hypothesen voorafgaand aan $B(d)$, te weten $A(a)$ en $R(b,c)$, en tenslotte komt d niet voor in $B(x)$.
- In de middelste bewijsfiguur is a *geen* nieuwe naam in $B(a)$ met betrekking tot $B(x)$. Immers, de naam a komt al voor in de aktieve hypothese $A(a)$.
- Voor de rechter bewijsfiguur ligt de zaak wat genuanceerder:
  - De naam d is een nieuwe naam in $R(d,d)$ met betrekking tot de formule $R(x,x)$. Dit volgt uit het feit dat $R(d,d) = R(x,x)[x:=d]$, en dat d niet voorkomt in de aktieve hypothesen $A(a)$ en $R(b,c)$, en ook niet in $R(x,x)$.
  - Schrijven we echter $R(d,d)$ als $R(x,d)[x:=d]$, dan moeten we concluderen dat d *geen* nieuwe naam is in $R(d,d)$ met betrekking tot $R(x,d)$ aangezien d reeds voorkomt in $R(x,d)$.

1.	$\triangle ABC$ is gelijkbenig	(hypothese)
$\vdots$	$\vdots$	
n.	$\triangle ABC$ heeft gelijke basishoeken	$(\ldots)$
$n{+}1$.	$\triangle ABC$ is gelijkbenig $\rightarrow$ $\triangle ABC$ heeft gelijke basishoeken	($\rightarrow$-intro,1,n)
$n{+}2$.	$\forall x[x$ is gelijkbenig $\rightarrow$ x heeft gelijke basishoeken$]$	($\forall$-intro,$n+1$)

Figuur 12.2: Structuur van een wiskundige redenering met ∀-intro.

Het begrip nieuwe naam of variabele wordt gebruikt in de afleidingsregels ∀-intro en ∃-elim (zie tabel 12.1). Het idee achter dit begrip is dat een nieuwe naam of variabele verwijst naar een *willekeurig* individu. Omdat een nieuwe naam of variabele niet voorkomt, respectievelijk vrij voorkomt in de voorafgaande hypothesen, zijn er geen aannamen over het individu waarnaar die nieuwe naam verwijst. Dit maakt dat het genoemde individu willekeurig is: we hebben er vooraf geen eisen aan gesteld. Bij de onderstaande behandeling van de afleidingsregels gaan we hier nog nader op in.

1. De regel ∀-intro brengt tot uitdrukking dat $\forall x A$ is afgeleid, als voor een *willekeurig* individu c is bewezen dat c de door A uitgedrukte eigenschap bezit, ofwel wanneer $A[x:=c]$ is afgeleid voor een naam of variabele c die nieuw is met betrekking tot A. Verder mag de formule $A[x:=c]$ waarop de regel ∀-intro wordt toegepast zelf geen hypothese zijn.

   Deze wijze van redeneren vinden we ook in de wiskunde terug. Als we bijvoorbeeld willen bewijzen dat in *alle* gelijkbenige driehoeken de basishoeken gelijk zijn, dan redeneren we als volgt: zij $\triangle ABC$ een willekeurige gelijkbenige driehoek. Vervolgens proberen uit deze aanname af te leiden dat de basishoeken gelijk zijn. Als dit ons gelukt is, hebben we afgeleid: als $\triangle ABC$ gelijkbenig is, dan zijn de basishoeken van $\triangle ABC$ gelijk. In deze uitspraak is $\triangle ABC$ een nieuwe naam met betrekking tot de formule $[x$ is gelijkbenig $\rightarrow$ x heeft gelijke basishoeken$]$. Dit kan worden geverifieerd door naar de structuur van het bewijs te kijken in figuur 12.2. Er zijn namelijk geen actieve hypothesen voor deze formule en ook komt de naam $\triangle ABC$ er niet in voor. Dit betekent dat de regel ∀-intro mag worden toegepast, waaruit de te bewijzen stelling volgt.

2. Bij de regel ∀-elim mag uit de premisse $\forall x A$ de conclusie $A[x:=t]$ worden afgeleid, waarin t een term naar keuze is. Dit klopt met de intuïtie. Als $\forall x A$ waar is, dan geldt de eigenschap uitgedrukt door A voor alle

1.	polynoom p heeft een nulpunt	(hypothese 1)
2.	$\exists x[p(x) = 0]$	(uit 1)
3.	$n : p(n) = 0$	(hypothese 2)
4.	$p^2(n) = 0$	(uit 3)
5.	$\exists x[p^2(x) = 0]$	($\exists$-intro,4)
6.	$\exists x[p^2(x) = 0]$	($\exists$-elim,2,3,5)
7.	p^2 heeft een nulpunt	(uit 6)
8.	p heeft nulpunt $\rightarrow$ p^2 heeft nulpunt	($\rightarrow$-intro,1,7)
9.	$\forall x[x$ heeft nulpunt $\rightarrow$ x^2 heeft nulpunt$]$	($\forall$-intro,8)

Figuur 12.3: Structuur van een wiskundige redenering met $\exists$-elim.

mogelijke individuen, dus in het bijzonder voor de individuen waarnaar termen verwijzen. Deze regel is vergelijkbaar met de betreffende regel van de boommethode (zie vorige hoofdstuk).

3. De regel $\exists$-intro heeft de volgende rechtvaardiging. Als voor een term t de formule $A[x{:=}t]$ is afgeleid, dan bestaat er blijkbaar een individu met de eigenschap die door A wordt uitgedrukt: namelijk het individu waarnaar t verwijst. Dit betekent dat $\exists xA$ kan worden geconcludeerd.

4. De redeneertrant bij de regel $\exists$-elim is de volgende. Stel we weten dat $\exists xA$, ofwel dat er een individu bestaat met de eigenschap die door A wordt uitgedrukt. We weten niet welk individu dat is. Laten we het voor het moment c noemen, zodat $A[x{:=}c]$ het geval is, waarbij c een nieuwe naam (of variabele) is in $A[x{:=}c]$ met betrekking tot A. Als we nu uit deze uitspraak een bewering B kunnen afleiden die niet afhankelijk is van de keuze van de naam (of variabele) c, met andere woorden c komt niet (vrij) voor in B, dan hebben we dus uit $\exists xA$ de bewering B afgeleid. De in de redenering gebruikte c is hier een tijdelijke naam voor het individu waarvan wordt gezegd dat het bestaat.

   Ook deze wijze van redeneren komt men vaak in de wiskunde tegen. We geven hier een heel eenvoudig voorbeeld. Stel we willen het volgende bewijzen: als een polynoom een nulpunt heeft, dan heeft het kwadraat ervan ook een nulpunt. We redeneren dan als volgt. Zij p een polynoom dat een nulpunt bezit. Deze uitspraak kan worden geschreven als $\exists x[p(x) = 0]$. We moeten dan aantonen dat $\exists x[p^2(x) = 0]$. Zij nu n een nulpunt van p, met andere woorden zij $p(n) = 0$. Hieruit volgt dat $p^2(n) = 0$. Er bestaat dus een x zodanig dat $p^2(x) = 0$, ofwel $\exists x[p^2(x) = 0]$. Anders gezegd, p^2 bezit ook een nulpunt, hetgeen te bewijzen was. Een en ander is schematisch weergegeven in figuur 12.3. Aan alle voorwaarden voor het gebruik van ∃-elim is voldaan: n is een nieuwe naam in $p(n) = 0$ met betrekking tot $p(x) = 0$, en n komt niet voor in de formule $\exists x[p^2(x) = 0]$. Merk op dat deze afleiding ook een voorbeeld van de regels ∃-intro en ∀-intro bevat.

5. De regels voor de indentiteit (zie tabel 12.2) komen overeen met de regels voor de identiteit voor de boommethode. De regel =-intro spreekt voor zich. De sub-regel drukt uit dat als de premissen $s = t$ en $A[x:=s]$ zijn afgeleid, de conclusie $A[x:=t]$ kan worden afgeleid. In de praktijk betekent dat het volgende. Stel dat F een formule is waarin de term s voorkomt, en G een formule verkregen uit F door daarin sommige, maar niet noodzakelijk alle, voorkomens van s door t te vervangen. Dan staat de sub-regel toe om uit de premissen $s = t$ en F de formule G af te leiden. Zie ook de discussie betreffende de hiermee corresponderende regel voor de boommethode in paragraaf 11.2.

In de onderstaande definitie wordt het bovenstaande formeel vastgelegd.

12.1.3 DEFINITIE **Toepassing van een afleidingsregel**
Zij gegeven een bewijsfiguur $\mathbf{D}$ *met interval* $D = [1,n]$, *formules* $F_1, \ldots, F_n$ *en hypothese-intervallen* $\mathbf{H}$. *Dan is formule* E *het resultaat van een* toepassing *van de afleidingsregel* R, *indien* E *de conclusie is van* R, *de premissen van* R *voorafgaan aan* E, *en indien voldaan is aan één van de volgende voorwaarden:*

1. $R \in \{\forall\text{-}elim, \exists\text{-}intro, =\text{-}intro, sub\}$.
   In dit geval moeten de premissen en de conclusie E *alle in* hetzelfde *interval liggen. De volgorde waarin de premissen voorkomen, mag daarbij afwijken van de volgorde zoals aangegeven in de tabellen 12.1 en 12.2. De regels =-intro en sub zijn alleen toepasbaar in* $\mathcal{PL}^=$ *en* $\mathcal{PL}^{f=}$.
2. $R = \forall\text{-}intro$.
   In dit geval moeten de premisse $A[x:=c]$ *en de conclusie* $E = \forall xA$ *in* hetzelfde *interval liggen. Verder dient* c *een* nieuwe naam *of* nieuwe variabele *in* $A[x:=c]$ *te zijn. Bovendien mag de premisse* $A[x:=c]$ *géén hypothese zijn.*
3. $R = \exists\text{-}elim$.
   Er moet een hypothese-interval $[k,l] \in \mathbf{H}$ *bestaan zodanig dat* $F_k = A[x:=c]$ *en* $F_l = B$. *Verder moet gelden dat de conclusie* B, *de premisse* $\exists xA$ *en het interval* $[k,l]$ *in* hetzelfde *interval liggen. Voor* c *geldt de restrictie dat* c *een* nieuwe naam *(of nieuwe variabele) moet zijn in* $A[x:=c]$ *die niet voorkomt (respectievelijk vrij voorkomt) in* B. *Merk op dat dit impliceert dat* c *niet voorkomt (of vrij voorkomt) in* $\exists xA$.

   Nota Bene, de gebruikte notatie 'c *: ' in de premisse* $c : A[x:=c]$ *is slechts een hulpmiddel om duidelijk te maken dat hier sprake is van een hypothese voor ∃-elim. Dit betekent dat* $A[x:=c]$ *verder wordt behandeld als een normale hypothese.*

Evenals bij de boommethode zullen we bij de toepassing van het systeem van Fitch werken met nieuwe namen en aannemen dat we over een voldoende

1. $\quad \left| \; \underline{P(a)} \right.$ (hypothese)
2. $\quad \left| \; \forall x P(x) \right.$ ($\forall$-intro,1) **fout!**
3. $P(a) \rightarrow \forall x P(x)$ ($\rightarrow$-intro,1,2)

Figuur 12.4: Illegale toepassing $\forall$-intro.

grote hoeveelheid nieuwe namen beschikken. Als de eerste-ordetaal onder beschouwing geen voldoende hoeveelheid namen bezit, dienen we deze in feite uit te breiden met namen. Bij het gebruik van nieuwe variabelen is het nooit nodig om een taal uit te breiden met nieuwe namen.

De definities van de begrippen *afleiding met/zonder hypothesen* en *afleidbaarheid* zijn volkomen identiek aan de desbetreffende definities voor de propositielogica (zie definities 6.1.3, 6.1.4 en 6.1.5).

Uit de volgende voorbeelden blijkt, dat men bij het toepassen van de regels $\forall$-intro en $\exists$-elim zorgvuldig moet nagaan of aan de restricties betreffende de naam c is voldaan. Als niet aan deze restricties is voldaan, dan verkrijgt men ongewenste resultaten.

12.1.4 VOORBEELD **Illegale** toepassingen van afleidingsregels.

1. In figuur 12.4 is de afleidingsregel $\forall$-intro **verkeerd** toegepast, $P(a)$ mag immers geen hypothese zijn. Het resultaat is een formule die niet algemeen geldig is. Zij namelijk $\mathcal{R} = \langle \mathbb{R}; S; 5 \rangle$ een structuur zodanig dat $S \subset \mathbb{R}$ de verzameling der positieve reële getallen voorstelt. Zij verder $P^{\mathcal{R}} = S$ en $a^{\mathcal{R}} = 5$, dan geldt *niet* dat $\mathcal{R} \models P(a) \rightarrow \forall x P(x)$: uit het gegeven dat 5 een positief reëel getal is, volgt niet dat alle reële getallen positief zijn.

2. In de bewijsfiguur in figuur 12.5 is de regel $\exists$-elim **foutief** toegepast. De naam a is niet nieuw in $R(a, a)$ ($= A[y{:=}a]$) met betrekking tot $R(a, y)$ ($= A$), want a komt voor in $R(a, y)$. Zij $\mathcal{R} = \langle \mathbb{R}; < \rangle$ de structuur der reële getallen met $R^{\mathcal{R}} = <$, de 'kleiner dan'-relatie. Nu geldt *niet* $\mathcal{R} \models \forall x \exists y R(x, y) \rightarrow \exists x R(x, x)$. Dus ook hier is de foutief afgeleide formule niet algemeen geldig. ∎

12.2 Afleidingsstrategieën en afleidingen

Bij de methode van Fitch voor de propositielogica hebben we een aantal vuistregels gegeven om formules van een bepaalde vorm af te leiden. We zullen

1.	$\forall x \exists y R(x,y)$	(hypothese 1)
2.	$\exists y R(a,y)$	($\forall$-elim,1)
3.	$a : \ R(a,a)$	(hypothese 2) **fout!**
4.	$\exists x R(x,x)$	($\exists$-intro,3)
5.	$\exists x R(x,x)$	($\exists$-elim,2,3,4) **fout!**
6.	$\forall x \exists y R(x,y) \rightarrow \exists x R(x,x)$	($\rightarrow$-intro,1,5)

Figuur 12.5: Illegale toepassing $\exists$-elim.

hetzelfde doen voor de predicatenlogica. De vuistregels uit paragraaf 6.2 kunnen worden aangevuld met de onderstaande twee. Neem aan dat we een formule $F \in \mathit{FORM}$ moeten afleiden. We onderscheiden de volgende gevallen:

1. $F = \forall x A$.
   Probeer voor een willekeurige, dus nieuwe naam (of variabele) c de formule $A[x:=c]$ af te leiden. Pas tenslotte de regel $\forall$-intro toe, zodat $\forall x A$ wordt verkregen.

2. $F = \exists x A$.
   Open een nieuw hypothese-interval met $\neg\exists x A$ als hypothese en probeer hieruit een contradictie B en $\neg B$ af te leiden. Het vinden van geschikte formules B en $\neg B$ is dan nog wel een probleem. Door toepassing van de regel $\neg$-intro, gevolgd door $\neg$-elim verkrijgt men dan $\exists x A$.

   Soms kan $\exists x A$ worden afgeleid door eerst voor een term t de formule $A[x:=t]$ af te leiden, waarna met $\exists$-intro het gewenste resultaat volgt. Dit geval treedt meestal op indien één van de hypothesen van de vorm $\exists y B$ is: het elimineren van $\exists y$ wordt dan gecombineerd met het introduceren van $\exists x$ (zie bijvoorbeeld in figuur 12.8 de afleiding van $\exists y[D(y) \wedge S(a,y)]$ in regel 11 uit $\exists y[P(y) \wedge S(a,y)]$ in regel 2).

De rest van deze paragraaf is gewijd aan voorbeelden van afleidingen.

12.2.1 STELLING *Zij $F \in \mathit{FORM}$, dan geldt:*

1. $\vdash \neg\forall x F \leftrightarrow \exists x \neg F$.

2. $\vdash \neg\exists x F \leftrightarrow \forall x \neg F$.

BEWIJS Zie respectievelijk de figuren 12.6 en 12.7. In de betreffende afleidingen hebben we $F[x:=a]$ steeds afgekort als $F(a)$. ∎

1.	$\mid\ \neg\forall xF$	(hypothese 1)
2.	$\mid\ \mid\ \neg\exists x\neg F$	(hypothese 2)
3.	$\mid\ \mid\ \mid\ \neg F(a)$	(hypothese 3)
4.	$\mid\ \mid\ \mid\ \exists x\neg F$	($\exists$-intro,3)
5.	$\mid\ \mid\ \mid\ \neg\exists x\neg F$	(rei,2)
6.	$\mid\ \mid\ \neg\neg F(a)$	($\neg$-intro,3,4,5)
7.	$\mid\ \mid\ F(a)$	($\neg$-elim,6)
8.	$\mid\ \mid\ \forall xF$	($\forall$-intro,7)
9.	$\mid\ \mid\ \neg\forall xF$	(rei,1)
10.	$\mid\ \neg\neg\exists x\neg F$	($\neg$-intro,2,8,9)
11.	$\mid\ \exists x\neg F$	($\neg$-elim,10)
12.	$\neg\forall xF\rightarrow\exists x\neg F$	($\rightarrow$-intro,1,11)
13.	$\mid\ \exists x\neg F$	(hypothese 4)
14.	$\mid\ \mid\ a:\ \neg F(a)$	(hypothese 5)
15.	$\mid\ \mid\ \mid\ \forall xF$	(hypothese 6)
16.	$\mid\ \mid\ \mid\ F(a)$	($\forall$-elim,15)
17.	$\mid\ \mid\ \mid\ \neg F(a)$	(rei,14)
18.	$\mid\ \mid\ \neg\forall xF$	($\neg$-intro,15,16,17)
19.	$\mid\ \neg\forall xF$	($\exists$-elim,13,14,18)
20.	$\exists x\neg F\rightarrow\neg\forall xF$	($\rightarrow$-intro,13,19)
21.	$\neg\forall xF\leftrightarrow\exists x\neg F$	($\leftrightarrow$-intro,12,20)

Figuur 12.6: $\vdash \neg\forall xF\leftrightarrow\exists x\neg F$.

1.	$\vert\ \neg\exists x F$	(hypothese 1)
2.	$\vert\ \vert\ F(a)$	(hypothese 2)
3.	$\vert\ \vert\ \neg\exists x F$	(rei,1)
4.	$\vert\ \vert\ \exists x F$	($\exists$-intro,2)
5.	$\vert\ \neg F(a)$	($\neg$-intro,2,3,4)
6.	$\vert\ \forall x \neg F$	($\forall$-intro,5)
7.	$\neg\exists x F \rightarrow \forall x \neg F$	($\rightarrow$-intro,1,6)
8.	$\vert\ \forall x \neg F$	(hypothese 3)
9.	$\vert\ \vert\ \exists x F$	(hypothese 4)
10.	$\vert\ \vert\ \vert\ a : F(a)$	(hypothese 5)
11.	$\vert\ \vert\ \vert\ \vert\ \exists x F$	(hypothese 6)
12.	$\vert\ \vert\ \vert\ \vert\ F(a)$	(rei,10)
13.	$\vert\ \vert\ \vert\ \vert\ \forall x \neg F$	(rei,8)
14.	$\vert\ \vert\ \vert\ \vert\ \neg F(a)$	($\forall$-elim,13)
15.	$\vert\ \vert\ \vert\ \neg\exists x F$	($\neg$-intro,11,12,14)
16.	$\vert\ \vert\ \neg\exists x F$	($\exists$-elim,9,10,15)
17.	$\vert\ \neg\exists x F$	($\neg$-intro,9,16)
18.	$\forall x \neg F \rightarrow \neg\exists x F$	($\rightarrow$-intro,8,17)
19.	$\neg\exists x F \leftrightarrow \forall x \neg F$	($\leftrightarrow$-intro,7,18)

Figuur 12.7: $\vdash \neg\exists x F \leftrightarrow \forall x \neg F$.

1.	$\forall x(P(x) \rightarrow D(x))$	(hypothese 1)
2.	$\exists y[P(y) \wedge S(a,y)]$	(hypothese 2)
3.	$b:\ P(b) \wedge S(a,b)$	(hypothese 3)
4.	$P(b)$	($\wedge$-elim,3)
5.	$S(a,b)$	($\wedge$-elim,3)
6.	$\forall x(P(x) \rightarrow D(x))$	(rei,1)
7.	$P(b) \rightarrow D(b)$	($\forall$-elim,6)
8.	$D(b)$	($\rightarrow$-elim,4,7)
9.	$D(b) \wedge S(a,b)$	($\wedge$-intro,5,8)
10.	$\exists y[D(y) \wedge S(a,y)]$	($\exists$-intro,9)
11.	$\exists y[D(y) \wedge S(a,y)]$	($\exists$-elim,2,3,10)
12.	$\exists y[P(y) \wedge S(a,y)] \rightarrow \exists y[D(y) \wedge S(a,y)]$	($\rightarrow$-intro,2,11)
13.	$\forall x\{\exists y[P(y) \wedge S(x,y)] \rightarrow \exists y[D(y) \wedge S(x,y)]\}$	($\forall$-intro,12)

Figuur 12.8: Afleiding behorende bij voorbeeld 12.2.2.

12.2.2 VOORBEELD In figuur 12.8 vindt men een afleiding van $A \vdash B$, waarbij:

- $A = \forall x(P(x) \rightarrow D(x))$.
- $B = \forall x\{\exists y[P(y) \wedge S(x,y)] \rightarrow \exists y[D(y) \wedge S(x,y)]\}$.

Zie ook voorbeeld 8.5.2(8). ∎

12.2.3 STELLING *De identiteit is reflexief, symmetrisch en transitief, ofwel:*

1. $\vdash \forall x(x = x)$.
2. $\vdash \forall x \forall y[(x = y) \rightarrow (y = x)]$.
3. $\vdash \forall x \forall y \forall z\{[(x = y) \wedge (y = z)] \rightarrow (x = z)\}$.

BEWIJS De afleidingen voor deze stelling vindt men respectievelijk in de figuren 12.9, 12.10 en 12.11. Merk op dat deze stelling alleen van toepassing is op de systemen $\mathcal{PL}^{=}$ en $\mathcal{PL}^{f=}$. ∎

1.	$a = a$	(=-intro)
2.	$\forall x(x = x)$	($\forall$-intro,1)

Figuur 12.9: Reflexiviteit van de identiteit.

1.	$\mid \underline{\ a = b\ }$	(hypothese 1)
2.	$\mid\ a = a$	(=-intro)
3.	$\mid\ b = a$	(sub,1,2)
4.	$(a = b) \rightarrow (b = a)$	($\rightarrow$-intro,1,3)
5.	$\forall y[(a = y) \rightarrow (y = a)]$	($\forall$-intro,4)
6.	$\forall x \forall y[(x = y) \rightarrow (y = x)]$	($\forall$-intro,5)

Figuur 12.10: Symmetrie van de identiteit.

1.	$\mid \underline{\ (a = b) \wedge (b = c)\ }$	(hypothese 1)
2.	$\mid\ a = b$	($\wedge$-elim,1)
3.	$\mid\ b = c$	($\wedge$-elim,1)
4.	$\mid\ a = c$	(sub,2,3)
5.	$[(a = b) \wedge (b = c)] \rightarrow (a = c)$	($\rightarrow$-intro,1,4)
6.	$\forall z\{[(a = b) \wedge (b = z)] \rightarrow (a = z)\}$	($\forall$-intro,5)
7.	$\forall y \forall z\{[(a = y) \wedge (y = z)] \rightarrow (a = z)\}$	($\forall$-intro,6)
8.	$\forall x \forall y \forall z\{[(x = y) \wedge (y = z)] \rightarrow (x = z]\}$	($\forall$-intro,7)

Figuur 12.11: Transitiviteit van de identiteit.

1.	$\forall x[P(x)\rightarrow\neg Q(x)]$	(hypothese 1)
2.	$y = z$	(hypothese 2)
3.	$R(y)$	(hypothese 3)
4.	$\forall x(R(x)\rightarrow Q(x))$	(hypothese 4)
5.	$P(y)$	(hypothese 5)
6.	$\forall x[P(x)\rightarrow\neg Q(x)]$	(rei,1)
7.	$P(y)\rightarrow\neg Q(y)$	($\forall$-elim,6)
8.	$\forall x(R(x)\rightarrow Q(x))$	(rei,4)
9.	$R(y)\rightarrow Q(y)$	($\forall$-elim,8)
10.	$R(y)$	(rei,3)
11.	$Q(y)$	($\rightarrow$-elim,9,10)
12.	$\neg Q(y)$	($\rightarrow$-elim,5,7)
13.	$\neg P(y)$	($\neg$-intro,5,11,12)
14.	$y = z$	(rei,2)
15.	$\neg P(z)$	(sub,13,14)
16.	$\forall x(R(x)\rightarrow Q(x))\rightarrow\neg P(z)$	($\rightarrow$-intro,4,15)
17.	$R(y)\rightarrow[\forall x(R(x)\rightarrow Q(x))\rightarrow\neg P(z)]$	($\rightarrow$-intro,3,16)

Figuur 12.12: Afleiding behorende bij voorbeeld 12.2.4.

12.2.4 VOORBEELD In figuur 12.12 wordt een afleiding gegeven van $A, B \vdash C$ waarbij:

- $A = \forall x[P(x)\rightarrow\neg Q(x)]$.
- $B = (y = z)$.
- $C = R(y)\rightarrow[\forall x(R(x)\rightarrow Q(x))\rightarrow\neg P(z)]$.

Merk op dat de formules B en C vrije variabelen bevatten, namelijk y en z. Vergelijk deze formules met die in voorbeeld 11.2.1(3) waarin geen vrije variabelen voorkomen, maar respectievelijk de namen a en b. ∎

12.2.5 VOORBEELD Een afleiding van $\vdash \forall x R(x, f(x))\rightarrow\forall x\exists y R(x,y)$ wordt gegeven in figuur 12.13. Het is een afleiding in het systeem $\mathcal{PL}^f$ of $\mathcal{PL}^{f=}$. ∎

De syntactische equivalenten van de corollaria 10.2.3 en 10.2.6 worden gegeven in de volgende stelling.

1.	$\forall x R(x, f(x))$	(hypothese 1)
2.	$R(a, f(a))$	($\forall$-elim,1)
3.	$\exists y R(a, y)$	($\exists$-intro,2)
4.	$\forall x \exists y R(x, y)$	($\forall$-intro,3)
5.	$\forall x R(x, f(x)) \rightarrow \forall x \exists y R(x, y)$	($\rightarrow$-intro,1,4)

Figuur 12.13: $\vdash \forall x R(x, f(x)) \rightarrow \forall x \exists y R(x, y)$.

1.	$s = t$	(hypothese 1)
2.	$f[x:=s] = f[x:=s]$	(=-intro)
3.	$f[x:=s] = f[x:=t]$	(sub,1,2)

Figuur 12.14: $s = t \vdash f[x:=s] = f[x:=t]$.

12.2.6 STELLING *Zij $f, s, t \in TERM$ en $A \in FORM$.*

1. $s = t \vdash f[x:=s] = f[x:=t]$,
2. $s = t \vdash A[x:=s] \leftrightarrow A[x:=t]$.

BEWIJS Zie de afleidingen in de figuren 12.14 en 12.15. Let op het gebruik van de sub-regel: niet *alle* voorkomens van s zijn door t vervangen in de formule $f[x:=s] = f[x:=s]$, respectievelijk $A[x:=s] \leftrightarrow A[x:=s]$. ∎

12.3 Het uitgebreide systeem van Fitch

Evenals in paragraaf 6.3 kunnen we het systeem van Fitch $\mathcal{F}$ uitbreiden tot een *uitgebreid systeem van Fitch* $\mathcal{F}_{uit}$. De afleidingsregels van dit uitgebreide systeem zijn de gewone afleidingsregels plus de regels uit paragraaf 6.3 en de twee nieuwe regels uit tabel 12.3. Voor alle nieuwe regels geldt dat $A, B \in FORM$. De regels uit tabel 12.3 worden gerechtvaardigd door stelling 12.2.1. Het is eenvoudig om te bewijzen dat de systemen $\mathcal{F}$ en $\mathcal{F}_{uit}$ equivalent zijn.

12.3.1 STELLING **Equivalentie van $\mathcal{F}$ en $\mathcal{F}_{uit}$**
Zij $\Gamma \subseteq FORM$ en $F \in FORM$. Dan geldt:

$$\Gamma \vdash_{\mathcal{F}} F \quad \Leftrightarrow \quad \Gamma \vdash_{\mathcal{F}_{uit}} F.$$

1.	$s = t$	(hypothese 1)
2.	$\quad \mid A[x{:=}s]$	(hypothese 2)
3.	$\quad \mid A[x{:=}s]$	(rei,2)
4.	$A[x{:=}s] \rightarrow A[x{:=}s]$	($\rightarrow$-intro,2,3)
5.	$A[x{:=}s] \leftrightarrow A[x{:=}s]$	($\leftrightarrow$-intro,4,4)
6.	$A[x{:=}s] \leftrightarrow A[x{:=}t]$	(sub,1,5)

Figuur 12.15: $s = t \vdash A[x{:=}s] \leftrightarrow A[x{:=}t]$.

$\forall$-regel	**$\exists$-regel**
$\neg\forall x A$	$\neg\exists x A$
$\vdots$	$\vdots$
$\boldsymbol{\exists x \neg A}$	$\boldsymbol{\forall x \neg A}$

Tabel 12.3: Regels voor de kwantoren in het systeem $\mathcal{F}_{uit}$.

Tot slot van deze paragraaf geven we een voorbeeld van een afleiding in het uitgebreide systeem van Fitch $\mathcal{F}_{uit}$.

12.3.2 Voorbeeld Afleiding in het systeem $\mathcal{F}_{uit}$.
De bewijsfiguur in figuur 12.16 is een afleiding van $\vdash \forall x P(x)) \rightarrow \exists x P(x)$. Dat afleidingen in $\mathcal{F}_{uit}$ niet noodzakelijk eenvoudiger zijn dan afleidingen in $\mathcal{F}$, maakt de bewijsfiguur in figuur 12.17 duidelijk. ∎

12.4 Opgaven

1. Zij A en B éénplaatsige predicaatsymbolen. Bewijs met behulp van het uitgebreide systeem van Fitch $\mathcal{F}_{uit}$ de volgende metabeweringen:

   (a) $\forall x[A(x) \rightarrow B(x)] \vdash \forall x A(x) \rightarrow \forall x B(x)$.

   (b) $\exists x[A(x) \rightarrow B(x)] \vdash \forall x A(x) \rightarrow \exists x B(x)$.

   (c) $\forall x A(x) \rightarrow \forall x B(x) \vdash \exists x[A(x) \rightarrow B(x)]$.

   (d) $\vdash \exists x[A(x) \rightarrow \forall y A(y)]$.

   (e) $\exists x A(x) \rightarrow \exists x B(x) \vdash \exists x[A(x) \rightarrow B(x)]$.

$$\begin{array}{r|ll}
1. & \neg[\forall xP(x)\rightarrow\exists xP(x)] & \text{(hypothese 1)} \\
\cline{2-2}
2. & \forall xP(x) \wedge \neg\exists xP(x) & (\rightarrow\text{-regel1,1}) \\
3. & \forall xP(x) & (\wedge\text{-elim,2}) \\
4. & \neg\exists xP(x) & (\wedge\text{-elim,2}) \\
5. & \forall x\neg P(x) & (\exists\text{-regel,4}) \\
6. & P(a) & (\forall\text{-elim,3}) \\
7. & \neg P(a) & (\forall\text{-elim,5}) \\
8. & \multicolumn{1}{l}{\neg\neg[\forall xP(x)\rightarrow\exists xP(x)]} & (\neg\text{-intro,1,6,7}) \\
9. & \multicolumn{1}{l}{\forall xP(x)\rightarrow\exists xP(x)} & (\neg\text{-elim,8})
\end{array}$$

Figuur 12.16: $\vdash \forall xP(x)\rightarrow\exists xP(x)$ afgeleid in $\mathcal{F}_{uit}$.

$$\begin{array}{r|ll}
1. & \forall xP(x) & \text{(hypothese 1)} \\
\cline{2-2}
2. & P(a) & (\forall\text{-elim,1}) \\
3. & \exists xP(x) & (\exists\text{-intro,2}) \\
4. & \multicolumn{1}{l}{\forall xP(x)\rightarrow\exists xP(x)} & (\rightarrow\text{-intro,1,3})
\end{array}$$

Figuur 12.17: $\vdash \forall xP(x)\rightarrow\exists xP(x)$ afgeleid in $\mathcal{F}$.

2. $R(x,y)$ representeert een *functie* als $R(x,y)$ voldoet aan:

$$\forall x\exists y[R(x,y) \wedge \forall z[R(x,z)\rightarrow y = z]]. \tag{12.1}$$

$R(x,y)$ representeert een *surjectieve* functie als $R(x,y)$ voldoet aan formule (12.1) en aan:

$$\forall y\exists xR(x,y). \tag{12.2}$$

$R(x,y)$ representeert een *injectieve* functie als $R(x,y)$ voldoet aan formule (12.1) en aan:

$$\forall x\forall y\forall z[(R(x,z) \wedge R(y,z))\rightarrow x = y]. \tag{12.3}$$

$R(x,y)$ representeert een *bijectieve* functie als $R(x,y)$ voldoet aan de formules (12.1), (12.2) en (12.3), óf $R(x,y)$ voldoet aan formule (12.1) en aan:

$$\forall y\exists x[R(x,y) \wedge \forall z[R(z,y)\rightarrow x = z]]. \tag{12.4}$$

In de volgende drie stappen bewijst u dat de voorwaarden (12.2) en (12.3) equivalent zijn met voorwaarde (12.4).

(a) Bewijs in $\mathcal{F}_{uit}$: $(12.2), (12.3) \vdash (12.4)$.

(b) Bewijs in $\mathcal{F}_{uit}$: $(12.4) \vdash (12.2)$.

(c) Bewijs in $\mathcal{F}_{uit}$: $(12.4) \vdash (12.3)$.

3. Zij R een tweeplaatsig predicaatsymbool in de volgende zinnen:

   F1. $\forall x \forall y[R(x,y) \rightarrow \exists z[R(x,z) \wedge R(z,y)]]$.

   F2. $\neg \exists x R(x,x)$.

   F3. $\forall x \forall y[R(x,y) \rightarrow \exists z[\neg(z = x) \wedge \neg(z = y)]]$.

   Bewijs nu in het uitgebreide systeem van Fitch $\mathcal{F}_{uit}$ dat: F1, F2 $\vdash$ F3. (Lees voor R de relatie $<$ op $\mathbb{R}$.)

4. Zij N een éénplaatsig en S een tweeplaatsig predicaatsymbool in de volgende zinnen:

   F1. $\forall x \forall y[(N(x) \wedge S(x,y)) \rightarrow N(y)]$.

   F2. $\forall x \forall y[S(x,y) \rightarrow \neg(x = y)]$.

   F3. $\forall x[N(x) \rightarrow \exists y S(x,y)]$.

   F4. $\forall x[N(x) \rightarrow \exists y[N(y) \wedge \neg(x = y)]]$.

   Bewijs in het uitgebreide systeem van Fitch $\mathcal{F}_{uit}$ dat: F1, F2, F3 $\vdash$ F4. (Lees voor $N(x)$ 'x is een natuurlijk getal' en voor $S(x,y)$ '$y = x + 1$'.)

5. Zij E een éénplaatsig en S een drieplaatsig predicaatsymbool in de volgende zinnen:

   F1. $\forall x \forall y[(E(x) \wedge E(y)) \rightarrow \exists z[E(z) \wedge S(z,x,y)]]$.

   F2. $\forall x \forall y[(\neg E(x) \wedge \neg E(y)) \rightarrow \exists z[E(z) \wedge S(z,x,y)]]$.

   F3. $\forall x \forall y \forall v \forall w[(S(v,x,y) \wedge S(w,x,y)) \rightarrow v = w]$.

   F4. $\forall x \exists y[E(y) \wedge \forall z[S(z,x,x) \rightarrow y = z]]$.

   Bewijs in het uitgebreide systeem van Fitch $\mathcal{F}_{uit}$ dat: F1, F2, F3 $\vdash$ F4. (Hint: lees voor $E(x)$ 'x is een even getal' en voor $S(z,x,y)$ 'z is de som van x en y'; ga vervolgens na wat de verschillende zinnen uitdrukken en probeer dan eerst uit F1 en F2 voor willekeurige a de zin $\exists z[E(z) \wedge S(z,a,a)]$ af te leiden.)

Hoofdstuk 13

Correctheid en Volledigheid van de Predicatenlogica

Evenals in de propositielogica kan men voor de predicatenlogica aantonen dat 'afleidbaarheid' en 'logisch gevolg' samenvallen, of anders gezegd dat $\Gamma \vdash F$ equivalent is met $\Gamma \models F$. Deze equivalentie wordt tot uitdrukking gebracht door de correctheidsstelling en de volledigheidsstelling. De wijze waarop beide stellingen kunnen worden bewezen is analoog aan die voor de corresponderende stellingen in de propositielogica.

Dit hoofdstuk is als volgt georganiseerd. In §13.1 behandelen we de correctheidsstelling, in §13.2 de compactheidsstelling, en in §13.3, tenslotte, de volledigheidsstelling voor de predicatenlogica.

13.1 Correctheid

De volgende stelling is nodig om de correctheid van de predicatenlogica te bewijzen. De stelling brengt tot uitdrukking dat de regels uit tabel 12.1 *waarheidsbehoudend* zijn. Dit betekent dat de afleidingsregels semantisch gezien correct zijn. Men kan er dus geen onware uitspraken mee afleiden uit ware uitspraken. De beweringen uit de stelling corresponderen achtereenvolgens met de regels $\forall$-intro, $\forall$-elim, $\exists$-intro, $\exists$-elim, $=$-intro en sub.

13.1.1 Stelling *Zij* $\Gamma \subseteq FORM$, $A, B, F \in FORM$, $s, t \in TERM$, *en zij* c *een naam of variabele die niet voorkomt, respectievelijk niet vrij voorkomt, in* Γ, A *en* B. *Dan geldt:*

1. $\Gamma \models A[x{:=}c] \quad \Rightarrow \quad \Gamma \models \forall x A$,
2. $\Gamma \models \forall x F \quad \Rightarrow \quad \Gamma \models F[x{:=}t]$,
3. $\Gamma \models F[x{:=}t] \quad \Rightarrow \quad \Gamma \models \exists x F$,

4. $\Gamma \models \exists x A \quad \& \quad \Gamma \cup \{A[x:=c]\} \models B \quad \Rightarrow \quad \Gamma \models B,$

5. $\Gamma \models t = t,$

6. $\Gamma \models s = t \quad \& \quad \Gamma \models F[x:=s] \quad \Rightarrow \quad \Gamma \models F[x:=t].$

BEWIJS *We bewijzen alleen de uitspraken 1 en 4. In het onderstaande nemen we aan dat c een naam is. De bewijzen ingeval c een variabele is, worden aan de lezer overgelaten.*

1. *Bewering 1 (*$\forall$*-intro).*

   Stel dat $\langle \mathcal{A}, \beta \rangle$ *een model is voor* Γ*, we moeten dan bewijzen dat* $\langle \mathcal{A}, \beta \rangle$ *tevens een model is voor* $\forall x A$.

   Zij $\mathcal{A}'$ *een structuur die hooguit verschilt van* $\mathcal{A}$ *wat betreft de denotatie van c. Aangezien c niet voorkomt in* Γ*, geldt wegens stelling 9.4.2 voor alle dergelijke* $\mathcal{A}'$ *dat* $\langle \mathcal{A}', \beta \rangle \models \Gamma$.

   Uit het gegeven $\Gamma \models A[x:=c]$ *volgt nu dat eveneens voor alle dergelijke* $\mathcal{A}'$ *geldt* $\langle \mathcal{A}', \beta \rangle \models A[x:=c]$*. Uit stelling 10.2.5 volgt dan voor al deze structuren* $\mathcal{A}'$ *dat* $\langle \mathcal{A}', \beta[x \mapsto c^{\mathcal{A}', \beta}] \rangle \models A$*. Aangezien c ook niet voorkomt in A, geldt wederom wegens stelling 9.4.2 dat:*

   $$\forall \mathcal{A}' \cdot \langle \mathcal{A}, \beta[x \mapsto c^{\mathcal{A}', \beta}] \rangle \models A. \tag{13.1}$$

   Nu neemt $c^{\mathcal{A}', \beta}$ *voor verschillende* $\mathcal{A}'$ *alle mogelijke waarden in* $|\mathcal{A}|$ *aan, zodat (13.1) equivalent is met:*

   $$\forall a \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[x \mapsto a] \rangle \models A,$$

   zodat $\langle \mathcal{A}, \beta \rangle \models \forall x A$*, hetgeen te bewijzen was.*

2. *Bewering 4 (*$\exists$*-elim).*

   Stel dat $\langle \mathcal{A}, \beta \rangle$ *een model is voor* Γ*, we moeten dan bewijzen dat* $\langle \mathcal{A}, \beta \rangle$ *tevens een model is voor* B.

   Uit het gegeven dat $\Gamma \models \exists x A$*, volgt dat* $\langle \mathcal{A}, \beta \rangle \models \exists x A$*, zodat voor zekere* $a \in |\mathcal{A}|$ *geldt* $\langle \mathcal{A}, \beta[x \mapsto a] \rangle \models A$*. Zij nu* $\mathcal{A}'$ *een structuur die gelijk is aan* $\mathcal{A}$*, behalve dat* $c^{\mathcal{A}'} = a$*. Aangezien c niet voorkomt in* Γ *en A, geldt dan wegens stelling 9.4.2 dat:*

   $$\langle \mathcal{A}', \beta \rangle \quad \models \quad \Gamma, \tag{13.2}$$

   $$\langle \mathcal{A}', \beta[x \mapsto c^{\mathcal{A}', \beta}] \rangle \quad \models \quad A. \tag{13.3}$$

   Passen we nu stelling 10.2.5 toe op (13.3), dan volgt:

   $$\langle \mathcal{A}', \beta \rangle \quad \models \quad A[x:=c]. \tag{13.4}$$

   In combinatie met het gegeven dat $\Gamma \cup \{A[x:=c]\} \models B$ *volgt uit (13.2) en (13.4) dat* $\langle \mathcal{A}', \beta \rangle \models B$*. Aangezien c niet voorkomt in B, kunnen we nogmaals stelling 9.4.2 toepassen, hetgeen* $\langle \mathcal{A}, \beta \rangle \models B$ *als resultaat heeft. Dit moest worden aangetoond.* ∎

Met behulp van de bovenstaande stelling en van stelling 7.1.3 die ook voor de predicatenlogica geldt (lees overal '*FORM*' waar '*PROP*' staat), kan de correctheidsstelling worden bewezen.

13.1.2 STELLING **Correctheidsstelling**

Zij $\Gamma \subseteq FORM$ *en* $F \in FORM$*, dan geldt:*

$$\Gamma \vdash F \quad \Rightarrow \quad \Gamma \models F.$$

BEWIJS *Het bewijs van deze stelling is vrijwel identiek aan het bewijs van de correctheidsstelling voor de propositielogica. Nu dient echter niet alleen de versie van stelling 7.1.3 voor de predicatenlogica, maar ook stelling 13.1.1 te worden gebruikt.* ∎

13.2 Compactheid

De volledigheidsstelling voor de predicatenlogica kan in principe op dezelfde wijze worden bewezen als die voor de propositielogica. Ook dan hebben we de compactheidsstelling nodig. Deze volgt echter bijna onmiddellijk uit een toepassing van de boomstelling (11.3.4).

Ook voor de predicatenlogica kan men het begrip eindige vervulbaarheid definiëren.

13.2.1 DEFINITIE **Eindige vervulbaarheid**
Een verzameling $\Gamma \subseteq FORM$ *noemt men* eindig vervulbaar *indien alle eindige deelverzamelingen van* Γ *vervulbaar zijn.*

13.2.2 STELLING **Compactheidsstelling**
Zij $\Gamma \subseteq ZIN$ *voor een gegeven eerste-ordetaal* L*, dan is* Γ *vervulbaar dan en slechts dan als* Γ *eindig vervulbaar is.*

BEWIJS ($\Rightarrow$)*: Triviaal.*

($\Leftarrow$)*: Stel* Γ *is niet vervulbaar. We moeten dan laten zien dat* Γ *niet eindig vervulbaar is. Met andere woorden, we moeten aantonen dat er een eindige deelverzameling van* Γ *bestaat die niet vervulbaar is.*

Uit de aanname dat Γ *niet vervulbaar is en de boomstelling (11.3.4) volgt dat er een boom* $B(\Gamma)$ *bestaat die sluit. Wegens König's Lemma (11.3.3) is deze boom eindig. Dat betekent dat* $B(\Gamma)$ *slechts een eindig aantal van de formules in* Γ *bevat. Zij* Γ_0 *de verzameling van alle formules uit* Γ *die in* $B(\Gamma)$ *voorkomen. Evident geldt dat* $B(\Gamma)$ *ook een sluitende boom is voor* Γ_0 *(zie ook definitie 11.3.1). Dit betekent dat* Γ_0 *niet vervulbaar is.* ∎

Evenals voor de propositielogica geldt voor de predicatenlogica het volgende corollarium. We vermelden het (uiteraard) zonder bewijs.

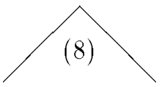
13.2.3 COROLLARIUM *Zij* $\Gamma \subseteq FORM$ *en* $F \in FORM$*, dan geldt dat* $\Gamma \models F$ *dan en slechts dan als er een eindige deelverzameling* $\Gamma_0 \subseteq \Gamma$ *bestaat zodanig dat* $\Gamma_0 \models F$*.*

13.3 Volledigheid

De volledigheidsstelling werd voor het eerst in 1930 bewezen door de geniale logicus Kurt Gödel. In deze paragraaf zullen wij een bewijs voor deze stelling geven dat men zelden in boeken tegenkomt. Meestal bewijst men namelijk eerst de zogenaamde *consistentiestelling.* Uit deze stelling volgen dan de volledigheids- en compactheidsstelling als corollaria.

De weg die wij hebben gekozen, gaat uit van de compactheidsstelling en van het idee dat sluitende bomen kunnen worden omgezet in afleidingen in

1. $\exists x[P(x) \vee Q(x)]$	(hypothese)
2. $\neg[\exists x P(x) \vee \exists x Q(x)]$	($\neg$ conclusie)
3. $\neg\exists x P(x) \wedge \neg\exists x Q(x)$	($\vee$-regel,2)
4. $\neg\exists x P(x)$	($\wedge$-regel,3)
5. $\neg\exists x Q(x)$	($\wedge$-regel,3)
6. $\forall x \neg P(x)$	($\exists$-regel,4)
7. $\forall x \neg Q(x)$	($\exists$-regel,5)
8. $P(a) \vee Q(a)$	($\exists$-regel,1)



9. $P(a)$		11. $Q(a)$	
10. $\neg P(a)$	($\forall$-regel,6)	12. $\neg Q(a)$	($\forall$-regel,7)
X(9,10)		**X**(11,12)	

Figuur 13.1: $\exists x[P(x) \vee Q(x)] \models \exists x P(x) \vee \exists x Q(x)$.

het uitgebreide systeem van Fitch. Deze benadering heeft als voordeel dat een direct verband wordt gelegd tussen de boommethode en natuurlijke deductie.

We zullen nu aan de hand van een voorbeeld demonstreren hoe een sluitende boom voor $\Gamma \models F$ waarbij Γ eindig is, kan worden getransformeerd naar een afleiding in het uitgebreide systeem van Fitch. In figuur 13.1 is een boom weergeven voor de bewering:

$$\exists x[P(x) \vee Q(x)] \models \exists x P(x) \vee \exists x Q(x).$$

In figuur 13.2 staat de ermee corresponderende afleiding in het uitgebreide systeem van Fitch. De methode voor de omzetting van een sluitende boom naar een afleiding in het uitgebreide systeem van Fitch is gebaseerd op de methode voor de propositielogica en is als volgt.

1. De afleiding begint met een hypothese-interval met als hypothesen de formules uit Γ. In de boom in figuur 13.1 bevat Γ slechts één hypothese: $\exists x[P(x) \vee Q(x)]$.

1.	$\mid\ \exists x[P(x) \vee Q(x)]$	(hypothese)
2.	$\mid\ \mid\ \neg[\exists x P(x) \vee \exists x Q(x)]$	($\neg$ conclusie)
3.	$\mid\ \mid\ \neg\exists x P(x) \wedge \neg\exists x Q(x)$	($\vee$-regel,2)
4.	$\mid\ \mid\ \neg\exists x P(x)$	($\wedge$-elim,3)
5.	$\mid\ \mid\ \neg\exists x Q(x)$	($\wedge$-elim,3)
6.	$\mid\ \mid\ \forall x \neg P(x)$	($\exists$-regel,4)
7.	$\mid\ \mid\ \forall x \neg Q(x)$	($\exists$-regel,5)
8.	$\mid\ \mid\ \mid\ a : P(a) \vee Q(a)$	($\exists$-hypothese)
9.	$\mid\ \mid\ \mid\ \mid\ P(a)$	(linker vertakking van 8)
10.	$\mid\ \mid\ \mid\ \mid\ \neg P(a)$	($\forall$-elim,6)
10A.	$\mid\ \mid\ \mid\ \mid\ X$	(contra,9,10)
11.	$\mid\ \mid\ \mid\ \mid\ Q(a)$	(rechter vertakking van 8)
12.	$\mid\ \mid\ \mid\ \mid\ \neg Q(a)$	($\forall$-elim,7)
12A.	$\mid\ \mid\ \mid\ \mid\ X$	(contra,11,12)
13.	$\mid\ \mid\ \mid\ X$	($\vee$-elim,8,10A,12A)
14.	$\mid\ \mid\ X = \exists x P(x) \vee \exists x Q(x)$	($\exists$-elim,1,8,13)
15.	$\mid\ \neg\neg[\exists x P(x) \vee \exists x Q(x)]$	($\neg$-intro,2,14)
16.	$\mid\ \exists x P(x) \vee \exists x Q(x)$	($\neg$-elim,15)

Figuur 13.2: $\exists x[P(x) \vee Q(x)] \vdash \exists x P(x) \vee \exists x Q(x)$.

2. Het volgende hypothese-interval heeft als hypothese de ontkenning van de conclusie, in dit geval $\neg[\exists x P(x) \vee \exists x Q(x)]$.

3. De afleiding volgt steeds letterlijk de boom totdat deze zich vertakt, of totdat een formule wordt tegengekomen die het resultaat is van een toepassing van de tweede $\exists$-regel (waarbij een nieuwe naam, of variabele c wordt geïntroduceerd).

4. Voor iedere formule die ontstaan is door toepassing van de tweede $\exists$-regel, wordt een nieuw hypothese-interval geopend met die formule als hypothese. Dit interval wordt later gebruikt voor de regel $\exists$-elim (zie stap 7). In de boom onder beschouwing is de formule op regel 8 door een toepassing van de tweede $\exists$-regel ontstaan waarbij a als nieuwe naam is geïntroduceerd.

5. Voor iedere eerste formule van een vertakking wordt een nieuw hypothese-interval geopend met die formule als hypothese.

6. Als een tak van de boom afsluit, wordt de contra-regel toegepast op de formules met het zelfde nummer als in de boom, met als resultaat een willekeurige formule X die later geïnstantieerd zal worden met de conclusie, in dit geval de formule $\exists x P(x) \vee \exists x Q(x)$.

7. Als alle formules in de boom aan de beurt zijn geweest, kan de afleiding compleet worden gemaakt door een aantal toepassingen van de regels $\exists$-elim en/of $\vee$-elim, en tenslotte door achtereenvolgens de regels $\neg$-intro en $\neg$-elim toe te passen.

13.3.1 STELLING **Volledigheidsstelling**
Zij $\Gamma \subseteq FORM$ *en* $F \in FORM$*, dan geldt:*

$$\Gamma \models F \quad \Rightarrow \quad \Gamma \vdash F.$$

BEWIJS Het bewijs van deze stelling is geheel analoog aan het bewijs van de volledigheid van de propositielogica. Het loopt als volgt. Uit $\Gamma \models F$ volgt wegens corollarium 13.2.3 dat er een eindige deelverzameling Γ_0 van Γ bestaat zodanig dat $\Gamma_0 \models F$. Hieruit volgt met de boomstelling dat er een sluitende boom bestaat voor de verzameling $\Gamma_0 \cup \{\neg F\}$. Deze sluitende boom kan volgens de bovenstaande methode worden omgezet naar een afleiding van $\Gamma_0 \vdash F$ in het uitgebreide systeem van Fitch. Dit impliceert tenslotte dat $\Gamma \vdash F$. Uiteraard moet nog wel worden bewezen dat de omzettingsmethode correct is. Het bewijs hiervan verloopt analoog aan dat voor de propositielogica. ■

Evenals in de propositielogica, is er een *consistentiestelling* voor de predicatenlogica. We definiëren eerst het begrip consistentie.

13.3.2 DEFINITIE **Consistentie**
Een verzameling $\Gamma \subseteq FORM$ *wordt* consistent *genoemd dan en slechts dan als er geen formule* $F \in FORM$ *bestaat zodanig dat geldt* $\Gamma \vdash F$ *en* $\Gamma \vdash \neg F$.

13.3.3 STELLING **Consistentiestelling**
Als $\Gamma \subseteq FORM$ *consistent is, dan is* Γ *vervulbaar.*

BEWIJS Het bewijs is geheel analoog aan dat voor de propositielogica. ■

13.3.4 DEFINITIE **Maximale consistentie**
Een verzameling formules $\Gamma \subseteq FORM$ *noemt men* maximaal consistent, *indien aan de volgende twee voorwaarden is voldaan:*

1. Γ *is consistent.*
2. *Als* $\Gamma \cup \{F\}$ *consistent is, dan* $F \in \Gamma$.

Op analoge wijze definieert men wanneer een verzameling zinnen maximaal consistent is.

Speciale maximaal consistente verzamelingen zijn de zogenaamde *Henkin-theorieën*.

13.3.5 DEFINITIE **Henkin-theorie**
Een verzameling zinnen $\Gamma \subseteq ZIN$ *noemt men een* Henkin-theorie, *indien aan de volgende twee voorwaarden is voldaan:*

1. Γ *is maximaal consistent.*
2. *Als* $\exists x A \in \Gamma$, *dan is er een naam c zodat* $A[x{:=}c] \in \Gamma$.

De rol van het begrip Henkin-theorie in de predicatenlogica is dat het wordt gebruikt in het directe bewijs van de consistentiestelling (beperkt tot verzamelingen van zinnen). Dit bewijs bestaat uit twee stappen:

1. Men laat eerst zien dat iedere consistente verzameling zinnen $\Gamma \subseteq ZIN$ kan worden uitgebreid tot een Henkin-theorie $T \supseteq \Gamma$.
2. Tenslotte bewijst men dat iedere Henkin-theorie vervulbaar is.

De lezer die geïnteresseerd is in de verdere details, verwijzen we naar de literatuur.

13.4 Opgaven

1. Zij A en B éénplaatsige predicaatsymbolen. Laat met behulp van de boommethode zien dat de volgende metabeweringen geldig zijn en transformeer daarna de verkregen boom tot een afleiding in het uitgebreide systeem van Fitch $\mathcal{F}_{uit}$:
   (a) $\forall x A(x) \rightarrow \forall x B(x) \models \exists x[A(x) \rightarrow B(x)]$.
   (b) $\forall x[A(x) \wedge B(x)] \models \forall x A(x) \wedge \forall x B(x)$.
2. Maak het bewijs van stelling 13.1.1 af.
3. Vul het bewijs van de correctheidsstelling voor de propositielogica aan tot een bewijs ervan voor de predicatenlogica.

Hoofdstuk 14

Theorieën en Peano-aritmetiek

In dit hoofdstuk introduceren we het begrip *theorie.*Theorieën zijn verzamelingen volzinnen die bepaalde stukken wetenschap representeren, zoals de theorie van Newton of de theorie van de natuurlijke getallen. Deze theorieën worden doorgaans gegeven in de vorm van een aantal axioma's. In paragraaf 14.1 behandelen we het begrip theorie en in paragraaf 14.2 geven als voorbeeld de *Peano-aritmetiek*, ofwel de theorie der natuurlijke getallen.

14.1 Theorieën

In de wiskunde en in de fysica is het gebruikelijk om over theorieën te spreken. Zo spreekt men bijvoorbeeld over de theorie van Newton, de theorie van de quantummechanica, of de theorie van de rekenkunde. Onder zo'n theorie verstaat men een samenhangende verzameling van uitspraken of stellingen. Onderzoek heeft dan als doel om nieuwe stellingen van de theorie te vinden. Dit betekent dat stellingen die afleidbaar zijn uit de bekende stellingen, ook tot de theorie behoren. In logische termen uitgedrukt is een *theorie* dus een verzameling T van volzinnen die gesloten is onder de relatie $\vdash$. Met het laatste wordt bedoeld dat alle volzinnen die uit T afleidbaar zijn, element zijn van T. Een ander aspect van een theorie is dat men graag wil dat deze consistent is. Dit is equivalent met de eis dat de theorie vervulbaar is.

14.1.1 DEFINITIE **Theorie**
Een theorie *is een verzameling* $T \subseteq ZIN$ *waarvoor geldt:*

1. *T bezit een model, en*
2. $T = \{A \in ZIN \mid T \vdash A\}$.

Een voorbeeld van een theorie wordt gegeven in de volgende stelling.

14.1.2 STELLING *Als $\mathcal{A}$ een structuur is, dan is de verzameling:*

$$T = \{F \in ZIN \mid \mathcal{A} \models F\}$$

een theorie.

BEWIJS Uiteraard is $\mathcal{A}$ een model voor T. Laat nu gegeven zijn dat $T \vdash F$, dan moeten we aantonen dat $F \in T$, en omgekeerd. Vanwege de correctheidsstelling volgt uit $T \vdash F$, dat $T \models F$. Aangezien $\mathcal{A}$ een model is voor T, moet dan gelden dat $\mathcal{A} \models F$, zodat $F \in T$. Neem tenslotte aan dat $F \in T$. Dan volgt hier onmiddellijk uit dat $T \vdash F$. Hiermee is de stelling bewezen. ■

14.1.3 DEFINITIE **Theorie van een structuur**
Als $\mathcal{A}$ een structuur is, dan noemt men de verzameling $\{F \in ZIN \mid \mathcal{A} \models F\}$, notatie $Th(\mathcal{A})$, de theorie van *$\mathcal{A}$.*

In het bovenstaande hebben we gezien dat een structuur $\mathcal{A}$ een theorie $Th(\mathcal{A})$ bepaalt. Een andere mogelijkheid om een theorie vast te leggen, is door een vervulbare verzameling $\Gamma \subseteq ZIN$ te kiezen. De theorie die door Γ wordt bepaald, bestaat dan uit alle zinnen die uit Γ afleidbaar zijn. De zinnen uit Γ worden *axioma's* genoemd. Soms worden axioma's gegeven in de vorm van *axiomaschema's*. Een axiomaschema legt een verzameling van axioma's schematisch vast. Men verkrijgt een zogenaamde *instantie* van een axiomaschema door voor de metavariabelen in het schema concrete formules in te vullen en van het resultaat de universele afsluiting te nemen om er een zin van te maken (zie definitie 9.4.6).

14.1.4 VOORBEELD Instantie van een axiomaschema.
Zij het volgende axiomaschema gegeven (zie ook voorbeeld 14.1.6):

$$\forall x \forall y[E(x,y) \rightarrow (A[z:=x] \rightarrow A[z:=y])].$$

In dit schema is A een metavariabele waarvoor we formules kunnen invullen. Zij nu $A = E(w,z)$, dan is de volgende zin een instantie van het gegeven axiomaschema:

$$\forall w \forall x \forall y[E(x,y) \rightarrow (E(w,x) \rightarrow E(w,y))].$$

Merk op dat $\forall w$ is toegevoegd om te zorgen dat het resultaat een zin is: nadat in het axiomaschema de metavariabele A is vervangen door de formule $E(w,z)$, is van de resulterende formule de universele afsluiting genomen. ■

Theorieën die gegeven zijn door een eindige verzameling van axioma's en/of axiomaschema's noemt men *axiomatiseerbare theorieën*. Dit zijn de voor ons 'interessante' theorieën omdat deze gebruikt kunnen worden in samenhang met *theoremabewijzers.* Dit zijn computerprogramma's waarmee het mogelijk is om theorema's automatisch of semi-automatisch te bewijzen. Dergelijke programma's worden steeds meer gebruikt als onderdeel van meer complexe programmatuur waarmee programmaspecificaties op hun correctheid kunnen worden beproefd.

14.1.5 DEFINITIE **Axiomatiseerbare theorie**
Een theorie T noemt men axiomatiseerbaar *indien er een eindige verzameling Γ van axioma's en/of axiomaschema's bestaat, zodanig dat:*

$$T = \{F \in \mathit{ZIN} \mid \Gamma \vdash F\}.$$

Met $\Gamma \vdash F$ wordt hier bedoeld dat F afleidbaar is uit de axioma's en instanties van de axiomaschema's uit Γ.

In het volgende voorbeeld wordt een theorie beschreven met behulp van één axioma en één axiomaschema.

14.1.6 VOORBEELD Axiomatiseerbare theorie.
Zij gegeven dat E een tweeplaatsig predicaat is en $A \in \mathit{FORM}$. Definieer nu:

1. $F_1 = \forall x E(x, x)$, en
2. $F_2 = \forall x \forall y[E(x, y) \rightarrow (A[z := x] \rightarrow A[z := y])]$.

Het axioma F_1 en het axiomaschema F_2 bepalen samen de volgende oneindige verzameling van axioma's:

$$\Gamma = \{F_1\} \cup \{A \in \mathit{ZIN} \mid A \text{ is een instantie van } F_2\},$$

$T = \{F \in \mathit{ZIN} \mid \Gamma \vdash F\}$ is dan de theorie die door F_1 en F_2 wordt voortgebracht. We zullen nu bewijzen de volgende twee volzinnen tot T behoren:

1. $A = \forall x \forall y[E(x, y) \rightarrow E(y, x)]$,
2. $B = \forall x \forall y \forall z[E(x, y) \rightarrow (E(y, z) \rightarrow E(x, z))]$.

In de afleidingen in de figuren 14.1 en 14.2 wordt bewezen dat $A, B \in T$. Men noteert dit als $\vdash_T A$, respectievelijk $\vdash_T B$. Men kan zelfs bewijzen dat:

$$\vdash_T \forall x \forall y[E(x, y) \leftrightarrow x = y].$$

Dit resultaat kan men als volgt herformuleren: de axioma's F_1 en F_2 axiomatiseren de identiteit $=$. ■

1.	$\forall x E(x,x)$	(axioma F_1)
2.	$\forall x \forall y [E(x,y) \rightarrow (E(x,x) \rightarrow E(y,x))]$	(instantie F_2)
3.	$E(a,b)$	(hypothese 1)
4.	$E(a,a)$	($\forall$-elim,1)
5.	$\forall y [E(a,y) \rightarrow (E(a,a) \rightarrow E(y,a))]$	($\forall$-elim,2)
6.	$E(a,b) \rightarrow (E(a,a) \rightarrow E(b,a))$	($\forall$-elim,5)
7.	$E(a,a) \rightarrow E(b,a)$	($\rightarrow$-elim,3,6)
8.	$E(b,a)$	($\rightarrow$-elim,4,7)
9.	$E(a,b) \rightarrow E(b,a)$	($\rightarrow$-intro,3,8)
10.	$\forall y [E(a,y) \rightarrow E(y,a)]$	($\forall$-intro,9)
11.	$\forall x \forall y [E(x,y) \rightarrow E(y,x)]$	($\forall$-intro,10)

Figuur 14.1: $\vdash_T \forall x \forall y [E(x,y) \rightarrow E(y,x)]$.

1.	$\forall x E(x,x)$	(axioma F_1)
2.	$\forall w \forall x \forall y [E(x,y) \rightarrow (E(x,w) \rightarrow E(y,w))]$	(instantie F_2)
3.	$E(a,b)$	(hypothese 1)
4.	$E(b,c)$	(hypothese 2)
5.	$\forall x \forall y [E(x,y) \rightarrow (E(x,c) \rightarrow E(y,c))]$	($\forall$-elim,2)
6.	$\forall y [E(b,y) \rightarrow (E(b,c) \rightarrow E(y,c))]$	($\forall$-elim,5)
7.	$E(b,a) \rightarrow (E(b,c) \rightarrow E(a,c))$	($\forall$-elim,6)
8.	$\boxed{\forall x \forall y [E(x,y) \rightarrow E(y,x)]}$	($\vdash_T A$)
9.	$\forall y [E(a,y) \rightarrow E(y,a)]$	($\forall$-elim,8)
10.	$E(a,b) \rightarrow E(b,a)$	($\forall$-elim,9)
11.	$E(b,a)$	($\rightarrow$-elim,3,10)
12.	$E(b,c) \rightarrow E(a,c)$	($\rightarrow$-elim,7,11)
13.	$E(a,c)$	($\rightarrow$-elim,4,12)
14.	$E(b,c) \rightarrow E(a,c)$	($\rightarrow$-intro,4,13)
15.	$E(a,b) \rightarrow (E(b,c) \rightarrow E(a,c))$	($\rightarrow$-intro,3,14)
16.	$\forall z [E(a,b) \rightarrow (E(b,z) \rightarrow E(a,z))]$	($\forall$-intro,15)
17.	$\forall y \forall z [E(a,y) \rightarrow (E(y,z) \rightarrow E(a,z))]$	($\forall$-intro,16)
18.	$\forall x \forall y \forall z [E(x,y) \rightarrow (E(y,z) \rightarrow E(x,z))]$	($\forall$-intro,17)

Figuur 14.2: $\vdash_T \forall x \forall y \forall z [E(x,y) \rightarrow (E(y,z) \rightarrow E(x,z))]$.

Naar aanleiding van het voorgaande voorbeeld geven we een precieze definitie van het begrip *afleidbaarheid in een geaxiomatiseerde theorie.*

14.1.7 DEFINITIE **Afleidbaarheid in een geaxiomatiseerde theorie**
Zij T een theorie die geaxiomatiseerd is door $\Delta \subseteq FORM$. Zij vervolgens $\Gamma \subseteq FORM$ en $F \in FORM$. Men zegt dan dat F in T afleidbaar *is* uit Γ, *notatie $\Gamma \vdash_T F$, indien $\Delta \cup \Gamma \vdash F$. Indien $\Gamma = \emptyset$, dan schrijft men $\vdash_T F$.*

14.2 Peano-aritmetiek

We geven nu een voorbeeld van een interessante theorie: de theorie van de *elementaire rekenkunde* of *Peano-aritmetiek.*

14.2.1 VOORBEELD Theorie van de Peano-aritmetiek.
Zij P de eerste-ordetaal met identiteit, de functiesymbolen $\mathbf{s}$, $\mathbf{+}$ en $\boldsymbol{\cdot}$, en de naam $\mathbf{0}$. P bevat geen predicaatsymbolen. De functiesymbolen $\mathbf{+}$ en $\boldsymbol{\cdot}$ zullen we als infixoperatoren beschouwen: we noteren dus $x \mathbf{+} y$ in plaats van $\mathbf{+}(x, y)$ en $x \boldsymbol{\cdot} y$ in plaats van $\boldsymbol{\cdot}(x, y)$.

De standaardinterpretatie voor P is de structuur:

$$\mathcal{N} = \langle \mathbb{N}; S, +, \cdot\,; 0 \rangle.$$

Hierin stelt S de *successorfunctie* voor. Deze functie is gedefinieerd door $S(x) = x + 1$. De functies $+$ en $\cdot$ stellen de *optelling*, respectievelijk het *product* voor, en het individu 0 is het natuurlijke getal nul.

De Peano-aritmetiek wordt gegeven door de volgende axioma's:

N1. $\forall x \forall y[\mathbf{s}(x) = \mathbf{s}(y) \to x = y]$,

N2. $\forall x \neg[\mathbf{s}(x) = \mathbf{0}]$,

N3. $\forall x[x \mathbf{+} \mathbf{0} = x]$,

N4. $\forall x[x \boldsymbol{\cdot} \mathbf{0} = \mathbf{0}]$,

N5. $\forall x \forall y[x \mathbf{+} \mathbf{s}(y) = \mathbf{s}(x \mathbf{+} y)]$,

N6. $\forall x \forall y[x \boldsymbol{\cdot} \mathbf{s}(y) = (x \boldsymbol{\cdot} y) \mathbf{+} x]$,

N7. $\{F[x{:=}\mathbf{0}] \wedge \forall y(F[x{:=}y] \to F[x{:=}\mathbf{s}(y)])\} \to \forall x F$.

De axioma's **N1** tot en met **N6** leggen de eigenschappen van de opvolgerfunctie S in relatie tot de $+$, de $\cdot$ en 0 vast. Het axiomaschema **N7** beschrijft het principe van volledige inductie voor natuurlijke getallen. In het vervolg zullen we de verzameling van deze axioma's en de theorie die erdoor bepaald wordt, aanduiden met N.

We geven een voorbeeld van een afleiding van een stelling van de theorie N. In het diagram in figuur 14.3 laten we zien dat $\vdash_N \forall x \forall y[\mathrm{s}(x) + y = \mathrm{s}(x + y)]$. In deze afleiding gebruiken we de volgende instantie van **N7**:

N7i. $\forall x\{\mathrm{s}(x) + \mathbf{0} = \mathrm{s}(x + \mathbf{0}) \wedge$
$\forall y[\mathrm{s}(x) + y = \mathrm{s}(x + y) \rightarrow \mathrm{s}(x) + \mathrm{s}(y) = \mathrm{s}(x + \mathrm{s}(y))] \rightarrow$
$\forall y[\mathrm{s}(x) + y = \mathrm{s}(x + y)]\}$.

We bewijzen de stelling dus via volledige inductie naar y. Om de afleiding niet nodeloos ingewikkeld te maken, zullen we soms in één keer tweemaal de regel $\forall$-elim toepassen op axioma **N5**. Dit noteren we dan als $2 * \forall$-elim. Verder zullen we de volgende formule afkorten als **N7ia**:

N7ia. $\mathrm{s}(a) + \mathbf{0} = \mathrm{s}(a + \mathbf{0}) \wedge$
$\forall y[\mathrm{s}(a) + y = \mathrm{s}(a + y) \rightarrow \mathrm{s}(a) + \mathrm{s}(y) = \mathrm{s}(a + \mathrm{s}(y))] \rightarrow$
$\forall y[\mathrm{s}(a) + y = \mathrm{s}(a + y)]$. ■

Men zou de vraag kunnen stellen of *iedere* theorie axiomatiseerbaar is. Helaas moet deze vraag ontkennend worden beantwoord. Voor de structuur $\mathcal{N}$ bewees de logicus Kurt Gödel in 1931 dat de theorie $Th(\mathcal{N})$ niet axiomatiseerbaar is. Dit resultaat staat bekend als *Gödel's onvolledigheidsstelling*. Preciezer geformuleerd: voor iedere verzameling N van axioma's voor de Peano-aritmetiek is er een zin F zodanig dat $\mathcal{N} \models F$ en zodanig dat *niet* $N \vdash F$. Nog anders geformuleerd: voor ieder stelsel axioma's voor de rekenkunde bestaat er een uitspraak over natuurlijke getallen die *waar* is, maar die niet uit die axioma's is af te leiden.

14.3 Opgaven

1. Bewijs voor de theorie uit voorbeeld 14.1.6:

   $$\vdash_T \forall x \forall y[E(x, y) \leftrightarrow x = y].$$

2. Bewijs voor de optelling in de Peano-aritmetiek N dat:

   $$\vdash_N \forall x \forall y[x + y = y + x].$$

   (Hint: gebruik de stelling die in figuur 14.3 is bewezen.)

3. Bewijs voor de vermenigvuldiging in de Peano-aritmetiek N dat:

   $$\vdash_N \forall x \forall y[x \cdot y = y \cdot x].$$

1.	**N3**, **N5** en **N7i**		
2.	$\mathrm{s}(a) + \mathbf{0} = \mathrm{s}(a)$		($\forall$-elim,**N3**)
3.	$a + \mathbf{0} = a$		($\forall$-elim,**N3**)
4.	$a + \mathbf{0} = a + \mathbf{0}$		($=$-intro)
5.	$a = a + \mathbf{0}$		(sub,3,4)
6.	$B: \ \mathrm{s}(a) + \mathbf{0} = \mathrm{s}(a + \mathbf{0})$		(sub,2,5)
7.		$\mathrm{s}(a) + b = \mathrm{s}(a + b)$	(inductiehyp.)
8.		$\mathrm{s}(a) + \mathrm{s}(b) = \mathrm{s}(\mathrm{s}(a) + b)$	($2 * \forall$-elim,**N5**)
9.		$\mathrm{s}(a) + \mathrm{s}(b) = \mathrm{s}(\mathrm{s}(a + b))$	(sub,7,8)
10.		$a + \mathrm{s}(b) = \mathrm{s}(a + b)$	($2 * \forall$-elim,**N5**)
11.		$a + \mathrm{s}(b) = a + \mathrm{s}(b)$	($=$-intro)
12.		$\mathrm{s}(a + b) = a + \mathrm{s}(b)$	(sub,10,11)
13.		$\mathrm{s}(a) + \mathrm{s}(b) = \mathrm{s}(a + \mathrm{s}(b))$	(sub,9,12)
14.	$\mathrm{s}(a) + b = \mathrm{s}(a + b) \rightarrow \mathrm{s}(a) + \mathrm{s}(b) = \mathrm{s}(a + \mathrm{s}(b))$		($\rightarrow$-intro,7,13)
15.	$I: \ \forall y[\mathrm{s}(a) + y = \mathrm{s}(a + y) \rightarrow \mathrm{s}(a) + \mathrm{s}(y) = \mathrm{s}(a + \mathrm{s}(y))]$		($\forall$-intro,14)
16.	**N7ia**		($\forall$-elim,**N7i**)
17.	$B \wedge I$		($\wedge$-intro,6,15)
18.	$\forall y[\mathrm{s}(a) + y = \mathrm{s}(a + y)]$		($\rightarrow$-elim,16,17)
19.	$\forall x \forall y[\mathrm{s}(x) + y = \mathrm{s}(x + y)]$		($\forall$-intro,18)

Figuur 14.3: $\vdash_N \forall x \forall y[\mathrm{s}(x) + y = \mathrm{s}(x + y)]$.

Deel III

Logisch Programmeren

Hoofdstuk 15

Resolutie in de Propositielogica

In dit hoofdstuk geven we een inleiding op het gebied van het automatisch bewijzen van theorema's. Het idee daarbij is dat een computerprogramma nagaat of een bepaalde bewering afleidbaar is of niet. Uiteraard is de constructie van dergelijke programma's, die men theoremabewijzers noemt, niet triviaal. Veel theoretisch onderzoek is en wordt daarom verricht naar de vraag hoe deze te construeren.

De indeling van dit hoofdstuk is als volgt. In paragraaf 15.1 wordt een algemene inleiding gegeven op het gebied van het automatisch bewijzen van theorema's. Daarna wordt in de paragrafen 15.2, 15.3 en 15.4 de zogenaamde *resolutiemethode* voor de propositielogica besproken. Dit is een natuurlijke-deductiemethode die bijzonder geschikt is voor het automatisch bewijzen van stellingen. De resolutiemethode bezit slechts één afleidingsregel: de *resolutieregel.*

15.1 Automatisch bewijzen van stellingen

Na de Tweede Wereldoorlog is zeer veel onderzoek gedaan naar het automatisch bewijzen van theorema's of stellingen met behulp van een computer. Men onderscheidt *bewijs-checkers* en *theoremabewijzers.* Een bewijs-checker controleert slechts of een gegeven afleiding correct is, terwijl een theoremabewijzer voor een gegeven bewering nagaat of deze afleidbaar is en eventueel een afleiding produceert. In zijn meest ideale vorm 'berekent' een theoremabewijzer in een eindige en liefst niet al te lange rekentijd een afleiding van de betreffende bewering, of stelt vast dat deze niet afleidbaar is. Dit alles relatief ten opzichte van de gebruikte afleidingsregels, axioma's en een eventuele theorie.

Het idee dat redeneren een soort rekenen is in een logische calculus, is al vrij oud. Men vindt dit onder andere in de geschriften van de duitse wiskundige en filosoof Gottfried Wilhelm Leibnitz (1646 – 1716). Afleidingen zouden in een dergelijke calculus berekend kunnen worden zoals het product van twee getallen in het decimale stelsel. Tot in het begin van deze eeuw geloofden gezaghebbende wiskundigen zoals Guiseppe Peano en David Hilbert, dat er een algoritme zou bestaan waarmee elk wiskundig probleem, mits goed geformuleerd, opgelost of 'uitgerekend' kon worden.

In de dertiger jaren bleek uit de onderzoekingen van Alonzo Church (1936) en Alan Turing (1936) dat zo'n algoritme niet bestaat. Zij bewezen, onafhankelijk van elkaar, dat er geen algoritme bestaat waarmee men de (on)vervulbaarheid van een willekeurige formule F uit de eerste-orde predicatenlogica in een eindig aantal stappen kan bepalen. Men zegt ook wel dat de eerste-orde predicatenlogica *onbeslisbaar* is. Nadere beschouwing van dit resultaat leert dat men wel een algoritme kan maken dat voor iedere onvervulbare formule F in een eindig aantal stappen de onvervulbaarheid vaststelt of bewijst. Is F echter vervulbaar, dan levert dit algoritme geen antwoord op in de zin dat het niet stopt voor de invoer F. Een dergelijke procedure noemt men een *semi-beslissingsprocedure*.

Een voorbeeld van een semi-beslissingsprocedure voor de eerste-orde predicatenlogica is de boommethode. Als F onvervulbaar is dan kan een boom voor F na een eindig aantal reductiestappen gesloten worden. In het geval F vervulbaar is, dan kan men in het algemeen steeds nieuwe reductiestappen uitvoeren, zonder dat de boom voor F gesloten kan worden.

De wiskunde kan beschreven worden in termen van de verzamelingentheorie en de verzamelingentheorie kan geformaliseerd worden in de eerste-orde logica. Dit heeft tot gevolg dat de wiskunde —alhoewel dit niet zo handig is— geformaliseerd kan worden in de eerste-orde predicatenlogica. Dit betekent dat iedere wiskundige stelling geformuleerd kan worden als een formule uit de eerste-orde predicatenlogica en ieder bewijs van een wiskundige stelling opgeschreven kan worden als een afleiding in de in de eerste-orde predicatenlogica. Deze afleiding kan axiomatisch zijn, of volgens een systeem van natuurlijke deductie. Uit de onbeslisbaarheidsresultaten van Church en Turing volgt dat de wiskunde als geheel onbeslisbaar is; anders gezegd, de veronderstelling van Peano en Hilbert dat er een algoritme voor de oplossing van alle 'goed-geformuleerde' wiskundige problemen zou bestaan, is hierdoor weerlegd. Bovenstaand resultaat houdt echter niet in dat alle delen van de wiskunde onbeslisbaar zijn. Zo is bijvoorbeeld door Alfred Tarski (1948) bewezen dat de elementaire vlakke meetkunde beslisbaar is. Hieruit volgt, dat men voor de elementaire vlakke meetkunde een theoremabewijzer kan maken.

Bij de constructie van theoremabewijzers maakt men primair gebruik van bewijsmethoden die ontleend zijn aan de logica. Deze worden meestal aangevuld met allerlei technieken om het bewijs sneller te vinden. Dit noemt men *heuristische methoden.* Theoremabewijzers die worden geconstrueerd in het kader van de kunstmatige intelligentie, maken ook vaak gebruik van theorieën uit de psychologie over het functioneren van het menselijke denken. Daarmee beoogt men de theoremabewijzer het menselijk redeneren te laten imiteren.

Voor de wiskunde bestaan er momenteel diverse theoremabewijzers. De hierin gebruikte logische methoden zijn gebaseerd op natuurlijke deductie, op de zogenaamde stelling van Herbrand op varianten van de resolutieregel (zie dit hoofdstuk en het volgende), en op heuristische methoden om 'overbodig werk' te vermijden. Doorgaans worden in een theoremabewijzer diverse bewijsmethoden gecombineerd. Momenteel kunnen met behulp van theoremabewijzers de meeste wiskunde- en logica-opgaven die men doorgaans aan eerstejaars studenten als oefenmateriaal geeft, bevredigend worden opgelost.

Theoremabewijzers worden toegepast in zogenaamde *kennisverwerkende systemen* en *expert systemen*. Kennisverwerkende systemen worden onder andere gebruikt bij het besturen van technische processen, zoals robotbesturing en besturing van onbemande ruimtevaartuigen. Expert systemen zijn computerprogramma's waarin kennis in de vorm van regels is opgeslagen. Dit soort systemen kan een gebruiker bijstaan bij het oplossen van bijvoorbeeld juridische of medische problemen.

15.2 De resolutieregel

In 1965 presenteerde J.A. Robinson een efficiënte methode om vast te stellen of een verzameling formules onvervulbaar is: de *resolutiemethode*. In feite is dit een *weerleggingsprocedure* net als de boommethode. Een weerleggingsprocedure is een methode om op een systematische wijze de onvervulbaarheid van een verzameling formules aan te tonen. De resolutiemethode is, in tegenstelling tot de boommethode, alleen toepasbaar op proposities die in conjunctieve normaalvorm staan. Daar staat tegenover dat de resolutiemethode slechts één afleidingsregel bezit, de zogenaamde *resolutieregel*.

We zullen aannemen dat propositionele formules steeds in conjunctieve normaalvorm staan en gerepresenteerd worden door verzamelingen van disjuncties van literalen. Een formule $F = D_1 \wedge \ldots \wedge D_n$ zal dus worden weergegeven als $F = \{D_1, \ldots, D_n\}$ waarin de D_i $(1 \leq i \leq n)$ disjuncties zijn van de vorm $L_{i1} \vee \ldots \vee L_{im_i}$. Deze disjuncties zullen we vaak ook voorstellen als verzamelingen: $D_i = \{L_{i1}, \ldots, L_{im_i}\}$. Dit is gerechtvaardigd aangezien iedere disjunctie D waarin een bepaalde literaal L meer dan één keer voorkomt, equivalent is met de disjunctie D' verkregen uit D door daarin op één na alle voorkomens van L te schrappen. We nemen derhalve aan dat geen enkele literaal meer dan één keer in dezelfde disjunctie voorkomt.

15.2.1 DEFINITIE **Complement van een literaal**
Zij L een literaal en p een propositiesymbool. Als $L = p$, dan is het complement *van L, notatie $\overline{L}$, gedefinieerd als $\neg p$; als $L = \neg p$, dan is $\overline{L} = p$.*

In de volgende definitie gebruiken we de volgende notatie: als D een disjunctie is en L een literaal, dan duiden we met $D - L$ de disjunctie aan waaruit L is verwijderd. Als we D als een verzameling opvatten, zou het dus meer correct zijn om te schrijven $D - \{L\}$.

15.2.2 DEFINITIE **Resolvent en resolutieregel**
Zij D_1 en D_2 disjuncties van literalen. Zij verder L een literaal zodanig dat $L \in D_1$ en $\overline{L} \in D_2$. Dan noemt men de verzameling:

$$\mathcal{R}(D_1, D_2) \;=\; (D_1 - L) \cup (D_2 - \overline{L})$$

de resolvent *van D_1 en D_2 naar L. Als $D_1 = \{L\}$ en $D_2 = \{\overline{L}\}$, dan is $\mathcal{R}(D_1, D_2)$ gelijk aan de lege disjunctie, notatie $\square$. Het voorschrift om de resolvent van twee disjuncties te bepalen wordt de* resolutieregel *genoemd.*

In de notatie $\mathcal{R}(D_1, D_2)$ komt niet tot uitdrukking naar welke literaal geresolveerd wordt. Door D_1, D_2 en $\mathcal{R}(D_1, D_2)$ te vergelijken wordt dit echter altijd duidelijk.

15.2.3 VOORBEELD Bepaling van een resolvent.

1. Zij $D_1 = \{p, q\}$ en $D_2 = \{\neg p, r\}$. De resolvent $\mathcal{R}(D_1, D_2)$ naar p is $\{q, r\}$. In formulevorm betekent dit: als $D_1 = (p \vee q)$ en $D_2 = (\neg p \vee r)$, dan $\mathcal{R}(D_1, D_2) = (q \vee r)$.
2. Voor de disjuncties $D_1 = \{p, q\}$ en $D_2 = \{p, q, \neg r\}$ bestaat geen resolvent.
3. $\mathcal{R}(p, \neg p) = \square$. ■

15.2.4 STELLING **Correctheid resolutieregel**
Als $\mathcal{R}(D_1, D_2)$ een resolvent is van de disjuncties D_1 en D_2, dan:

$$D_1, D_2 \models \mathcal{R}(D_1, D_2).$$

BEWIJS Stel $D_1 = (L \vee E_1)$ en $D_2 = (\overline{L} \vee E_2)$, waarin L een literaal is, en E_1 en E_2 disjuncties van literalen zijn. Nu is $\mathcal{R}(D_1, D_2) = E_1 \cup E_2$. Er zijn nu verschillende mogelijkheden.

Stel dat E_1 en E_2 beide niet-leeg zijn. Voor ieder model v voor D_1 en D_2 geldt ofwel $v(L) = 0$, ofwel $v(L) = 1$. In het eerste geval moet gelden $v(E_1) = 1$, en in het tweede geval $v(E_2) = 1$. Hieruit volgt $v(E_1 \cup E_2) = 1$.

Als E_1 en E_2 beide leeg zijn, dan bestaat er geen valuatie v die zowel D_1 als D_2 vervult. Hiermee is dit geval afgehandeld.

Als E_1 leeg is en E_2 niet-leeg, dan geldt voor een model v voor D_1 en D_2 dat $v(L) = 1$. Dit impliceert dat $v(E_2) = 1$, zodat het gestelde volgt. Evenzo voor het geval dat E_2 leeg is en E_1 niet-leeg. ∎

15.2.5 DEFINITIE **Resolutie-afleiding**

Zij D een disjunctie en F een niet-lege verzameling disjuncties. Een afleiding *van D uit F is een rij disjuncties:*

$$R_1, R_2, \ldots, R_n = D,$$

waarin iedere R_i $(1 \leq i \leq n)$ een disjunctie uit F is, òf de resolvent van twee disjuncties R_j en R_k waarvoor $1 \leq j, k \leq i$. Als er een resolutie-afleiding van D uit F bestaat, dan zegt men dat D afleidbaar *is uit F, notatie $F \vdash_{\mathcal{R}} D$. Tenslotte noemt men n de* lengte *van de afleiding.*

15.2.6 VOORBEELD Resolutie-afleiding.

Zij $F = \{\neg p \vee q,\ \neg q,\ p\}$. De rij disjuncties:

1.	$\neg p \vee q$	(element F)
2.	$\neg q$	(element F)
3.	p	(element F)
4.	$\neg p$	(resolvent 1,2)
5.	$\square$	(resolvent 3,4)

is een afleiding van $\square$ uit F, zodat $F \vdash_{\mathcal{R}} \square$. ∎

In de volgende paragraaf zullen we zien dat we vooral geïnteresseerd zijn in afleidingen van $F \vdash_{\mathcal{R}} \square$. Immers, in dat geval is F onvervulbaar volgens stelling 15.4.2, die we in paragraaf 15.3 zullen aantonen.

15.2.7 STELLING **Correctheid resolutie**

Zij $F \in PROP$ een formule in conjunctieve normaalvorm en D een disjunctie. Dan geldt:

$$F \vdash_{\mathcal{R}} D \quad \Rightarrow \quad F \models D.$$

BEWIJS *Door middel van inductie over de lengte van de afleiding.* ∎

15.3 De resolutiemethode

Bij stelling 15.4.2 zullen we bewijzen dat een formule F onvervulbaar is dan en slechts dan als $F \vdash_{\mathcal{R}} \square$. Op dit resultaat kan een elegant algoritme worden gebaseerd waarmee men in een eindig aantal stappen kan nagaan of een formule F onvervulbaar is of niet. Bovendien bevat het algoritme slechts één gemakkelijk te implementeren afleidingsregel, namelijk de resolutieregel. Het algoritme is als volgt.

Resolutie-algoritme

Stap 1 Genereer, indien mogelijk, een nieuwe resolvent R van disjuncties uit F; anders **Stop**: F is vervulbaar.

Stap 2 Als $R = \square$ dan **Stop**: F is onvervulbaar; anders $F := F \cup \{R\}$ en ga naar **Stap 1**.

Het algoritme stopt in elk geval als de invoer F onvervulbaar is, want dan wordt de lege disjunctie $\square$ gegenereerd (stelling 15.4.2). Als F vervulbaar is, dan stopt het algoritme ook. Immers, aangezien F eindig veel disjuncties bevat, kunnen er bij stap 1 slechts een eindig aantal *nieuwe* resolventen worden gegenereerd. Zodra er geen nieuwe resolventen meer gegenereerd kunnen worden —dit is in een eindig aantal stappen te controleren—, stopt het algoritme en is F vervulbaar.

15.3.1 VOORBEELD Resolutie-afleiding.
We laten met behulp van resolutie zien dat:

$$p \rightarrow q, q \rightarrow r \vdash p \rightarrow r.$$

We moeten dus aantonen dat de verzameling disjuncties:

$$F = \{\neg p \vee q,\ \neg q \vee r,\ p,\ \neg r\}$$

onvervulbaar is. Dit doen we door $F \vdash \square$ af te leiden.

1. $\neg p \vee q$ (element F)
2. $\neg q \vee r$ (element F)
3. p (element F)
4. $\neg r$ (element F)
5. q (resolvent 1,3)
6. r (resolvent 2,5)
7. $\square$ (resolvent 4,6) ■

15.3.2 VOORBEELD Resolutie-afleiding.
We laten met behulp van resolutie zien dat:

$$(p \to \neg r) \vee (p \to q), r \to p \vdash r \to q.$$

Hiertoe leiden we af dat de verzameling:

$$F \;=\; \{\neg p \vee \neg r \vee q,\ \neg r \vee p,\ r,\ \neg q\}$$

onvervulbaar is. Dit doen we weer door $F \vdash_{\mathcal{R}} \Box$ af te leiden.

1. $\neg p \vee q \vee \neg r$ (element F)
2. $p \vee \neg r$ (element F)
3. $\neg q$ (element F)
4. r (element F)
5. p (resolvent 2,4)
6. $q \vee \neg r$ (resolvent 1,5)
7. q (resolvent 4,6)
8. $\Box$ (resolvent 3,7) ■

15.4 Volledigheid van de resolutiemethode

We zullen in deze paragraaf de stelling bewijzen die ten grondslag ligt aan de resolutiemethode.

Eerst introduceren we een algoritme waarmee eindige verzamelingen F van disjuncties kunnen worden 'vereenvoudigd'. Het idee achter dit algoritme is dat alle disjuncties die geen invloed hebben op het (on)vervulbaar zijn van F, uit F worden verwijderd met als resultaat de verzameling F_0.

Vereenvoudigingsalgoritme

Stap 1 $F_0 := F$.

Stap 2 Verwijder uit F_0 alle disjuncties die een paar complementaire literalen L en $\overline{L}$ bevatten.

Stap 3 Verwijder uit F_0 alle disjuncties die een literaal L bevatten waarvan het complement $\overline{L}$ in geen enkele andere disjunctie in F voorkomt.

15.4.1 STELLING *Zij $F \in PROP$ een formule in conjunctieve normaalvorm, en zij F_0 het resultaat van het toepassen van het vereenvoudigingsalgoritme op F, dan geldt:*

$$F_0 \text{ is onvervulbaar} \quad \Leftrightarrow \quad F \text{ is onvervulbaar.}$$

Als $F_0 = \emptyset$ dan wordt hierbij aangenomen dat F_0 vervulbaar is.

BEWIJS Wordt aan de lezer overgelaten. ■

15.4.2 STELLING **Volledigheid resolutie**
Zij $F \in PROP$ een formule in conjunctieve normaalvorm. Dan geldt:

$$F \vdash_{\mathcal{R}} \Box \quad \Leftrightarrow \quad F \text{ is onvervulbaar.}$$

BEWIJS

($\Rightarrow$) *We bewijzen de uitspraak met een redenering uit het ongerijmde. Stel dat* $F \vdash_{\mathcal{R}} \Box$ *en dat* F *vervulbaar is. Omdat* $F \vdash_{\mathcal{R}} \Box$*, bestaat er een afleiding:*

$$R_1, R_2, \ldots, R_n = \Box.$$

Uit de definitie van een afleiding volgt dat $F \vdash_{\mathcal{R}} R_i$ *voor* $1 \leq i \leq n$*. Toepassing van stelling 15.2.7 levert dat* $F \models R_i$ *voor* $1 \leq i \leq n$*. Als* F *vervulbaar is, bestaat er een valuatie* v *zodanig dat* $v(F) = 1$*. In combinatie met* $F \models R_i$ *volgt hieruit dat* $v(R_i) = 1$ *voor alle* $1 \leq i \leq n$*. Omdat* $F \vdash_{\mathcal{R}} \Box$*, is er een literaal* L *zodanig dat* $L = R_j$ *en* $\overline{L} = R_k$ *voor* $1 \leq j < k < n$*. Dit betekent dat* $v(L) = v(\overline{L}) = 1$*, hetgeen onmogelijk is.*

($\Leftarrow$) *Eerst passen we het vereenvoudigingsalgoritme toe op* F *met als resultaat een deelverzameling* F_0 *van* F *die aan de volgende eigenschap voldoet:*

$$F_0 \text{ is onvervulbaar} \quad \Leftrightarrow \quad F \text{ is onvervulbaar.}$$

Omdat het evident is dat $F \vdash_{\mathcal{R}} \Box$ *indien* $F_0 \vdash_{\mathcal{R}} \Box$*, volgt nu dat de stelling bewezen is, als we kunnen aantonen:*

$$F_0 \text{ is onvervulbaar} \quad \Rightarrow \quad F_0 \vdash_{\mathcal{R}} \Box.$$

Neem aan dat F_0 *de propositiesymbolen* $q_1, \ldots, q_n$ *bevat, waarbij* $n \geq 0$*. Het bewijs verloopt via volledige inductie naar* n*.*

- *Basisstap:* $n = 0$*.*

  In dit geval is $F_0 = \emptyset$*. De bewering geldt omdat* $\emptyset$ *per definitie vervulbaar is.*

- *Inductiestap:* $n > 0$*.*

  De inductiehypothese luidt dat de stelling klopt voor alle formules F_0 *die minder dan* n *verschillende propositiesymbolen bevatten.*

  We definiëren nu een verzameling G *van disjuncties, die aan de volgende eigenschappen voldoet:*

$$F_0 \text{ is onvervulbaar} \quad \Rightarrow \quad G \text{ is onvervulbaar,} \qquad (15.1)$$

$$G \text{ is onvervulbaar} \quad \Rightarrow \quad G \vdash_{\mathcal{R}} \Box, \qquad (15.2)$$

$$G \vdash_{\mathcal{R}} \Box \quad \Rightarrow \quad F_0 \vdash_{\mathcal{R}} \Box. \qquad (15.3)$$

  Als dit kan worden aangetoond, dan is de stelling bewezen.

  Het idee is dat G *alle disjuncties bevat die in één resolutiestap met betrekking tot* q_n *uit* F_0 *kunnen worden verkregen, en alle disjuncties die* q_n *(of* $\neg q_n$*) niet bevatten.*

  Constructie van G.

  Omdat F_0 *het propositiesymbool* q_n *bevat en vereenvoudigd is, kunnen we* F_0 *schrijven als:*

$$\begin{aligned} F_0 \quad = \quad & \{q_n \vee C_1, \ldots, q_n \vee C_k, \neg q_n \vee D_1, \ldots, \neg q_n \vee D_l, \\ & E_1, \ldots, E_m\}, \end{aligned}$$

  waarbij $C_1, \ldots, C_k, D_1, \ldots, D_l$ *en* $E_1, \ldots, E_m$ *(*$k, l \geq 1$ *en* $m \geq 0$*) disjuncties zijn waarin* q_n *niet voorkomt. Definieer* G *als de verzameling:*

$$G \quad = \quad \{C_i \cup D_j \mid 1 \leq i \leq k \text{ en } 1 \leq j \leq l\} \cup \{E_1, \ldots, E_m\}.$$

  G *bestaat dus uit alle resolventen van* $q_n \vee E_i$ *en* $\neg q_n \vee E_j$*, en alle disjuncties* $E_1, \ldots, E_m$*. Dit betekent dat het propositiesymbool* q_n *niet in* G *voorkomt.*

  Merk op dat $F_0 \vdash_{\mathcal{R}} \Box$*, als er een* i *en een* j *bestaat zodanig dat* $C_i \cup D_j = \Box$ *(*$1 \leq i \leq k$ *en* $1 \leq j \leq l$*). In dat geval is de inductiestap voltooid. Neem daarom in het vervolg van het bewijs aan dat dergelijke* i *en* j *niet bestaan.*

Bewijs van bewering (15.1).

We bewijzen met contrapositie dat G onvervulbaar is, als F_0 *dat is. Als G vervulbaar is, dan bestaat er een valuatie v zodanig dat* $v(E_1) = \ldots = v(E_m) = 1$, *en* $v(C_i \cup D_j) = 1$ *voor alle* $1 \leq i \leq k$ *en* $1 \leq j \leq l$ *(hier maken we gebruik van de aanname dat* $C_i \cup D_j \neq \Box$, *zodat het zinvol is om te praten over* $v(C_i \cup D_j)$*). Hieruit volgt dat* $v(C_i) = 1$ *voor alle* $1 \leq i \leq k$, *of* $v(D_j) = 1$ *voor alle* $1 \leq j \leq l$.

In het geval dat $v(C_i) = 1$ *voor alle* $1 \leq i \leq k$, *dan wordt de verzameling* F_0 *vervuld door een valuatie w waarvoor geldt dat* $w(q_s) = v(q_s)$ *voor* $1 \leq s \leq n-1$, *en* $w(q_n) = 0$.

In het geval dat $v(D_j) = 1$ *voor alle* $1 \leq j \leq l$, *dan wordt de verzameling* F_0 *vervuld door een valuatie w waarvoor geldt dat* $w(q_s) = v(q_s)$ *voor* $1 \leq s \leq n-1$, *en* $w(q_n) = 1$.

Hieruit volgt dat de verzameling F_0 *vervulbaar is.*

Bewijs van bewering (15.2).

G is een verzameling van disjuncties waarin het propositiesymbool q_n *niet voorkomt. Volgens de inductiehypothese geldt dus voor G:*

$$G \text{ is onvervulbaar} \quad \Rightarrow \quad G \vdash_{\mathcal{R}} \Box.$$

Bewijs van bewering (15.3).

Als $G \vdash_{\mathcal{R}} \Box$, *dan is er een afleiding* α *van* $\Box$ *uit G. De elementen van G die in* α *voorkomen, zijn of een resolvent van twee elementen uit* F_0, *of een element uit* F_0. *Door de disjuncties uit* F_0 *achter elkaar te zetten en daarachter de rij* α *verkrijgt men een afleiding van* $\Box$ *uit* F_0. *Dit betekent dat* $F_0 \vdash_{\mathcal{R}} \Box$.

Hiermee is de stelling bewezen. ■

15.4.3 COROLLARIUM *Zij* $F \in PROP$ *een formule in conjunctieve normaalvorm, en zij* F_0 *het resultaat van het toepassen van het vereenvoudigingsalgoritme op F, dan geldt:*

$$F_0 \vdash_{\mathcal{R}} \Box \quad \Leftrightarrow \quad F \vdash_{\mathcal{R}} \Box.$$

BEWIJS Wordt aan de lezer overgelaten. ■

Om te bepalen of een eindige verzameling disjuncties F onvervulbaar is, volstaat het dus om te bepalen of voor de vereenvoudiging F_0 van F geldt $F_0 \vdash_{\mathcal{R}} \Box$.

15.5 Opgaven

1. Zij p, q en r propositiesymbolen. Bewijs door middel van resolutie dat:
   (a) $\vdash (p \to (q \to r)) \to (q \to (p \to r))$.
   (b) $\vdash ((p \vee (p \to q)) \to q) \to q$.
   (c) $p \leftrightarrow q, (\neg p \wedge \neg q) \to \neg r, r \vdash q$.
   (d) $p \wedge q \to \neg r, \neg(r \to s), q \vee s \vdash \neg p$.

2. Bewijs stelling 15.2.7.
3. Bewijs stelling 15.4.1.
4. Bewijs corollarium 15.4.3.

Hoofdstuk 16

Resolutie in de Predicatenlogica

In dit hoofdstuk zullen we de resolutiemethode generaliseren voor de predicatenlogica. Hierbij moet rekening worden gehouden met het feit dat de predicatenlogica kwantoren bezit en met het feit dat disjuncties variabelen kunnen bevatten. Het eerstgenoemde probleem wordt opgevangen door formules in zogenaamde *Skolemnormaalvorm* (zie paragraaf 16.1) te brengen waardoor de kwantoren worden geëlimineerd. Daarna kan de kwantorvrije formule in conjunctieve normaalvorm worden gebracht. Het probleem van de variabelen wordt opgelost door de resolutieregel uit te breiden met *unificatie*. Met behulp van unificatie kunnen geschikte substituties voor variabelen in disjuncties worden gevonden ter bepaling van resolventen. De theorie van substitutie en unificatie zal worden behandeld in paragraaf 16.2. Voor unificatie bestaan verschillende algoritmen. We behandelen in paragraaf 16.3 het algoritme van Robinson en geven er een correctheidsbewijs voor. Daarna wordt in paragraaf 16.4 de resolutieregel voor de predicatenlogica geformuleerd. Met behulp van deze resolutieregel kan op eenvoudige wijze een resolutie-algoritme worden verkregen. Dit algoritme en voorbeelden van toepassingen ervan zijn het onderwerp van paragraaf 16.5. Het hoofdstuk wordt besloten met het bewijs van de volledigheid van de resolutie (§16.6).

16.1 De Skolemnormaalvorm

Om in de propositielogica de resolutiemethode te kunnen toepassen dienen de formules in conjunctieve normaalvorm te staan. Dat is ook het geval in de predicatenlogica. Bovendien is de resolutiemethode voor de predicatenlogica beperkt tot formules die geen identiteit (=) bevatten. Om een willekeurige formule uit de predicatenlogica te kunnen schrijven als een verzameling disjuncties moeten eerst de kwantoren worden weggewerkt. Dit proces noemt men *skolemiseren*. De uiteindelijke normaalvorm die na het skolemiseren ontstaat, noemt men de *Skolemnormaalvorm*. Deze normaalvorm berust op de volgende ideeën.

Iedere formule $F \in FORM$ kan getransformeerd worden in een prenexnormaalvorm F^p met een prenex bestaande uit een rij kwantoren en een matrix die geen kwantoren bevat (zie hoofdstuk 10). De zo verkregen formule F^p is equivalent met F. De matrix van F^p is een formule zonder kwantoren en kan dus getransformeerd worden in een conjunctieve normaalvorm (zie hoofdstuk 4). De zo verkregen formule $F^{p,c}$ is equivalent met F^p en dus ook met F. Merk op dat zowel de prenexnormaalvorm als de conjunctieve normaalvorm van een formule F niet éénduidig zijn bepaald: bij één formule kan men meerdere formules in prenexnormaalvorm, respectievelijk conjunctieve normaalvorm vinden.

Het idee van de Skolemnormaalvorm is om de formule $F^{p,c}$ om te zetten in een formule F^s die geen existentiële kwantoren meer bevat. We lichten dit toe aan een voorbeeld.

16.1.1 VOORBEELD Skolemnormaalvorm.
Zij $F = \forall x \exists y P(x, y)$ voor een tweeplaatsig predicaatsymbool P. Deze formule brengt tot uitdrukking dat er voor iedere x een y bestaat zodanig dat $P(x, y)$. Het is evident dat deze y over het algemeen afhankelijk is van de x. Met andere woorden: y is een functie van x. Stel $y = f(x)$ en beschouw de formule $F^s = \forall x P(x, f(x))$ voor een nieuw functiesymbool f. De formule F^s noemt men dan de *Skolemnormaalvorm* van F. Het functiesymbool f dat in de formule F^s is geïntroduceerd, noemt men een *Skolemfunctie.* De Skolemfunctie f brengt dus tot uitdrukking dat y afhankelijk is van x.

De formules F en F^s zijn niet equivalent. Immers, zij $\mathcal{A}$ een structuur met $|\mathcal{A}| = \mathbb{N}$, $P^{\mathcal{A}} = <$ en $f^{\mathcal{A}}(n) = n$ voor $n \in \mathbb{N}$, dan geldt evident dat $\mathcal{A} \models F$, maar *niet* $\mathcal{A} \models F^s$.

Gelukkig is er een belangrijke eigenschap die de formules F en F^s beide bezitten:

$$F \text{ is onvervulbaar} \quad \Leftrightarrow \quad F^s \text{ is onvervulbaar.}$$

Dit is de eigenschap van F en F^s die we nodig blijken te hebben.

Zij nu $F = \forall x \forall y \exists z Q(x, y, z)$, dan kan men een Skolemnormaalvorm F^s van F vinden door een tweeplaatsige Skolemfunctie g te introduceren; de variabele z is namelijk afhankelijk van x en y. Dit geeft $F^s = \forall x \forall y Q(x, y, g(x, y))$. Er geldt weer dat F onvervulbaar is dan en slechts dan als F^s onvervulbaar is.

Als er geen universele kwantoren vóór de existentiële kwantor staan, dan introduceert men een *nulplaatsige* functie, ofwel een individuele constante. Als $F = \exists x R(x)$, dan geldt dus dat $F^s = R(a)$. Ook in dit geval noemt men a de Skolemfunctie.

In alle drie de gevallen zegt men dat de existentiële kwantor is *geskolemiseerd* door introductie van een Skolemfunctie. ∎

Het idee van de Skolemnormaalvorm moge nu duidelijk zijn. Voor een formule $F \in FORM$ bepaalt men eerst de formule $F^{p,c}$. Daarna verwijdert men één voor één de existentiële kwantoren uit de prenex van $F^{p,c}$ door deze op de wijze die hierboven is geschetst, te skolemiseren. We zetten alles nog eens op een rijtje in de volgende definities.

16.1.2 Definitie **Skolemiseren**
Zij $F \in FORM$ met betrekking tot een eerste-ordetaal L en zij F van de vorm:

$$\forall y_1 \ldots \forall y_{n-1} \exists y_n M,$$

waarin $n \geq 1$ en $M \in FORM$. Zij vervolgens f een individuele constante (als $n = 1$) of een functiesymbool (als $n > 1$) waarvoor geldt dat f niet voorkomt in L. Dan noemt men de formule:

$$\forall y_1 \ldots \forall y_{n-1} M[y_n := f(y_1, \ldots, y_{n-1})]$$

het resultaat van het skolemiseren *van de existentiële kwantor $\exists y_n$. Het symbool f noemt men een* Skolemfunctie.

Uit de bovenstaande definitie blijkt dat een formule F' die het resultaat is van het skolemiseren van een existentiële kwantor uit een formule F uit een eerste-ordetaal L, geen formule is uit L. De formule F' behoort tot de taal L' die de uitbreiding is van L met de Skolemfunctie f.

16.1.3 Definitie **Skolemnormaalvorm**
Zij $F \in FORM$. Een Skolemnormaalvorm *F^s van F is een formule die verkregen wordt door:*

1. *F in een prenexnormaalvorm $F^{p,c}$ te brengen waarvan de matrix in conjunctieve normaalvorm staat, en daarna*
2. *één voor één de existentiële kwantoren in de prenex van $F^{p,c}$ te skolemiseren.*

16.1.4 Voorbeeld Skolemnormaalvorm.

1. Zij $F = \exists x \forall y P(x, y) \rightarrow \exists z Q(z)$. De volgende twee formules zijn prenexnormaalvormen van F:

$$\begin{aligned} F_1^{p,c} &= \forall x \exists y[\neg P(x, y) \vee Q(y)], \\ F_2^{p,c} &= \forall x \exists y \exists z[\neg P(x, y) \vee Q(z)]. \end{aligned}$$

De twee hiermee corresponderende Skolemnormaalvormen zijn:

$$\begin{aligned} F_1^s &= \forall x[\neg P(x, f(x)) \vee Q(f(x))], \\ F_2^s &= \forall x[\neg P(x, f(x)) \vee Q(g(x))]. \end{aligned}$$

Dit illustreert dat zowel prenexnormaalvormen, als Skolemnormaalvormen niet éénduidig bepaald zijn. Verder geldt dat $F_1^{p,c} \approx F_2^{p,c}$, maar *niet* $F_1^s \approx F_2^s$ (ga na).

2. De Skolemnormaalvorm van:

$$\exists x \forall y \exists u \forall v \forall w \exists z M(x, y, u, v, w, z),$$

is gelijk aan:

$$\forall y \forall v \forall w M(a, y, f(y), v, w, g(y, v, w)).$$

Hierin zijn a, f en g de Skolemfuncties. ∎

16.1.5 STELLING *Zij $F \in ZIN$ met betrekking tot een eerste-ordetaal L, en zij L' de uitbreiding van L met de Skolemfuncties die in de Skolemnormaalvorm F^s van F voorkomen. Dan geldt voor L' dat:*

$$F \textit{ is onvervulbaar} \quad \Leftrightarrow \quad F^s \textit{ is onvervulbaar.}$$

BEWIJS *We bewijzen de volgende bewering. Zij $G \in FORM$ een formule in prenexnormaalvorm waarvan de prenex tenminste één existentiële kwantor bevat, en zij G' het resultaat van het skolemiseren van de eerste existentiële kwantor in G. Dan geldt:*

$$G \textit{ is vervulbaar} \quad \Leftrightarrow \quad G' \textit{ is vervulbaar.}$$

Als dit is bewezen, dan volgt de stelling door volledige inductie naar het aantal existentiële kwantoren in de prenex van de formule $F^{p,c}$ (een prenexnormaalvorm van F waarvan de matrix in conjunctieve normaalvorm staat). Hierbij gebruikt men dat $F \approx F^{p,c}$.

Stel dat voor een $M \in FORM$, $n \geq 1$ en een Skolemfunctie f:

$$\begin{aligned} G &= \forall y_1 \ldots \forall y_{n-1} \exists y_n M, \\ G' &= \forall y_1 \ldots \forall y_{n-1} M[y_n := f(y_1, \ldots, y_{n-1})]. \end{aligned}$$

We zullen nu bewijzen dat G vervulbaar is dan en slechts dan als G' vervulbaar is.

($\Rightarrow$) *Als G vervulbaar is, dan is er een interpretatie $\langle \mathcal{A}, \beta \rangle$ zodanig dat:*

$$\forall a_1 \in |\mathcal{A}| \ \ldots \ \forall a_{n-1} \in |\mathcal{A}| \ \exists a_n \in |\mathcal{A}| \cdot \langle \mathcal{A}, \beta[y_1 \mapsto a_1] \cdots [y_{n-1} \mapsto a_{n-1}][y_n \mapsto a_n] \rangle \models M.$$

Zij nu $\mathcal{B}$ een structuur die gelijk is aan $\mathcal{A}$, behalve dat voor gegeven $a_1, \ldots, a_{n-1} \in \mathcal{A}$:

$$f^{\mathcal{B}}(a_1, \ldots, a_{n-1}) = a_n,$$

voor een $a_n \in \mathcal{A}$ *zodanig dat:*

$$\langle \mathcal{A}, \beta[y_1 \mapsto a_1]\cdots[y_{n-1} \mapsto a_{n-1}][y_n \mapsto a_n]\rangle \models M.$$

Dan is:

$$f(y_1, \ldots, y_{n-1})^{\mathcal{B}, \beta[y_1 \mapsto a_1]\cdots[y_{n-1} \mapsto a_{n-1}]} \quad = \quad a_n.$$

Uit het bovenstaande en de substitutiestelling volgt dat:

$$\forall a_1 \in |\mathcal{A}| \ \ldots \ \forall a_{n-1} \in |\mathcal{A}| \cdot$$
$$\langle \mathcal{B}, \beta[y_1 \mapsto a_1]\cdots[y_{n-1} \mapsto a_{n-1}]\rangle \models M[y_n := f(y_1, \ldots, y_{n-1})].$$

Hieruit volgt $\langle \mathcal{B}, \beta\rangle \models G'$, *zodat* G' *vervulbaar is.*

($\Leftarrow$) *Als* G' *vervulbaar is, dan is er een interpretatie* $\langle \mathcal{A}, \beta\rangle$ *zodanig dat:*

$$\forall a_1 \in |\mathcal{A}| \ \ldots \ \forall a_{n-1} \in |\mathcal{A}| \cdot$$
$$\langle \mathcal{A}, \beta[y_1 \mapsto a_1]\cdots[y_{n-1} \mapsto a_{n-1}]\rangle \models M[y_n := f(y_1, \ldots, y_{n-1})].$$

Passen we de substitutiestelling (10.2.5) toe, en stellen we:

$$f(y_1, \ldots, y_{n-1})^{\mathcal{A}, \beta[y_1 \mapsto a_1]\cdots[y_{n-1} \mapsto a_{n-1}]} \quad = \quad a,$$

dan verkrijgen we:

$$\forall a_1 \in |\mathcal{A}| \ \ldots \ \forall a_{n-1} \in |\mathcal{A}| \cdot$$
$$\langle \mathcal{A}, \beta[y_1 \mapsto a_1]\cdots[y_{n-1} \mapsto a_{n-1}][y_n \mapsto a]\rangle \models M,$$

Hieruit volgt:

$$\forall a_1 \in |\mathcal{A}| \ \ldots \ \forall a_{n-1} \in |\mathcal{A}| \ \exists a_n \in |\mathcal{A}| \cdot$$
$$\langle \mathcal{A}, \beta[y_1 \mapsto a_1]\cdots[y_{n-1} \mapsto a_{n-1}][y_n \mapsto a_n]\rangle \models M,$$

Dit betekent:

$$\langle \mathcal{A}, \beta\rangle \models \forall y_1 \cdots \forall y_{n-1} \exists y_n M,$$

zodat G *vervulbaar is.*

Hiermee is de stelling bewezen. ∎

Een zin F in Skolemnormaalvorm heeft de volgende vorm:

$$F = \forall y_1, \ldots, \forall y_n [D_1 \wedge \ldots \wedge D_n].$$

Hierin zijn de conjunctieleden D_i $(1 \leq i \leq n)$ *disjuncties* van literalen. Vaak wordt deze Skolemnormaalvorm genoteerd als de verzameling van haar disjuncties:

$$F = \{D_1, \ldots, D_n\}.$$

De universele kwantoren worden dan weggelaten. Men kan de vrije variabelen in de disjuncties daarbij zodanig kiezen dat in verschillende disjuncties verschillende variabelen voorkomen. Dit wordt gerechtvaardigd door het feit dat voor alle formules $A, B \in \mathit{FORM}$ het volgende geldt:

$$\begin{aligned} \forall x[A \wedge B] &\approx \forall x A \wedge \forall x B, \\ &\approx \forall x A \wedge \forall y B[x := y], \\ &\approx \forall x \forall y[A \wedge B[x := y]], \end{aligned}$$

voor een geschikte variabele y zodanig dat $y \notin VV(B)$.

16.1.6 DEFINITIE **Disjunctie**
Zij $F = \{D_1, \ldots, D_n\}$ $(n \in \mathbb{N})$ *een zin in Skolemnormaalvorm. De elementen* D_i $(1 \leq i \leq n)$ *van* F *worden de* disjuncties *van* F *genoemd.*

In Engelstalige literatuur worden disjuncties *clauses* genoemd.

16.1.7 VOORBEELD Disjuncties.
Zij F de formule:

$$\forall x \exists y \exists z[\neg(P(x,z) \rightarrow Q(x,y)) \vee R(x,y,z)].$$

Een Skolemnormaalvorm F^s van F is:

$$\forall x[(P(x,g(x)) \vee R(x,f(x),g(x))) \wedge (\neg Q(x,f(x)) \vee R(x,f(x),g(x)))].$$

Men kan F^s als volgt schrijven als een verzameling disjuncties:

$$\{P(x,g(x)) \vee R(x,f(x),g(x)),\ \neg Q(y,f(y)) \vee R(y,f(y),g(y))\}.$$

Merk op dat de variabele x in de tweede disjunctie is herbenoemd tot y. ■

16.2 Substitutie en unificatie

Bij resolutie in de predicatenlogica beperkt men zich tot zinnen die in Skolemnormaalvorm staan. Hierbij wordt zo'n zin over het algemeen gerepresenteerd als een verzameling van disjuncties. Net als in het hoofdstuk over resolutie voor de propositielogica is gedaan, kunnen disjuncties worden gerepresenteerd als verzamelingen van literalen.

Als een zin in Skolemnormaalvorm uitsluitend gronddisjuncties bevat, dan wordt de resolvent van twee disjuncties op dezelfde manier bepaald als in de propositielogica. Dus de resolvent van $P(f(a))$ en $\neg P(f(a)) \vee Q(a)$ naar $P(f(a))$ is gelijk aan $Q(a)$. Voor disjuncties die variabelen bevatten wordt dit anders. We illustreren dit in het volgende voorbeeld.

16.2.1 VOORBEELD De resolvent van disjuncties met variabelen.
Beschouw de volgende twee disjuncties.

$$\begin{aligned} D_1 &= P(x) \vee Q(x), \\ D_2 &= \neg P(f(y)) \vee R(y). \end{aligned}$$

Deze disjuncties kunnen niet zonder meer geresolveerd worden, want D_1 en D_2 bevatten geen complementaire literalen. Als we echter op D_1 en D_2 de substitutie:

$$\sigma = [x{:=}f(a), y{:=}a]$$

toepassen (dat wil zeggen, $[x{:=}f(a)]$ en $[y{:=}a]$ *simultaan* toepassen), dan resulteert dit in de gronddisjuncties:

$$\begin{aligned} D_1' &= D_1\sigma = P(f(a)) \vee Q(f(a)), \\ D_2' &= D_2\sigma = \neg P(f(a)) \vee R(a), \end{aligned}$$

met de complementaire literalen $P(f(a))$ en $\neg P(f(a))$. D_1' en D_2' zijn grondinstanties van D_1, respectievelijk D_2. D_1' en D_2' kunnen nu geresolveerd worden naar $P(f(a))$ met resolvent:

$$R_1 = Q(f(a)) \vee R(a).$$

Als we echter op D_1 en D_2 de substitutie $\theta = [x{:=}f(y)]$ toepassen, dan verkrijgen we de instanties $D_1'' = D_1\theta$ en $D_2'' = D_2\theta$ met $D_1'' = P(f(y)) \vee Q(f(y))$ en $D_2'' = \neg P(f(y)) \vee R(y)$. Dit betekent dat $D_1\theta$ en $D_2\theta$ resolveerbaar zijn naar $P(f(y))$ met resolvent $R_2 = Q(f(y)) \vee R(y)$.

Merk op dat $R_1 = R_2\lambda$ voor $\lambda = [y{:=}a]$. In resolvent R_2 is nog geen binding gemaakt voor de variabele y, in R_1 wel: daar is y gebonden aan a. We zullen zien dat de keuze voor een substitutie die minder vastlegt en in die zin algemener is, de meest verstandige is. ■

Uit het bovenstaande blijkt, dat we om een resolutieprocedure te verkrijgen voor de predicatenlogica, de resolutieprocedure voor de propositielogica moeten uitbreiden met een manier om geschikte substituties te vinden. Om deze reden zullen we substituties nader onder de loep nemen.

Een substitutie is een afbeelding die termen afbeeldt op termen, en formules op formules (zie ook hoofdstuk 8). Bij een enkelvoudige substitutie van de vorm $[y{:=}t]$ worden alle vrije voorkomens van de variabele y in een term of formule —na eventuele herbenoeming van de gebonden variabelen— vervangen door de term t. De substituties die we hier zullen beschouwen, zijn *simultane* substituties. Hierbij worden alle voorkomens van n verschillende variabelen gelijktijdig vervangen door n termen ($n \geq 0$). Zo'n substitutie noteren we als $[y_1{:=}t_1, \ldots, y_n{:=}t_n]$. Dergelijke simultane substituties zullen we kortweg *substituties* noemen, en aanduiden met Griekse letters. Een substitutie zullen we formeel opvatten als een *verzameling*. Dit betekent dat bijvoorbeeld $[x_3{:=}t_1, x_5{:=}t_2]$ en $[x_5{:=}t_2, x_3{:=}t_1]$ gelijke substituties zijn. Tenslotte nemen we aan dat er in een substitutie geen paren van de vorm $y{:=}y$ voorkomen.

16.2.2 DEFINITIE **Substitutie**
Zij $y_1, \ldots, y_n \in \mathit{VAR}$ *en* $t_1, \ldots, t_n \in \mathit{TERM}$ $(n \geq 0)$*, zij* $y_i \neq t_i$*, en* $y_i \neq y_j$ *als* $i \neq j$ $(1 \leq i, j \leq n)$*. Dan wordt de verzameling van paren* (y_i, t_i) *met* $1 \leq i \leq n$*, notatie* $[y_1{:=}t_1, \ldots, y_n{:=}t_n]$*, een* substitutie *genoemd. Als alle*

termen t_i gesloten zijn, dan spreekt men van een grondsubstitutie. *Als $n = 0$, dan spreekt men van de* lege substitutie, *notatie ϵ.*

Het resultaat van het toepassen van een substitutie op een term of een formule zonder kwantoren noemt men een *instantie*. Men kan ook definiëren wat een instantie is van een formule *met* kwantoren. In dat geval moet rekening gehouden worden met het feit dat de variabelen in de te substitueren termen niet onbedoeld worden gebonden door kwantoren. Dit kan worden voorkomen door een geschikte herbenoeming van gebonden variabelen.

16.2.3 DEFINITIE **Instantie**
Zij $\theta = [y_1 := t_1, \ldots, y_n := t_n]$ een substitutie, en zij E een term of een formule zonder kwantoren. Zij verder $E\theta$ het resultaat van het simultaan vervangen van alle voorkomens van y_i in E door t_i $(1 \leq i \leq n)$. Dan noemt men $E\theta$ een instantie *van E. Als alle termen t_i gesloten zijn, dan noemt men $E\theta$ een* grondinstantie.

Uit bovenstaande definitie volgt dat $E\epsilon = E$ voor alle termen of formules zonder kwantoren E. Ook geldt dat $E\theta = E$ als geen der variabelen y_i in E voorkomt.

16.2.4 DEFINITIE **Compositie van substituties**
Zij $\theta = [v_1 := s_1, \ldots, v_m := s_m]$ en $\sigma = [w_1 := t_1, \ldots, w_n := t_n]$ substituties. Dan wordt de compositie *van θ en σ, notatie $\theta\sigma$, verkregen door uit de verzameling:*

$$[v_1 := s_1\sigma, \ldots, v_m := s_m\sigma, w_1 := t_1, \ldots, w_n := t_n],$$

de volgende paren te verwijderen:

1. *paren van de vorm $v_i := s_i\sigma$, indien $s_i\sigma = v_i$, en*
2. *paren van de vorm $w_j := t_j$, indien $w_j \in \{v_1, \ldots, v_m\}$.*

De bovenstaande definitie impliceert dat $\theta\sigma$ een substitutie is, als θ en σ dat zijn. Verder is de compositie zo gedefinieerd, dat de volgende gelijkheid geldt:

$$E(\theta\sigma) \;=\; (E\theta)\sigma.$$

Ook kan men bewijzen dat, als λ, θ en σ substituties zijn:

$$(\lambda\theta)\sigma \;=\; \lambda(\theta\sigma),$$
$$\epsilon\theta \;=\; \theta\epsilon \;=\; \theta.$$

Met andere woorden, de compositie van substituties is *associatief*, en ϵ is een zogenaamd linker en rechter *eenheidselement*. Het bewijs van de bovenstaande eigenschappen wordt aan de lezer overgelaten.

16.2.5 Voorbeeld Compositie van substituties.
Zij $\theta = [x{:=}f(y), y{:=}a, z{:=}y]$ en $\sigma = [x{:=}g(a), y{:=}z, w{:=}b]$. De compositie $\theta\sigma$ is gelijk aan de substitutie:

$$[x{:=}f(z), y{:=}a, w{:=}b],$$

terwijl de compositie $\sigma\theta$ gelijk is aan:

$$[x{:=}g(a), w{:=}b, z{:=}y].$$

Hieruit volgt dat $\theta\sigma \neq \sigma\theta$. Dit betekent dat de compositie van substituties niet commutatief is. ■

In voorbeeld 16.2.1 hebben we gezien dat we bij het resolveren van disjuncties die variabelen bevatten, vaak eerst een substitutie moeten toepassen om complementaire literalen te verkrijgen. Een substitutie die twee literalen aan elkaar gelijk maakt, noemt men een *unificator*, en het zoeken naar een dergelijke substitutie *unificatie*.

16.2.6 Definitie **Unificator en mgu**
Zij $V = \{E_1, \ldots, E_n\}$ $(n \geq 2)$ *een verzameling van termen of formules zonder kwantoren.*

1. *Een substitutie* θ *is een* unificator *van* V *als* $E_1\theta = \ldots = E_n\theta$. *Een verzameling* V *wordt* unificeerbaar *genoemd als er een unificator voor* V *bestaat.*
2. *Een substitutie* θ *wordt de* meest algemene unificator *of* mgu *(Engels:* most general unifier*) van* V *genoemd als er voor iedere unificator* σ *van* V *een substitutie* λ *bestaat zodanig dat* $\sigma = \theta\lambda$.

16.2.7 Voorbeeld Unificators en mgu's.
Zij $V = \{P(y, f(x, b)),\ P(a, z)\}$. De substituties $\theta = [y{:=}a, z{:=}f(x, b)]$ en $\sigma = [x{:=}a, y{:=}a, z{:=}f(a, b)]$ zijn unificators van V. De unificator θ is de mgu van V. Merk op dat $\sigma = \theta[x{:=}a]$. ■

16.3 Het unificatie-algoritme

Er bestaat een algoritme om de meest algemene unificator van een verzameling termen of formules zonder kwantoren te vinden. Het bestaat eruit dat de symbolen in de termen of formules symbool voor symbool worden vergeleken tot er een verschil is gevonden. Dit verschil wordt door een substitutie teniet gedaan indien mogelijk, waarna weer wordt verder gegaan met vergelijken.

16.3.1 DEFINITIE **Verschilverzameling**
Zij $V = \{E_1, \ldots, E_n\}$ $(n \geq 2)$ *een verzameling van termen of formules zonder kwantoren, en zij* j *het grootste natuurlijke getal zodanig dat de eerste* j *symbolen van* $E_1, \ldots, E_n$ *gelijk zijn. Zij verder* s_i *de subterm of subformule van* E_i *zodanig dat het eerste symbool van* s_i *gelijk is aan het symbool op positie* $j+1$ *van* E_i $(1 \leq i \leq n)$*. Dan noemt men* $\{s_i \mid 1 \leq i \leq n\}$ *de* verschilverzameling *van* V*.*

16.3.2 VOORBEELD Verschilverzamelingen.

1. Voor de verzameling $\{P(x, f(a, x)),\ P(x, b),\ P(x, g(x, y))\}$ is de verschilverzameling gelijk aan $\{f(a, x),\ b,\ g(x, y)\}$.
2. Voor $W = \{P(x),\ Q(y)\}$ is de verschilverzameling gelijk aan W. ■

Met behulp van het volgende, zogenaamde *unificatie-algoritme* kan de mgu van een gegeven eindige, niet-lege verzameling V van termen of formules zonder kwantoren worden bepaald. Het algoritme is afkomstig van Robinson (1965). De invoer is de verzameling V, en de uitvoer is de mgu σ_V van V, of de mededeling dat V niet unificeerbaar is.

Unificatie-algoritme

Stap 1 $k := 0$ en $\sigma_k := \epsilon$.

Stap 2 Als $V\sigma_k$ precies één element bevat, dan **Stop**: $\sigma_V = \sigma_k$ is een mgu van V.

Stap 3 Als de verschilverzameling van $V\sigma_k$ een variabele v_k bevat en een term t_k waarin v_k niet voorkomt, dan $\sigma_{k+1} := \sigma_k[v_k := t_k]$, $k := k+1$ en ga naar **Stap 2**; anders **Stop**: V is niet unificeerbaar.

Merk op dat het algoritme niet deterministisch is. In stap 3 wordt een willekeurige keuze gemaakt voor een variabele v_k en een term t_k in de verschilverzameling van $V\sigma_k$. Het algoritme kan op eenvoudige wijze deterministisch worden gemaakt door de elementen in de verschilverzamelingen te ordenen, en dan voor v_k de eerste variabele in de verschilverzameling van $V\sigma_k$ te nemen, en voor t_k de eerste term waarin v_k niet voorkomt.

16.3.3 VOORBEELD Bepaling van mgu.
We zullen laten zien hoe het unificatie-algoritme werkt aan de hand van een voorbeeld. Zij $V = \{P(x, y, f(x, g(x, a))),\ P(a, h(z), f(a, z))\}$. De mgu van V wordt als volgt door het algoritme bepaald:

1. Stap 1: $\sigma_0 := \epsilon$.

2. Stap 2: $V\sigma_0 = V$ bevat meer dan één element.

3. Stap 3: De verschilverzameling van $V\sigma_0$ is $\{x,\ a\}$. Neem $v_0 = x$ en $t_0 = a$, zodat:

   $$\sigma_1 := \epsilon[x{:=}a] = [x{:=}a].$$

4. Stap 2: $V\sigma_1 = \{P(a, y, f(a, g(a, a))),\ P(a, h(z), f(a, z))\}$ bevat meer dan één element.

5. Stap 3: De verschilverzameling van $V\sigma_1$ is $\{y,\ h(z)\}$. Neem $v_1 = y$ en $t_1 = h(z)$, zodat:

   $$\sigma_2 := \sigma_1[y{:=}h(z)] = [x{:=}a, y{:=}h(z)].$$

6. Stap 2: $V\sigma_2 = \{P(a, h(z), f(a, g(a, a))),\ P(a, h(z), f(a, z))\}$ bevat meer dan één element.

7. Stap 3: De verschilverzameling van $V\sigma_2$ is $\{z,\ g(a, a)\}$. Neem $v_2 = z$ en $t_2 = g(a, a)$, zodat:

   $$\sigma_3 = \sigma_2[z{:=}g(a, a)] = [x{:=}a, y{:=}h(g(a, a)), z{:=}g(a, a)].$$

8. Stap 2: $V\sigma_3 = \{P(a, h(g(a, a)), f(a, g(a, a)))\}$ bevat precies één element. Stel $\sigma_V = \sigma_3$ en stop.

De mgu van V is dus gelijk aan $\sigma_V = [x{:=}a, y{:=}h(g(a, a)), z{:=}g(a, a)]$. ∎

Uit de omschrijving blijkt dat het algoritme voor iedere eindige invoerverzameling V termineert. Was dat namelijk niet het geval, dan zou er een oneindige rij $V\sigma_0, V\sigma_1, V\sigma_2, \ldots$ worden gegenereerd met de eigenschap dat iedere verzameling uit deze rij één variabele minder bevat dan zijn voorganger. Immers, voor iedere $k \in \mathbb{N}$ geldt dat v_k voorkomt in $V\sigma_k$, maar niet in $V\sigma_{k+1}$. Omdat de verzameling V slechts een eindig aantal verschillende variabelen bevat, kan een dergelijke oneindige rij niet worden voortgebracht.

16.3.4 STELLING **Correctheid unificatie-algoritme**
Zij V een eindige, niet-lege verzameling van termen of formules zonder kwantoren. Als V unificeerbaar is, dan stopt het unificatie-algoritme in stap 2 en is σ_V de mgu van V.

BEWIJS *Stel dat V unificeerbaar is, en dat θ een unificator is van V. Stel verder dat stap 3 van het algoritme n keer wordt uitgevoerd wanneer het wordt toegepast op V. We laten zien dat het algoritme stopt in stap 2, en dat er voor iedere k waarvoor $0 \leq k \leq n$ een substitutie λ_k bestaat zodanig dat:*

$$\theta = \sigma_k \lambda_k .$$

Dit laatste bewijzen we met volledige inductie naar k. Als het bovenstaande is bewezen, volgt dat na terminatie van het algoritme $\sigma_V = \sigma_n$ een mgu van V is.

1. *Basisstap: $k = 0$.*

   Aangezien $\sigma_0 = \epsilon$, kan $\lambda_0 = \theta$ worden genomen. Omdat stap 3 nog geen enkele keer is uitgevoerd, kan het algoritme niet zijn gestopt in stap 3. Als $V\sigma_0$ precies één element bevat, dat stopt het algoritme in stap 2, en is σ_0 een mgu van V.

2. *Inductiestap: $k > 0$.*

   De inductiehypothese is dat $\theta = \sigma_i \lambda_i$ voor alle $0 \leq i \leq k$.

   Als $k = n$ en $V\sigma_k$ precies één element bevat, dan stopt het algoritme in stap 2, en is wegens de inductiehypothese $\sigma_V = \sigma_k$ een mgu van V.

   Als $k < n$, of $V\sigma_k$ bevat meer dan één element, dan wordt stap 3 van het algoritme uitgevoerd. Zij W_k de verschilverzameling van V_k. Omdat θ een unificator is van V, is volgens de inductiehypothese $\sigma_k \lambda_k$ ook een unificator van V. Omdat W_k de verschilverzameling is van $V\sigma_k$, moet λ_k een unificator zijn van W_k. Dit betekent dat W_k een variabele v_k moet bevatten. Zij nu t_k een ander element van W_k dat verschillend is van v_k. Omdat λ_k een unificator is van W_k moet er gelden:

   $$v_k \lambda_k = t_k \lambda_k .$$

   Omdat v_k en t_k verschillende elementen zijn van W_k, volgt hieruit dat het onmogelijk is dat v_k voorkomt in t_k. Het bovenstaande impliceert dat het algoritme niet in stap 3 termineert, maar teruggaat naar stap 2. Alvorens dit te doen krijgt σ_{k+1} de waarde $\sigma_k[v_k := t_k]$. Zij nu:

   $$\lambda_{k+1} = \lambda_k - [v_k := t_k \lambda_k],$$

   dan kunnen we afleiden:

   $$\begin{aligned} \lambda_k &= [v_k := t_k \lambda_k] \cup \lambda_{k+1}, \\ &= [v_k := t_k \lambda_{k+1}] \cup \lambda_{k+1}, \\ &= [v_k := t_k] \lambda_{k+1} . \end{aligned}$$

   Hieruit volgt tenslotte:

   $$\begin{aligned} \theta &= \sigma_k \lambda_k, \\ &= \sigma_k [v_k := t_k] \lambda_{k+1}, \\ &= \sigma_{k+1} \lambda_{k+1} . \end{aligned}$$

Hiermede is de stelling bewezen. ■

16.4 De resolutieregel

In paragraaf 16.1 hebben we gezien, dat we in een zin in Skolemnormaalvorm $\{D_1, \ldots, D_n\}$ de variabelen in verschillende disjuncties verschillend kunnen kiezen. We zullen er in het vervolg vanuit gaan, dat dit inderdaad is gebeurd.

Voordat we de resolutieregel definiëren, introduceren we een aantal begrippen die we daarbij nodig hebben.

16.4.1 DEFINITIE **Factor**
Als twee of meer literalen in een disjunctie D een mgu σ bezitten, dan noemt men $D\sigma$ een factor *van D. Als $D\sigma$ precies één literaal bevat, dan noemt men $D\sigma$ een* eenheidsfactor.

16.4.2 DEFINITIE **Complement van een literaal**
Zij L een literaal en A een atoom. Als $L = A$, dan is het complement *van L, notatie $\overline{L}$, gedefinieerd als $\neg A$; als $L = \neg A$, dan is $\overline{L} = A$.*

16.4.3 DEFINITIE **Binaire resolvent**
Zij D_1 en D_2 disjuncties die geen gemeenschappelijke variabelen bezitten, en zij L_i een literaal in D_i $(i = 1, 2)$. Als L_1 en $\overline{L_2}$ een mgu σ bezitten, dan noemt men de disjunctie:

$$(D_1\sigma - L_1\sigma) \cup (D_2\sigma - L_2\sigma)$$

een binaire resolvent *van D_1 en D_2. Als $D_1 = \{L_1\}$ en $D_2 = \{L_2\}$, dan is de binaire resolvent gelijk aan de lege disjunctie $\Box$.*

16.4.4 VOORBEELD Factor en binaire resolvent.

1. Zij $D = P(x, a) \vee P(f(y), z) \vee R(z)$. De substitutie $\sigma = [x{:=}f(y), z{:=}a]$ is een mgu van de eerste twee literalen. Dit betekent dat $D\sigma = P(f(y), a) \vee R(a)$ een factor is van D.
2. Zij $D_1 = P(x, f(x)) \vee Q(x)$ en $D_2 = \neg P(a, y) \vee R(z)$. Een mgu van $P(x, f(x))$ en $P(a, y)$ is $[x{:=}a, y{:=}f(a)]$. Hieruit volgt dat $Q(a) \vee R(z)$ een binaire resolvent van D_1 en D_2 is. ∎

16.4.5 DEFINITIE **Resolvent en resolutieregel** *Zij D_1 en D_2 disjuncties. Dan is een* resolvent *van D_1 en D_2:*

- *òf een binaire resolvent van D_1 en D_2,*
- *òf een binaire resolvent van D_1 en een factor van D_2,*
- *òf een binaire resolvent van een factor van D_1 en D_2,*
- *òf een binaire resolvent van een factor van D_1 en een factor van D_2.*

Het voorschrift om een resolvent van twee disjuncties te bepalen wordt de resolutieregel *genoemd.*

Net als voor de propositielogica kunnen we definiëren wat we onder een *resolutie-afleiding* verstaan.

16.4.6 DEFINITIE **Resolutie-afleiding**
Zij F een zin in Skolemnormaalvorm, zodat F een eindige verzameling van disjuncties is, en zij D een disjunctie. Een afleiding *van D uit F is een rij disjuncties:*

$$R_1, R_2, \ldots, R_n = D,$$

waarin iedere R_i $(1 \leq i \leq n)$ een disjunctie uit F is, òf een resolvent van twee disjuncties R_j en R_k waarvoor $1 \leq j, k \leq i$. Als er een resolutie-afleiding van D uit F bestaat, dan zegt men dat D afleidbaar *is uit F, notatie $F \vdash_{\mathcal{R}} D$. Tenslotte noemt men n de* lengte *van de afleiding.*

We zijn, wederom net als in de propositielogica, vooral geïnteresseerd in afleidingen van $F \vdash_{\mathcal{R}} \Box$. In dat geval is F onvervulbaar zoals we in paragraaf 16.6 zullen zien.

16.5 De resolutiemethode

Er is een simpel algoritme om na te gaan of een zin F in Skolemnormaalvorm onvervulbaar is. Namelijk, genereer net zolang resolventen totdat de lege disjunctie $\Box$ wordt voortgebracht. In dat geval luidt de conclusie dat F onvervulbaar is. Dit is het zelfde algoritme als voor de propositielogica, dat we hier herhalen.

Resolutie-algoritme

Stap 1 Genereer, indien mogelijk, een nieuwe resolvent R van disjuncties uit F; anders **Stop**: F is vervulbaar.

Stap 2 Als $R = \Box$, dan **Stop**: F is onvervulbaar; anders $F := F \cup \{R\}$ en ga naar **Stap 1**.

Dit algoritme is nu een semi-beslissingsprocedure. Het algoritme behoeft namelijk niet voor alle invoer te stoppen. Als de invoer onvervulbaar is, dan zal het algoritme ooit stoppen. Als de invoer echter vervulbaar is, dan zal het algoritme in sommige gevallen steeds weer nieuwe resolventen genereren, door steeds andere substituties te proberen.

16.5.1 VOORBEELD Resolutie-afleiding.
Bewijs $P_1, P_2 \vdash C$ met:

$$\begin{aligned} P_1 &= \forall x[M(x) \rightarrow S(x)], \\ P_2 &= M(a), \\ C &= \exists x S(x). \end{aligned}$$

Hiertoe brengen we de formule $P_1 \wedge P_2 \wedge \neg C$ eerst in Skolemnormaalvorm. Dit levert:

$$F = \{\neg M(x) \vee S(x),\ M(a),\ \neg S(y)\}.$$

De afleiding van $\Box$ uit F verloopt als volgt:

1. $\neg M(x) \vee S(x)$ (element F)
2. $M(a)$ (element F)
3. $\neg S(y)$ (element F)
4. $S(a)$ (resolvent 1,2 met mgu $[x{:=}a]$)
5. $\Box$ (resolvent 3,4 met mgu $[y{:=}a]$) ■

16.5.2 Voorbeeld Resolutie-afleiding.
Bewijs $P \vdash C$ met:

$$\begin{aligned} P &= \forall x[P(x) \to D(x)], \\ C &= \forall x[\exists y(P(y) \wedge S(x,y)) \to \exists y(D(y) \wedge S(x,y))]. \end{aligned}$$

(Dit is een formalisering van de redenering: *Als paarden dieren zijn, dan zijn paardestaarten dierestaarten.*)

We laten het aan de lezer over om te verifiëren dat de onderstaande verzameling F een Skolemnormaalvorm van de zin $P \wedge \neg C$ is:

$$F = \{\neg P(x) \vee D(x),\ P(b),\ S(a,b),\ \neg D(y) \vee \neg S(a,y)\}.$$

De afleiding van $\Box$ uit F verloopt als volgt:

1. $\neg P(x) \vee D(x)$ (element F)
2. $P(b)$ (element F)
3. $S(a,b)$ (element F)
4. $\neg D(y) \vee \neg S(a,y)$ (element F)
5. $D(b)$ (resolvent 1,2 met mgu $[x{:=}b]$)
6. $\neg S(a,b)$ (resolvent 4,5 met mgu $[y{:=}b]$)
7. $\Box$ (resolvent 3,6 met mgu ϵ) ■

16.5.3 Voorbeeld Resolutie-afleiding.
We zullen bewijzen dat een driehoek met gelijke basishoeken gelijkbenig is. Het idee van het bewijs is dat voor een driehoek $\triangle ABC$ waarvan $\angle CAB = \angle CBA$, geldt dat $\triangle CAB \cong \triangle CBA$ (het symbool $\cong$ staat voor *is congruent met*). Hieruit volgt dan dat $AC = BC$.

Ten behoeve van het bewijs introduceren we vier predicaten:

$$\begin{aligned} T(x,y,z) &: \; xyz \text{ is een driehoek}, \\ C(u,v,w,x,y,z) &: \; \triangle uvw \cong \triangle xyz, \\ L(u,v,x,y) &: \; uv = xy, \\ H(u,v,w,x,y,z) &: \; \angle uvw = \angle xyz. \end{aligned}$$

De predicaten T, C, L en H voldoen aan de volgende axioma's die als disjuncties worden gegeven. Hierbij hebben we niet steeds verse variabelen gebruikt, vanwege het grote aantal verschillende variabelen dat nodig zou zijn. In dit geval levert dat geen problemen.

1. $\neg T(x,y,z) \vee T(y,z,x)$.
2. $\neg T(x,y,z) \vee T(y,x,z)$.

3. $L(x,y,x,y)$.
4. $\neg L(u,v,x,y) \vee L(x,y,u,v)$.
5. $\neg L(u,v,x,y) \vee L(v,u,x,y)$.

6. $H(x,y,z,x,y,z)$.
7. $\neg H(u,v,w,x,y,z) \vee H(w,v,u,x,y,z)$.
8. $\neg H(u,v,w,x,y,z) \vee H(x,y,z,u,v,w)$.

9. $\neg C(u,v,w,x,y,z) \vee H(u,v,w,x,y,z)$.
10. $\neg C(u,v,w,x,y,z) \vee H(v,w,u,y,z,x)$.
11. $\neg C(u,v,w,x,y,z) \vee H(w,u,v,z,x,y)$.

12. $\neg C(u,v,w,x,y,z) \vee L(u,v,x,y)$.
13. $\neg C(u,v,w,x,y,z) \vee L(v,w,y,z)$.
14. $\neg C(u,v,w,x,y,z) \vee L(u,w,x,z)$.

15. $\neg T(u,v,w) \vee \neg T(x,y,z) \vee \neg L(u,v,x,y)$
    $\vee \neg H(w,u,v,z,x,y) \vee \neg H(w,v,u,z,y,x) \vee C(u,v,w,x,y,z)$.

Niet al deze axioma's worden gebruikt in de afleiding. Ook is het niet zo dat de bovenstaande axioma's volledig zijn. Axioma 15 brengt tot uitdrukking dat $\triangle ABC \cong \triangle DEF$ als $AB = DE$, $\angle CAB = \angle FDE$ en $\angle CBA = \angle FED$ (het zogenaamde HZH-geval). Uiteraard zijn er nog andere axioma's nodig om alle gevallen van congruentie te dekken. Verder ontbreken er ook axioma's voor de transitiviteit van gelijkheid van hoeken en lijnstukken.

We moeten uitgaande van het gegeven dat in $\triangle ABC$ de basishoeken $\angle CAB$ en $\angle CBA$ gelijk zijn, bewijzen dat $\triangle ABC$ gelijkbenig is ofwel $AC = BC$. Dit wordt als volgt opgeschreven.

16. $T(a, b, c)$.
17. $H(c, a, b, c, b, a)$.
18. $\neg L(a, c, b, c)$.

De afleiding van $\square$ uit de disjuncties 1 t/m 18 verloopt als volgt:

19. $T(b, a, c)$,
    resolvent van 2 en 16 met mgu $[x{:=}a, y{:=}b, z{:=}c]$.
20. $\neg T(x, y, z) \vee \neg L(b, a, x, y) \vee \neg H(c, b, a, z, x, y) \vee \neg H(c, a, b, z, y, x)$
    $\vee C(b, a, c, x, y, z)$,
    resolvent van 15 en 19 met mgu $[u{:=}b, v{:=}a, w{:=}c]$.
21. $\neg L(b, a, a, b) \vee \neg H(c, b, a, c, a, b) \vee \neg H(c, a, b, c, b, a)$
    $\vee C(b, a, c, a, b, c)$,
    resolvent van 16 en 20 met mgu $[x{:=}a, y{:=}b, z{:=}c]$.
22. $L(y, x, x, y)$,
    resolvent van 3 en 5 met mgu $[u{:=}x, v{:=}y]$.
23. $\neg H(c, b, a, c, a, b) \vee \neg H(c, a, b, c, b, a) \vee C(b, a, c, a, b, c)$,
    resolvent van 21 en 22 met mgu $[x{:=}a, y{:=}b]$.
24. $H(c, b, a, c, a, b)$,
    resolvent van 8 en 17 met mgu
    $[x{:=}c, y{:=}b, z{:=}a, u{:=}c, v{:=}a, w{:=}b]$.
25. $\neg H(c, a, b, c, b, a) \vee C(b, a, c, a, b, c)$,
    resolvent van 23 en 24 met mgu ϵ.
26. $C(c, a, b, a, b, c)$,
    resolvent van 17 en 25 met mgu ϵ.
27. $L(a, c, b, c)$,
    resolvent van 13 en 26 met mgu
    $[x{:=}a, y{:=}b, z{:=}c, u{:=}c, v{:=}a, w{:=}b]$.
28. $\square$,
    resolvent van 18 en 27 met mgu ϵ.

Hiermee is de stelling bewezen. $\blacksquare$

16.6 Volledigheid van de resolutiemethode

De volgende stelling maakt gebruik van de *universele afsluiting* $U(F)$ van een formule F. In hoofdstuk 9 hebben we gezien dat $U(F)$ uit F verkregen wordt door voor iedere vrije variabele in F een universele kwantor voor die variabele vóór F te plaatsen.

16.6.1 STELLING **Correctheid resolutieregel**
Als R een resolvent is van twee disjuncties D_1 en D_2, dan geldt:

$$U(D_1), U(D_2) \models U(R).$$

BEWIJS Wordt aan de lezer overgelaten. ∎

16.6.2 STELLING **Correctheid resolutie**
Zij $F \in$ ZIN een zin in Skolemnormaalvorm, en D een disjunctie. Dan geldt:

$$F \vdash_{\mathcal{R}} D \quad \Rightarrow \quad F \models U(D).$$

BEWIJS *Door middel van inductie over de lengte van de afleiding met gebruikmaking van stelling 16.6.1.* ∎

Merk op dat in de bovenstaande stelling de formule F zoals genoemd in $F \vdash_{\mathcal{R}} D$, wordt voorgesteld als een niet-lege, eindige verzameling van disjuncties, terwijl deze in $F \models U(D)$ wordt gedacht als een zin zonder vrije variabelen.

De volgende stelling staat in de Engelstalige literatuur bekend als het *lifting lemma*. De inhoud ervan is dat een binaire resolvent R' van grondinstanties van disjuncties D_1 en D_2 zelf een instantie is van een resolvent R van D_1 en D_2. Anders gezegd, R' kan worden 'opgetild' tot R.

16.6.3 STELLING **Lifting lemma**
Zij D_1 en D_2 disjuncties die geen gemeenschappelijke variabelen bezitten. Als D_1' en D_2' grondinstanties zijn van D_1 en D_2, en R' een resolvent is van D_1' en D_2', dan bestaat er een resolvent R van D_1 en D_2 zodanig dat R' een instantie is van R.

BEWIJS Omdat D_1' een instantie is van D_1 en D_2' van D_2, en omdat D_1 en D_2 verschillende variabelen bevatten, bestaat er een substitutie θ zodanig dat $D_1' = D_1\theta$ en $D_2' = D_2\theta$. Omdat R' een resolvent is van D_1' en D_2', bevat D_1' een literaal L_1 en D_2' een literaal L_2 zodanig dat $L_2 = \overline{L_1}$. Zij V_i $(i = 1, 2)$ de verzameling van literalen in D_i die door θ op L_i worden afgebeeld.

Zij λ_i een mgu van V_i $(i = 1, 2)$, en zij $\lambda = \lambda_1 \cup \lambda_2$. Omdat de variabelen in V_1 en V_2 verschillend zijn, is λ een mgu van V_1 en V_2. Hieruit volgt dat L_1 een instantie is van $V_1\lambda$, en L_2 van $V_2\lambda$, zodat $V_1\lambda$ en $\overline{V_2\lambda}$ unificeerbaar zijn. Zij nu σ een mgu van $V_1\lambda$ en $\overline{V_2\lambda}$, dan kan men bewijzen (zie opgave 4) dat $\lambda\sigma$ een mgu is voor de verzameling bestaande uit de literalen in V_1 en de complementen van de literalen in V_2.

Nu geldt dat:

$$R \;=\; (D_1\lambda\sigma - V_1\lambda\sigma) \cup (D_2\lambda\sigma - V_2\lambda\sigma)$$

een resolvent is van D_1 en D_2. Voor de resolvent R' geldt:

$$R' \;=\; (D_1\theta - V_1\theta) \cup (D_2\theta - V_2\theta).$$

Omdat θ een unificator is van de literalen in V_1 en de complementen van de literalen in V_2, en omdat $\lambda\sigma$ hiervoor een mgu is, bestaat er een substitutie δ zodanig dat $\theta = \lambda\sigma\delta$. Hieruit en uit de gevonden uitdrukkingen voor R en R' volgt dat R' een instantie is van R. ■

16.6.4 STELLING **Volledigheid resolutie** *Zij $F \in$ ZIN een zin in Skolemnormaalvorm. Dan geldt:*

$$F \vdash_{\mathcal{R}} \Box \qquad \Leftrightarrow \qquad F \textit{ is onvervulbaar.}$$

BEWIJS

($\Rightarrow$) *Uit het ongerijmde. Stel dat $F \vdash_{\mathcal{R}} \Box$ en dat F vervulbaar is. Dan bestaat er een afleiding*

$$R_1, R_2, \ldots, R_n = \Box.$$

Uit de definitie van een afleiding volgt dat $F \vdash_{\mathcal{R}} R_i$ voor $(1 \leq i \leq n)$. Toepassing van stelling 16.6.2 levert dat $F \models U(R_i)$ voor $(1 \leq i \leq n)$. Als F vervulbaar is, dan bestaat er een structuur $\mathcal{A}$ zodanig dat $\mathcal{A} \models F$. In combinatie met $F \models U(R_i)$ volgt hieruit dat dat $\mathcal{A} \models U(R_i)$ voor alle $(1 \leq i \leq n)$. Omdat $F \vdash_{\mathcal{R}} \Box$, zijn er disjuncties R_j en R_k, en een mgu σ zodanig dat $R_j\sigma = \overline{R_k\sigma}$ voor $1 \leq j < k < n$. Maar dan moet gelden $\mathcal{A} \models U(R_j\sigma)$ en $\mathcal{A} \models U(R_k\sigma)$, wat onmogelijk is.

($\Leftarrow$) *Stel dat F onvervulbaar is. Uit de stelling van Herbrand volgt, dat er een eindige verzameling E van grondinstanties van disjuncties uit F bestaat die onvervulbaar is. Uit de volledigheid van de resolutie voor de propositielogica volgt nu dat er een afleiding:*

$$R_1, R_2, \ldots, R_n = \Box$$

van $\Box$ uit E bestaat. Uit het lifting lemma volgt dat deze afleiding kan worden 'opgetild' tot een afleiding

$$S_1, S_2, \ldots, S_n = \Box$$

van $\Box$ uit F, waarbij R_i een grondinstantie is van S_i $(1 \leq i \leq n)$. ■

16.7 Opgaven

1. Breng de volgende zinnen in Skolemnormaalvorm:
   (a) $\neg[\forall x P(x) \rightarrow \exists y \forall z Q(y,z)]$.
   (b) $\forall x[\forall y P(x,y) \rightarrow \forall y \neg \forall z Q(y,z)]$.

2. Bepaal, indien mogelijk, een mgu voor elk van de volgende verzamelingen:
   (a) $\{P(f(x)),\ P(f(y))\}$,
   (b) $\{P(f(x,a)),\ Q(f(b,y))\}$,
   (c) $\{P(x, f(x,a,y)),\ P(y, f(b,z,z))\}$,

   (d) $\{P(x, f(x, y), g(x, b, z)),\ P(h(y, a), f(h(y, z), b), g(w, y, a))\}$.

3. Bewijs met behulp van resolutie dat:
   (a) $\forall x[A(x) \rightarrow B(x)] \vdash \exists x A(x) \rightarrow \exists x B(x)$.
   (b) $\forall x \exists y R(x, y), \forall x \forall y[R(x, y) \rightarrow S(x, y)] \vdash \forall x \exists y S(x, y)$.
   (c) $\exists x[A(x) \wedge \forall y(B(y) \rightarrow R(y, x))] \vdash \forall x[B(x) \rightarrow \exists y(A(y) \wedge R(x, y))]$.
4. Bewijs stelling 16.6.1.
5. Bewijs de bewering uit het bewijs van het lifting lemma dat $\lambda\sigma$ een mgu is voor de verzameling bestaande uit de literalen in V_1 en de complementen van de literalen in V_2.

Hoofdstuk 17

Logisch Programmeren en Prolog

Iets bewijzen en iets berekenen zijn natuurlijk geheel verschillende dingen. Het leveren van een bewijs betekent zoveel als het aangeven van een rij redeneerstappen die van de aannamen op verantwoorde wijze leidt tot de gewenste conclusie. Het is vaak niet eenvoudig om zo'n rij te vinden; heel in het algemeen valt er eigenlijk alleen van te zeggen dat het een gecompliceerd zoekproces is. Het maken van een berekening daarentegen is het stap voor stap uitvoeren van een rekenvoorschrift. Er is volstrekte duidelijkheid over wat de volgende stap in de berekening is.

Toch is er een overeenkomst tussen het maken van een bewijs en het uitvoeren van een berekening, in de zin dat de opeenvolgende stappen van een bewijs overeenkomen met de opeenvolgende stappen van een berekening. Om het uitvoeren van een berekening en het vinden van een bewijs dichter bij elkaar te brengen, moeten we de mate van onbepaaldheid in het proces van het vinden van de volgende stap in een bewijs verminderen.

Wij lichten eerst de overeenkomst tussen rekenen en bewijzen toe aan de hand van enkele voorbeelden. Vervolgens bespreken we op welke wijze het construeren van een bewijs zodanig efficiënt kan worden gemaakt, dat het verantwoord is om te spreken over een berekening in plaats van over een bewijs. De besproken methode is de methode die ook toegepast wordt bij de programmeertaal Prolog (**Pro**grammeren in **log**ica).

Het idee dat logica kan worden gebruikt om te programmeren, met andere woorden, om (efficiënt) executeerbare specificaties te maken —of bij Prolog nog gesproken mag worden over specificaties is overigens de vraag—, is een ontwikkeling begonnen eind zestiger, begin zeventiger jaren. Sinds die tijd is hieraan wereldwijd onderzoek gedaan. De belangrijkste geestelijke vaders zijn Robert A. Kowalski en Alain Colmerauer.

De indeling van dit hoofdstuk is als volgt. In paragraaf 17.1 werken we het idee uit dat bewijzen een vorm van berekenen is. Daarna bespreken we in paragraaf 17.2 een aantal resolutiestrategieën om het zoeken naar een resolutiebewijs te versnellen. Vervolgens worden in paragraaf 17.3 de zogenaamde *Horn-clauses* geïntroduceerd, een speciaal soort disjuncties waarop Prolog is gebaseerd. En tenslotte behandelen we in paragraaf 17.4 *SLD-resolutie*, de meest voorkomende resolutiestrategie waarvan Prolog-implementaties gebruik maken.

17.1 Bewijzen = berekenen?

We nemen als voorbeeld het rekenen met natuurlijke getallen. We gaan daarbij uit van voorbeeld 14.2.1 op bladzijde 203; dit betreft de theorie van de Peano-aritmetiek. Ons 'probleem' zal zijn het berekenen van een verschil; om precies te zijn het berekenen van $5 - 2$. Daartoe hebben we niet alle axioma's van eerder genoemd voorbeeld nodig, alleen **N3** en **N5**.

17.1.1 VOORBEELD De theorie N_+ van de optelling.
De conventies zijn die van voorbeeld 14.2.1. We gebruiken de volgende axioma's

N3. $\forall x[x + \mathbf{0} = x]$,

N5'. $\forall x \forall y \forall z[x + y = z \rightarrow x + \mathsf{s}(y) = \mathsf{s}(z)]$.

In de Peano-aritmetiek zoals gegeven in voorbeeld 14.2.1 gebruiken we nog andere eigenschappen van $=$ dan die gespecificeerd in de axioma's **N1** tot en met **N7** zoals reflexiviteit, symmetrie, transitiviteit en in het bijzonder ook de substitutie eigenschappen van de gelijkheid. Dat willen we hier vermijden, ten einde het concept 'bewijs' zo nauwkeurig mogelijk aan te laten sluiten bij het concept 'berekening'. Om dat te bereiken, moeten alle gebruikte eigenschappen goed zichtbaar zijn. Om deze reden gebruiken we axioma **N5'**, een variant van axioma **N5**. ∎

We moeten nu het probleem om $5 - 2$ te berekenen, formuleren als een bewering die in de theorie N_+ moet worden bewezen. Deze bewering is: $\exists x[x + 2 = 5]$. Aangezien we alleen de successor s en $\mathbf{0}$ in onze taal ter beschikking hebben, moeten we deze eigenschap als volgt formuleren:

$$\exists x[x + \mathsf{s}(\mathsf{s}(\mathbf{0})) = \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathbf{0})))))].$$

Deze stelling gaan we eerst bewijzen met de boommethode. Het voordeel daarvan is, dat bewijzen dat $\neg\exists x[x + \mathsf{s}(\mathsf{s}(\mathbf{0})) = \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathbf{0})))))]$ inconsistent is met **N1** en **N5'**, tevens een waarde oplevert die we moeten invullen voor x om te zien dat onze te bewijzen stelling inderdaad waar is. Dat zal dus hopelijk $[x{:=}\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathbf{0})))]$ zijn. Deze substitutie heet een *antwoordsubstitutie*, een goed gekozen naam vanuit ons berekeningsstandpunt. Het 'vinden' van deze antwoordsubstitutie gaat bij de boommethode niet echt vanzelf. Later zullen we zien dat bij gebruik van de resolutiemethode het bepalen van de antwoordsubstitutie onderdeel van de weerleggingsprocedure is, zodat met meer geloofwaardigheid kan worden gesteld dat het proces deze antwoordsubstitutie 'oplevert'. Het bewijs dat:

$$\textbf{N3, N5'} \models \exists x[x + \mathrm{s}(\mathrm{s}(\mathbf{0})) = \mathrm{s}(\mathrm{s}(\mathrm{s}(\mathrm{s}(\mathrm{s}(\mathbf{0})))))]$$

met $x = \mathrm{s}(\mathrm{s}(\mathrm{s}(\mathbf{0})))$, staat in figuur 17.1. We gebruiken hierbij en in het vervolg de volgende notatieconventie:

$$\mathrm{s}^k(\mathbf{0}) \;=\; \underbrace{\mathrm{s}(\mathrm{s}(\ldots\mathrm{s}(}_{k \text{ voorkomens}}\mathbf{0})\ldots))).$$

Eveneens gebruiken we de notatie $s \neq t$ in plaats van $\neg(s = t)$.

De in de figuur weergegeven rij stappen is niet echt een berekening van het antwoord, maar eerder het controleren dat de beginsubstitutie $[x{:=}\mathrm{s}(\mathrm{s}(\mathrm{s}(\mathbf{0})))]$ inderdaad het antwoord is. We kunnen het meer als een berekening opvatten als het bepalen van deze antwoordsubstitutie wordt uitgesteld. Het is mogelijk om de boommethode zo in te richten dat met ongeïnstantieerde termen wordt gewerkt, die pas op het moment dat duidelijk wordt welke specifieke term moet worden gekozen, nader worden bepaald. In resolutiebewijzen is dit al ingebouwd, dus laat ons eerst eens zien hoe daarin ons probleem (bereken het verschil $5 - 2$) wordt behandeld.

Eerst schrijven we de formules **N3** en **N5'**, die we samen als een programma zullen beschouwen, als disjuncties:

N3. $x + \mathbf{0} = x$,

N5'. $x + \mathrm{s}(y) = \mathrm{s}(z) \vee x + y \neq z$.

De opdracht aan het programma is te laten zien dat:

$$x + \mathrm{s}(\mathrm{s}(\mathbf{0})) \neq \mathrm{s}(\mathrm{s}(\mathrm{s}(\mathrm{s}(\mathrm{s}(\mathbf{0})))))$$

inconsistent is met het programma en daarbij een substitutie te leveren waaruit dat ook blijkt. Het resolutiebewijs staat in figuur 17.2. Dit bewijs is inderdaad een doelgerichte weerlegging die als een berekening van het antwoord 3 kan worden beschouwd. Het rekenen in het unaire stelsel is natuurlijk erg onhandig, maar dat doet er nu niet toe. Iets soortgelijks kan worden opgeschreven bij een andere getalrepresentatie. Daarbij horen dan andere formules die de optelling definiëren en dus ook een andere berekening, maar de manier van berekenen blijft verder hetzelfde.

N3 $* \ \forall x[x \mathbf{+} \mathbf{0} = x]$ (hypothese)

N5' $* \ \forall x \forall y \forall z[x \mathbf{+} y = z \rightarrow x \mathbf{+} s(y) = s(z)]$ (hypothese)

1. $\neg \exists x[x \mathbf{+} \mathrm{s}^2(\mathbf{0}) = \mathrm{s}^5(\mathbf{0})]$ ($\neg$ conclusie)
2. $\forall x[x \mathbf{+} \mathrm{s}^2(\mathbf{0}) \neq \mathrm{s}^5(\mathbf{0})]$ ($\exists$-regel,1)
3. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathbf{0} = \mathrm{s}^3(\mathbf{0})$ ($\forall$-regel,**N3**)
4. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathbf{0} = \mathrm{s}^3(\mathbf{0}) \rightarrow \mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}(\mathbf{0}) = \mathrm{s}^4(\mathbf{0})$ ($3 * \forall$-regel,**N5'**)
5. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathbf{0} \neq \mathrm{s}^3(\mathbf{0}) \vee \mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}(\mathbf{0}) = \mathrm{s}^4(\mathbf{0})$ ($\rightarrow$-regel,4)

(5)

6. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathbf{0} \neq \mathrm{s}^3(\mathbf{0})$

X(3,6)

7. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}(\mathbf{0}) = \mathrm{s}^4(\mathbf{0})$
8. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}(\mathbf{0}) = \mathrm{s}^4(\mathbf{0}) \rightarrow$
   $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}^2(\mathbf{0}) = \mathrm{s}^5(\mathbf{0})$ ($3 * \forall$-regel,**N5'**)
9. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}(\mathbf{0}) \neq \mathrm{s}^4(\mathbf{0}) \vee$
   $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}^2(\mathbf{0}) = \mathrm{s}^5(\mathbf{0})$ ($\rightarrow$-regel,8)

(8)

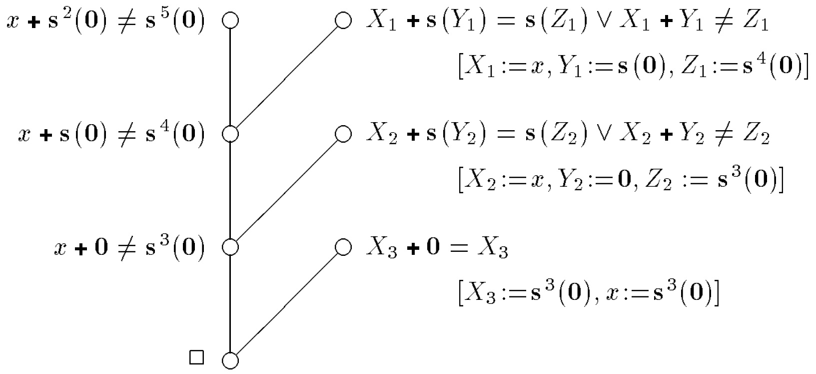
10. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}(\mathbf{0}) \neq \mathrm{s}^4(\mathbf{0})$

X(7,10)

11. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}^2(\mathbf{0}) = \mathrm{s}^5(\mathbf{0})$
12. $\mathrm{s}^3(\mathbf{0}) \mathbf{+} \mathrm{s}^2(\mathbf{0}) \neq \mathrm{s}^5(\mathbf{0})$ ($\forall$-regel,2)

X(11,12)

Figuur 17.1: **N3**, **N5'** $\models \exists x[x \mathbf{+} \mathrm{s}(\mathrm{s}(\mathbf{0})) = \mathrm{s}(\mathrm{s}(\mathrm{s}(\mathrm{s}(\mathrm{s}(\mathbf{0})))))]$.



Antwoordsubstitutie: $[x := \mathbf{s}(\mathbf{s}(\mathbf{s}(\mathbf{0})))]$.

Figuur 17.2: **N3**, **N5'**, $x + \mathbf{s}(\mathbf{s}(\mathbf{0})) \neq \mathbf{s}(\mathbf{s}(\mathbf{s}(\mathbf{s}(\mathbf{s}(\mathbf{0}))))) \vdash_{\mathcal{R}} \square$.

We kunnen het maken van het resolutiebewijs opvatten als het uitvoeren van een berekening van het programma **N3, N5'** op invoer $\neg\exists x[x + \mathrm{s}^2(\mathbf{0}) = \mathrm{s}^5(\mathbf{0})]$, of voor de vraag 'voor welke x is $x + \mathrm{s}^2(\mathbf{0}) = \mathrm{s}^5(\mathbf{0})$?'.

- De berekening begint met de aanroep $x + \mathrm{s}^2(\mathbf{0}) \neq \mathrm{s}^5(\mathbf{0})$.
- We passen de procedure, de disjunctie, $x + \mathrm{s}(y) = \mathrm{s}(z) \vee x + y \neq z$ (**N5'**) toe. We nemen daarbij voor iedere toepassing 'verse' copieën van de variabelen; nieuwe variabelen dus. Dit garandeert dat de resolutiemethode volledig is (zie de voorwaarde van het lifting lemma in hoofdstuk 16). Toepassen van de procedure leidt via de voor resolutie nodige unificatie tot bindingen van deze nieuwe variabelen.

  In ons geval leidt toepassen van de procedure tot een nieuwe aanroep $x + \mathrm{s}(\mathbf{0}) \neq \mathrm{s}^4(\mathbf{0})$. We passen dezelfde procedure toe —met weer nieuwe variabelen—, wat opnieuw leidt tot een volgende aanroep, namelijk $x + \mathbf{0} \neq \mathrm{s}^3(\mathbf{0})$.
- Nu passen we een andere procedure toe, namelijk **N3**. Dit leidt niet tot verdere aanroepen zodat de berekening termineert en wel met de antwoordsubstitutie die de compositie is van alle tijdens de unificaties geproduceerde substituties. In ons geval is dat $[x := \mathrm{s}^3(\mathbf{0})]$, als we alleen het deel uitschrijven dat betrekking heeft op de variabelen in de aanroep waar de berekening mee begon.

Het hier besproken voorbeeld kan, na onbelangrijke syntactische aanpassing, worden aangeboden aan een Prolog-systeem en functioneert dan probleemloos: het systeem berekent keurig netjes het verschil, als dat gevraagd wordt. De wijzigingen betreffen het gebruik van de speciale notatie van disjuncties in Prolog, maar laten het programma verder ongewijzigd. In Prolog wordt een disjunctie genoteerd door eerst alle positieve literalen gescheiden door komma's te schrijven, vervolgens het symbool `:-`, spreek uit 'als', en tenslotte alle negatieve literalen weer gescheiden door komma's. Aangezien zo al duidelijk is welke literalen negatief zijn, wordt het negatie symbool weg gelaten. Zo wordt bijvoorbeeld de disjunctie:

$$p(f(x)) \vee \neg q(f(x), y) \vee r(y, f(y)) \vee \neg s(x, g(x, y))$$

als volgt in Prolog-notatie weergegeven:

```
p(f(x)), r(y,f(y)) :- q(f(x),y)), s(x,g(x,y)).
```

Wat links van het teken `:-` staat heet de *head* en wat rechts ervan staat de *body* van de disjunctie. (We veroorloven ons hier een kleine vrijheid: in Prolog bestaat de head uit precies één literaal; dus de disjunctie hierboven is strikt genomen geen Prolog.) Het Prolog programma voor het berekenen van het verschil is als volgt:

```
:- op(500, yfx, ===).
X+0 === X.
X+s(Y) === s(Z) :- X+Y === Z.
```

De aanwijzing `:- op(500 ,xfy, ===)` maakt dat men `a === b` mag schrijven in plaats van `===(a,b)`. We gebruiken het symbool `===` in plaats van `=` omdat `=` in Prolog al een andere betekenis heeft.

Na inlezen van dit programma is Prolog gereed vragen te beantwoorden. Voorbeeld van een consultatie:

```
?- X+s(s(0)) === s(s(s(s(s(0))))).
   X = s(s(s(0)))
```

Hetzelfde programma kan trouwens ook gebruikt worden voor andere aanroepen, bijvoorbeeld $2+3=?$, uitgedrukt door $\mathbf{s}^2(\mathbf{0}) \mathbf{+} \mathbf{s}^3(\mathbf{0}) \neq x$. Nog een consultatie van het Prolog-systeem dat het rekenprogramma al heeft ingelezen:

```
?- s(s(0)) + s(s(s(0))) === X
   X = s(s(s(s(s(0)))))
```

We geven nog een voorbeeld van programmeren in logica, en wel het aaneenschakelen van lijsten. Een lijst stellen we voor als een term $h+t$, waarin $+$ een binaire operator, h het eerste element van de lijst en t de rest van de lijst is. De lege lijst representeren we als de constante *nil*. Dus de lijst $[a,b,a]$ heeft de term representatie $a+(b+(a+\mathit{nil}))$. De theorie van het aaneenschakelen van lijsten wordt bepaald door de volgende axiomatisering waarin $\mathit{schakel}(x,y,z)$ betekent dat z bestaat uit de lijst x met daar achter de lijst y:

$$\forall x[\mathit{schakel}(\mathit{nil},x,x)],$$
$$\forall w\forall x\forall y\forall z[\mathit{schakel}(x,y,z)\rightarrow \mathit{schakel}(w+x,y,w+z)].$$

Geschreven in disjunctieve vorm:

$$\mathit{schakel}(\mathit{nil},x,x),$$
$$\neg\mathit{schakel}(x,y,z)\vee \mathit{schakel}(w+x,y,w+z).$$

en in Prolog-vorm:

```
schakel(nil, X, X).
schakel(W+X, Y, W+Z) :- schakel(X, Y, Z).
```

Om de lijsten $a+(b+nil)$ en $c+(d+(a+nil))$ aaneen te schakelen, weerleggen we de formule:

$$\neg\exists x[schakel(a+(b+nil), c+(d+(a+nil)), x)].$$

Een voorbeeld van een consultatie van het Prolog programma:

```
?- schakel(a+(b+nil), c+(d+(a+nil)), X).
   X = a+(b+(c+(d+(a+nil))))

?- schakel(X,c+(d+(a+nil)), a+(b+(c+(d+(a+nil))))).
   X = a+(b+nil)
```

Zoals uit de tweede antwoordsubstitutie blijkt, beschrijven de gegeven disjuncties voor het predicaat *schakel* behalve het aaneenschakelen van twee lijsten, ook het splitsen van een lijst in twee delen.

In deze voorbeelden loopt alles erg mooi. De oorspronkelijke aanroep, en ook de aanroepen die tijdens de berekening ontstaan, bestaan allemaal uit slechts één literaal zodat duidelijk is welke literaal voor resolutie moet worden gebruikt. In het algemene geval is een aanroep een disjunctie, en moet nog worden bepaald welke literaal voor resolutie zal worden gebruikt; in termen van berekening, welke procedure zal worden aangeroepen. In het algemene geval moeten we de resolutieprocedure uitbreiden met een component die bepaalt op welke literaal zal worden geresolveerd.

Als duidelijk is welke literaal gebruikt zal worden, moet in het algemene geval nog een geschikte, bijpassende disjunctie worden gekozen. In onze voorbeelden gaf dat ook geen problemen omdat er steeds slechts één unificeerbare disjunctie was. Voor het algemene geval zullen we de resolutieprocedure moeten uitbreiden met een component die bepaalt welke clause wordt gebruikt.

Er staan ons nu twee wegen open.

- De resolutieprocedure correct en volledig houden zodat de volledige predicatenlogica kan worden behandeld; de procedure wordt daarbij uitgebreid met zo mogelijk slimme methoden om de gezochte afleiding te bepalen met zo weinig mogelijk disjuncties die achteraf overbodig blijken. Er zijn geen methoden bekend die zodanig efficiënt werken dat in alle gevallen gesproken kan worden van berekenen in plaats van bewijs zoeken. In de volgende paragraaf geven we een indruk van de bestaande methoden.

- We kunnen onze inspanning richten op de efficiëntie van de resulterende resolutieprocedure zodat in ieder geval gesproken kan worden van berekenen. Daartoe leveren we iets in van de volledigheid. Dat inleveren kan, en zo gebeurt het ook, door beperkingen op te leggen aan de soort disjuncties waarvoor de procedure werkt. Correctheid moet natuurlijk blijven, anders berekent het systeem wel iets, maar de vraag is wat voor waarde aan het resultaat kan worden gehecht.

Deze weg is gekozen bij de ontwikkeling van Logic Programming en van de programmeertaal Prolog. In paragraaf 17.3 zullen we bespreken tot welke gespecialiseerde resolutieprocedure dit heeft geleid.

17.2 Resolutiestrategieën

Globaal gesproken, is een resolutieprocedure een methode om steeds nieuwe disjuncties af te leiden uit gegeven of eerder afgeleide disjuncties totdat de lege disjunctie $\Box$ is verkregen. De bedoeling van een resolutiestrategie is om de verzameling van toe te voegen disjuncties zo klein mogelijk te houden; zo mogelijk alleen te laten bestaan uit die disjuncties die voor het afleiden van $\Box$ nodig zijn. Dit laatste lukt natuurlijk niet; gestreefd wordt dus naar het genereren van weinig toe te voegen disjuncties, zonder dat daarbij afleidingspaden worden afgesneden. In de volgende twee subparagrafen bezien we in vogelvlucht wat er zoal te koop is.

Disjunctie-georiënteerde strategieën

Bij de onbeperkte resolutie worden de disjuncties in een rij R gezet en aan het einde daarvan worden disjuncties A toegevoegd die in één resolutiestap kunnen worden verkregen uit disjuncties die al in de rij voorkomen. Op deze wijze worden eerst alle mogelijke resolutie afleidingen van lengte 1 onderzocht, dan die van lengte 2, en zo voorts.

De eerste verfijning hiervan bestaat uit het toevoegen van een *bruikbaarheidstest* om alleen die disjuncties aan R toe te voegen waarvan niet bij voorbaat al vast staat dat ze nooit in een afleiding van de lege disjunctie gebruikt kunnen worden. Men zegt dat disjunctie C de disjunctie D *subsumeert* (Engels: *subsumes*) als er een substitutie σ bestaat zodanig dat $C\sigma \subseteq D$. Als dit het geval is en de rij R bevat reeds de disjunctie C dan heeft het geen zin de disjunctie D aan R toe te voegen. Immers er geldt dat R vervulbaar is dan en slechts dan als $R \cup \{D\}$ vervulbaar is. Alvorens dus een disjunctie D toe te voegen controleren we eerst dat D geen tautologie is in de zin dat D een paar complementaire literalen bevat, en dat R geen disjunctie C bevat die D subsumeert. Deze strategie is volledig: $\Box$ is via resolutie afleidbaar uit S dan en slechts dan als deze uit S afleidbaar is door resolutie met bruikbaarheidstest.

Een andere techniek, die eveneens volledig is, is de zogenaamde *lock-resolutie*. We nummeren de literalen in alle disjuncties en staan alleen resolutie op twee disjuncties toe als daarbij de laagst genummerde literalen worden geresolveerd. Bijvoorbeeld:

$$\frac{{}_1P \vee {}_2Q \vee {}_3\neg R \qquad {}_4\neg P \vee {}_5Q \vee {}_6R}{{}_2Q \vee {}_3\neg R \vee {}_6R}$$

is een legale resolutiestap. (Van verschillende voorkomens van een literaal wordt alleen het laagst genummerde voorkomen opgenomen.) Als een andere nummering gebruikt wordt, kan resolutie tussen deze twee disjuncties worden geblokkeerd.

$$\frac{{}_1Q \vee {}_2P \vee {}_3\neg R \qquad {}_4\neg P \vee {}_5Q \vee {}_6R}{\begin{array}{l}\text{Niet resolveerbaar, } {}_1Q \text{ en } {}_4\neg P\\ \text{zijn niet complementair}\end{array}}$$

Een derde, mooie en krachtige techniek is de zogenaamde *semantische resolutie*. Het idee is om willekeurig een interpretatie te kiezen en alleen resolutie toe te staan tussen disjuncties die verschillend zijn met betrekking tot deze interpretatie; dat wil zeggen niet allebei waar en ook niet allebei onwaar. Ook deze strategie is volledig. Het bewijzen van de volledigheid van al deze strategieën is soms lastig en in ieder geval bewerkelijk; verder bespreken hiervan valt buiten het bestek van dit boek.

Afleidingsgeoriënteerde strategieën

Bij de in de voorgaande subparagraaf besproken strategieën worden nieuwe disjuncties afgeleid uit gegeven of reeds eerder afgeleide disjuncties. De manier waarop een bepaalde disjunctie is afgeleid speelt daarbij geen rol; dit wordt dus ook niet bijgehouden. De strategieën van deze subparagraaf doen dat wel. In plaats van aan een rij disjuncties die door de resolutieprocedure wordt aangevuld, moeten we hier denken aan een verzameling afleidingen waarbij de resolutieprocedure steeds een of meer van deze afleidingen aanvult door daar een volgende resolutiestap aan toe te voegen. Wat de strategieën beperken, is de aard van de toegelaten resolutie-afleidingen.

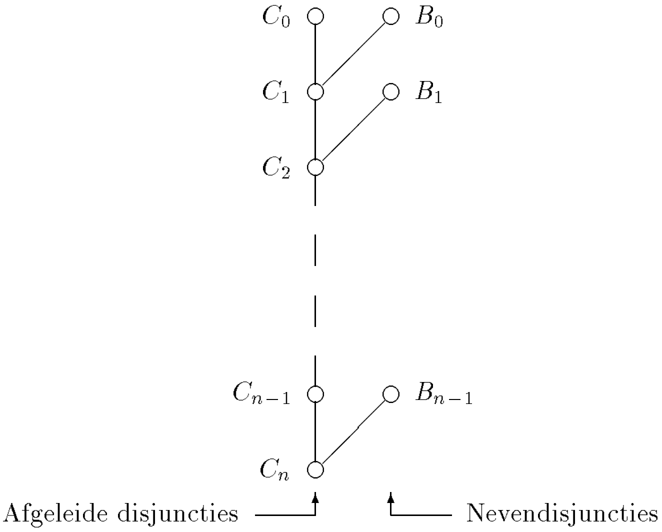
Zij F een verzameling disjuncties en $C_0 \in F$. Een afleiding met *lineaire resolutie* en *topdisjunctie* C_0 is een afleiding van de vorm geschetst in figuur 17.3. De disjuncties C_i heten *afgeleide disjuncties* (Engels: *center clauses*) en de disjuncties B_i *nevendisjuncties* (Engels: *side clauses*). Voor de nevendisjuncties geldt dat $B_i \in F \cup \{C_j \mid 0 \leq j < i\}$.

De lineaire resolutieprocedure houdt een verzameling van lineaire resolutie-afleidingen bij; in elke stap van de procedure wordt een van die afleidingen verlengd met een lineaire resolutiestap tot $\Box$ is afgeleid. In figuur 17.4 staat een resolutie-afleiding van $\Box$ uit de verzameling

$$F \;=\; \{p \vee q, \neg p \vee q, \neg p \vee q, \neg p \vee \neg q\}.$$

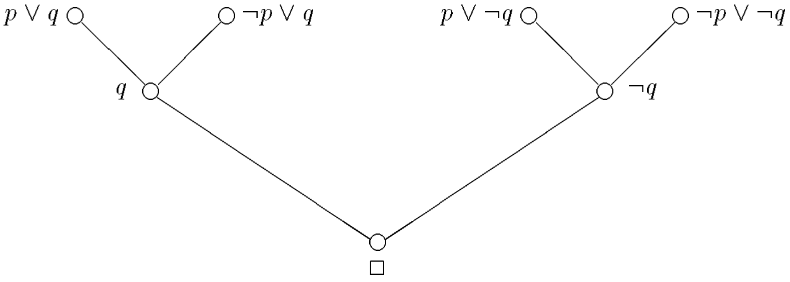
Deze resolutie-afleiding wordt door de lineaire resolutieprocedure niet gevonden. Er is echter ook een lineaire resolutie-afleiding, deze is geschetst in figuur 17.5. Lineaire resolutie is volledig: $\Box$ kan met resolutie uit een verzameling F van clauses worden afgeleid als en alleen als $\Box$ uit F kan worden afgeleid met lineaire resolutie.

Een verfijning van lineaire resolutie is de zogenaamde *input-resolutie*. Dit is lineaire resolutie met als extra beperking dat alle nevendisjuncties uit F komen; het is niet toegestaan om een eerder afgeleide disjunctie te gebruiken

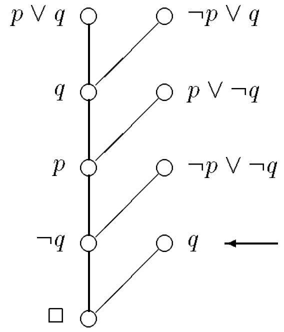


$$C_0 \in F \text{ en } B_i \in F \cup \{C_j \mid 0 \leq j < i\}.$$

Figuur 17.3: Een afleiding met lineaire resolutie.



Figuur 17.4: Een niet-lineaire afleiding van $\Box$ uit $\{p \vee q, \neg p \vee q, \neg p \vee q, \neg p \vee \neg q\}$.



Figuur 17.5: Een lineaire afleiding van $\Box$ uit $\{p \vee q, \neg p \vee q, \neg p \vee q, \neg p \vee \neg q\}$.

als nevendisjunctie. Input-resolutie is *niet* volledig. Het voorgaande voorbeeld, de afleiding van $\Box$ uit $F = \{p \vee q, \neg p \vee q, \neg p \vee q, \neg p \vee \neg q\}$, toont dit aan. Immers, als een nevendisjunctie alleen een disjunctie uit F mag zijn, bestaat de resolvent altijd uit minstens één literaal. Dus $\Box$ kan nooit het resultaat van een input-resolutiestap zijn.

Input-resolutie is *wel* volledig voor $\mathcal{PL}^{\mathcal{H}}$, een deeltaal van $\mathcal{PL}^f$. Deze taal bestaat uit zogenaamde *Horn-clauses* en staat centraal in Logic Programming en Prolog.

17.3 Horn-clauses

17.3.1 DEFINITIE **Horn-clause**
Een Horn-clause *is een disjunctie die ten hoogste één positieve literaal bevat. Een* program-clause *is een Horn-clause die één positieve literaal bevat en een* goal-clause *een Horn-clause die géén positieve literaal bevat. Een* feit *is een program-clause die geen negatieve literalen bevat, dus uit een enkele positieve literaal bestaat. De eerste-ordetaal* $\mathcal{PL}^{\mathcal{H}}$ *bestaat uit de Skolemnormaalvormen van alle formules uit* $\mathcal{PL}^f$ *waarvan de disjuncties Horn-clauses zijn.*

Voor Horn-clauses gebruiken we de notatie die ook voor Prolog gebruikelijk is (zie bladzijde 244) met dien verstande dat we in plaats van het Prolog-symbool `:-` het symbool $\leftarrow$ zullen gebruiken. Dus we schrijven alle positieve literalen gescheiden door komma's, dan het symbool $\leftarrow$ (spreek uit: *als*), en vervolgens alle negatieve literalen gescheiden door komma's. Bijvoorbeeld:

$$\begin{aligned} \text{Program-clause: } & H \leftarrow B_1, B_2, \ldots, B_n, \\ \text{Goal-clause: } & \leftarrow C_1, C_2, \ldots, C_m. \end{aligned}$$

Deze notatie ondersteunt visueel het idee dat een Horn-clause een procedure is met een *head* en een *body*, die zelf weer bestaat uit nul of meer procedure-aanroepen.

17.3.2 STELLING *De Horn-clauselogica* $\mathcal{PL}^{\mathcal{H}}$ *is gesloten onder resolutie.*

BEWIJS Er zijn drie gevallen te onderscheiden: het bepalen van de resolvent van twee program-clauses, twee goal-clauses en van een program-clause en een goal-clause.

1. Twee resolveerbare program-clauses bevatten twee, onderling verschillende positieve literalen. Bij de resolutiestap verdwijnt er precies één, zodat de resolvent weer een program-clause is.
2. Twee goal-clauses bevatten alleen negatieve literalen zodat er geen complementaire literalen in de clauses voorkomen en deze niet resolveerbaar zijn.
3. Een goal-clause bestaat uitsluitend uit negatieve literalen en een program-clause bevat precies één positieve literaal. Als de twee clauses resolveerbaar zijn, verdwijnt bij resolutie die ene positieve literaal, zodat de resolvent weer een goal-clause is.

De voorgaande drie gevallen overdekken alle mogelijkheden en leveren als resolvent steeds een Horn-clause op. Dus $\mathcal{PL}^{\mathcal{H}}$ is gesloten onder resolutie. ■

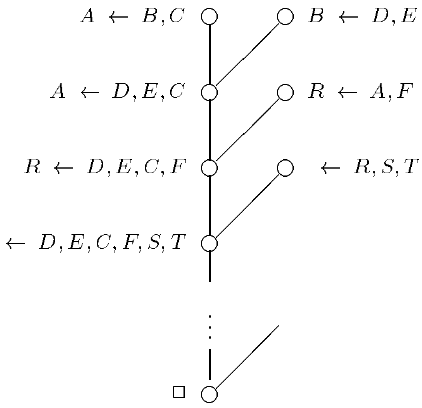
17.3.3 VOORBEELD Resolutie van Horn-clauses.
Gegeven zijn twee program-clauses:

$$\begin{aligned} & H \leftarrow A, B_b, \\ & B_h \leftarrow C, D. \end{aligned}$$

Neem aan dat B_b en B_h unificeerbaar zijn met de substitutie σ als mgu. De resolvent is:

$$H\sigma \leftarrow A\sigma, C\sigma, D\sigma.$$

Het resultaat is dus een nieuwe program-clause waarin de aanroep B_b in de body van procedure H is vervangen door de body van procedure met head



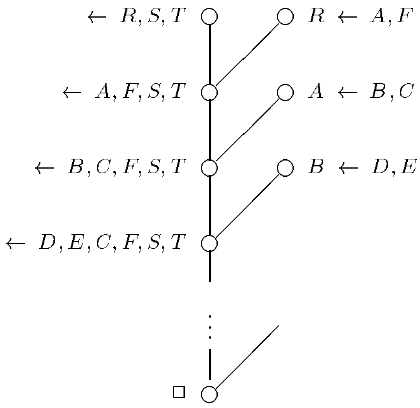
Figuur 17.6: Een lineaire resolutie-afleiding.

B_h. Dit vervangen van een procedure aanroep door de body van de procedure, wordt ook gedaan bij programmatuur ontwikkeling; het heet daar *unfolding*. (Het omgekeerde komt ook voor, dan neemt men een aantal literalen bij elkaar en maakt daar een procedure van. Waar de literalen in een body voorkomen, vervangt men dan deze groep door een aanroep van de nieuwe procedure. Dit heet *folding*.) ■

17.3.4 STELLING **Volledigheid input-resolutie**

Input-resolutie is volledig voor de Horn-clauselogica $\mathcal{PL}^{\mathcal{H}}$.

Het bewijs van deze stelling valt buiten het kader van dit hoofdstuk, maar voor de propositionele Hornlogica is het niet moeilijk de juistheid ervan in te zien uitgaande van de in dit boek evenmin bewezen stelling dat lineaire resolutie volledig is voor de predicatenlogica $\mathcal{PL}^f$. Beschouw een afleiding van $\Box$ uit een verzameling Horn-clauses. In die afleiding wordt minstens één goal-clause gebruikt, want anders zijn alle afgeleide clauses program-clauses en wordt $\Box$ niet bereikt. Als eenmaal een goal-clause is gebruikt, is de afgeleide disjunctie zelf ook een goal-clause. Daarna kunnen dus alleen nog program-clauses als nevendisjunctie worden gebruikt. De afleiding bestaat dus uit een beginstuk waarin via 'unfolding' als het ware hulp-clauses worden afgeleid, vervolgens komt een goal-clause in het spel, vervolgens worden de afgeleide clauses — allemaal goal-clauses— verder verwerkt door resolutie met program-clauses. In figuur 17.6 staat een schets van zo'n lineaire resolutie-afleiding.



Figuur 17.7: Input-resolutie-afleiding equivalent met de lineaire afleiding van figuur 17.6.

Deze afleiding kan in een afleiding met input-resolutie worden omgezet. De topdisjunctie daarvan is de gebruikte goal-clause, hier dus $\leftarrow R, S, T$. Aanroepen van de 'hulp-clauses' in de oorspronkelijke afleiding worden vervangen door een aantal stappen die alleen program-clauses gebruiken. Daarvoor worden dezelfde program-clauses in de omgekeerde volgorde gebruikt als bij de te imiteren lineaire afleiding. Het resultaat daarvan is geschetst in figuur 17.7.

De resolutiemethode die we hier besproken hebben, *input-resolutie voor Horn-clauses*, staat in de literatuur algemeen bekend onder de naam *SLD-resolutie.* De betekenis van de letters is: **L**ineaire resolutie met **S**electiefunctie voor **D**efinite-clauses (Definite-clause is hetzelfde als Horn-clause).

17.3.5 DEFINITIE **SLD-resolutie**

Zij P een verzameling program-clauses en G een goal-clause. Een SLD-afleiding *uit $P \cup \{G\}$ is een al dan niet eindige rij*

$$G_0 = G,\ (G_1, C_1, \sigma_1),\ \ldots,\ (G_i, C_i, \sigma_i),\ \ldots$$

waarin voor alle $i > 0$ geldt:

- *C_i is een verse copie van een program-clause uit P, dat wil zeggen een copie met allemaal nieuwe variabelen, die derhalve nog niet eerder in de rij voorkomen;*
- *σ_i is de mgu (most general unifier) van G_{i-1} en C_i;*

- *G_i is de resolvent van G_{i-1} en C_i.*

Als de rij eindigt met (G_n, C_n, σ_n) en $G_n = \Box$, dan heet de afleiding een SLD-weerlegging*. In dat geval is er een* antwoordsubstitutie *$\alpha = \sigma_1\sigma_2 \ldots \sigma_n$. Ook de restrictie van α tot de variabelen die voorkomen in de gegeven goal-clause G heet* antwoordsubstitutie*.*

17.3.6 Definitie **Correcte antwoordsubstitutie**
Zij P een verzameling program-clauses, $\leftarrow A_1, A_2, \ldots, A_n$ een goal-clause en σ een substitutie. De substitutie is een correcte antwoordsubstitutie *dan en slechts dan als:*

$$U(P) \models U([A_1 \wedge A_2 \wedge \ldots \wedge A_n]\sigma).$$

Hierin is $U(F)$ de universele afsluiting *van de formule F.*

We hebben al gezien dat input-resolutie voor Horn-clauses, en dus SLD-resolutie correct en volledig is. De volgende stelling drukt dit nogmaals uit door te stellen dat SLD-resolutie alle en alleen correcte antwoordsubstituties oplevert.

17.3.7 Stelling **Correctheid en volledigheid SLD-resolutie**
Zij P een verzameling program-clauses, $\leftarrow A_1, A_2, \ldots, A_n$ een goal-clause en σ een substitutie. De substitutie σ is een correcte antwoordsubstitutie dan en slechts dan als er een SLD-weerlegging uit $P \cup \{\leftarrow A_1, A_2, \ldots, A_n\}$ met antwoordsubstitutie α en een substitutie β is zodanig dat $\sigma = \alpha\beta$.

Deze stelling zegt dus dat SLD-resolutie correct en volledig is voor $\mathcal{PL}^{\mathcal{H}}$. Dat aannemend, blijven er twee vragen.

1. Hoe krachtig is $\mathcal{PL}^{\mathcal{H}}$? Als er nauwelijks interessante dingen in $\mathcal{PL}^{\mathcal{H}}$ kunnen worden uitgedrukt, zijn de bereikte resultaten niet interessant.
2. Hoe efficiënt kan de SLD-resolutieprocedure met behoud van correctheid en volledigheid worden geïmplementeerd?

Wat punt 1 betreft, $\mathcal{PL}^{\mathcal{H}}$ is voldoende krachtig. Het kan worden bewezen dat $\mathcal{PL}^{\mathcal{H}}$ gelijkwaardig is aan een programmeertaal zoals Modula-2. Preciezer uitgedrukt: voor iedere berekenbare, n-plaatsige functie $f : \mathbb{N}^n \rightarrow \mathbb{N}$ is er een eindige verzameling program-clauses met daarin een predicaatsymbool p_f zodanig dat $f(a_1, \ldots, a_n)$ gedefinieerd is en het antwoord a oplevert dan en slechts dan als er een SLD-weerlegging uit $P \cup \{\leftarrow p_f(a_1, \ldots, a_n, x)\}$ met antwoordsubstitutie $[x := a]$ is. Bij het bewijs van deze stelling, dat buiten het bestek van dit boek valt, gebruiken we de unaire getalvoorstelling die ook al bij de inleidende beschouwingen van dit hoofdstuk werd gebruikt.

17.4 SLD-resolutie en Prolog

In de voorgaande paragraaf hebben we gezien dat SLD-resolutie correct en volledig is voor $\mathcal{PL}^{\mathcal{H}}$ en dat deze taal interessant is in de zin dat alle algoritmen die men in een programmeertaal als Modula-2 kan opschrijven, ook in $\mathcal{PL}^{\mathcal{H}}$ kunnen worden uitgedrukt. In deze paragraaf besteden we nog aandacht aan enkele implementatiekwesties van SLD-resolutieprocedures. Bij een efficiënte implementatie van een krachtig systeem zoals SLD-resolutie spelen zoals bekend allerlei dingen een rol; de data representatie bijvoorbeeld. We gaan op al deze zaken gaan niet in; we beperken ons tot de vraag hoe we de clauses die voor een resolutiestap zullen worden gebruikt, kunnen bepalen.

Beschouw nogmaals het voorbeeld van de theorie van lijsten (zie paragraaf 17.1 pagina 245 en verder). We voegen aan deze theorie een predicaat *element* toe; de betekenis van $element(x, y)$ is, dat element x op de lijst y voorkomt. Voor de representatie van lijsten door termen gebruiken we nog steeds het binaire functiesymbool $+$. De eigenschappen van *element* zijn bepaald door de volgende axioma's:

$$\forall x[element(x, x + y)],$$
$$\forall x \forall y \forall z[element(x, z) \rightarrow element(x, y + z)].$$

In disjunctieve vorm, en gebruik makend van de speciale Horn-clausenotatie:

$$element(x, x + y),$$
$$element(x, y + z) \;\leftarrow\; element(x, z).$$

en in Prolog-vorm:

```
element(X, X+Y).
element(X, Y+Z) :- element(X, Z).
```

Beschouw nu de goal-clause:

$$\leftarrow\; element(x, a + (b + nil)), element(x, c + (a + nil)).$$

In woorden: gevraagd wordt een x die zowel op de lijst $a + (b + nil)$ als op de lijst $c + (a + nil)$ voorkomt.

Bij het berekenen van de SLD-weerlegging moeten we bepalen met welke literaal uit de goal-clause we de berekening zullen beginnen, de eerste of de tweede. Aangezien we ze toch allebei moeten verwerken, lijkt dat er niet zoveel toe te doen. We doen het maar in de natuurlijke volgorde.

Vervolgens moeten we kiezen welk van de twee program-clauses die het predicaatsymbool *element* in de head-literaal hebben, voor de resolutiestap zal

$$\begin{array}{l}
\leftarrow\ \mathit{element}(x, a + (b + \mathit{nil})), \mathit{element}(x, c + (a + \mathit{nil})), \\
\qquad\qquad \left[\begin{array}{l} \mathit{element}(x_0, x_0 + y_0), \\ {[x\!:=\!a, x_0\!:=\!a, y_0\!:=\!b + \mathit{nil}]} \end{array}\right] \\
\leftarrow\ \mathit{element}(a, c + (a + \mathit{nil})), \\
\qquad\qquad \left[\begin{array}{l} \mathit{element}(x_1, y_1 + z_1)\ \leftarrow\ \mathit{element}(x_1, z_1), \\ {[x_1\!:=\!a, y_1\!:=\!c, z_1\!:=\!a + \mathit{nil}]} \end{array}\right] \\
\leftarrow\ \mathit{element}(a, a + \mathit{nil}), \\
\qquad\qquad \left[\begin{array}{l} \mathit{element}(x_2, x_2 + y_2), \\ {[x_2\!:=\!a, y_2\!:=\!\mathit{nil}]} \end{array}\right] \\
\square\,.
\end{array}$$

Antwoordsubstitutie: $\alpha = [x\!:=\!a]$

Figuur 17.8: Berekening van een element dat zowel op de lijst $a + (b + \mathit{nil})$ als op de lijst $c + (a + \mathit{nil})$ voorkomt.

worden genomen. Dat doet er veel toe, als deze verkeerd wordt gekozen, loopt de afleiding vast. In termen van onze definities is dat geen probleem: we praten over het *bestaan* van SLD-resolutie-afleidingen; hoe we die vinden is vers twee. In de praktijk, op een computer, zullen we niet steeds de juiste keuze kunnen doen en dus moeten zorgen dat het mogelijk is om een eerder gemaakte keuze weer in te trekken en op dat punt een andere keuze te maken. Er zijn allerlei manieren om dit efficiënt te doen. We gaan daar hier niet verder op in. In ons voorbeeld bestaat er natuurlijk een SLD-weerlegging. Ze staat in figuur 17.8.

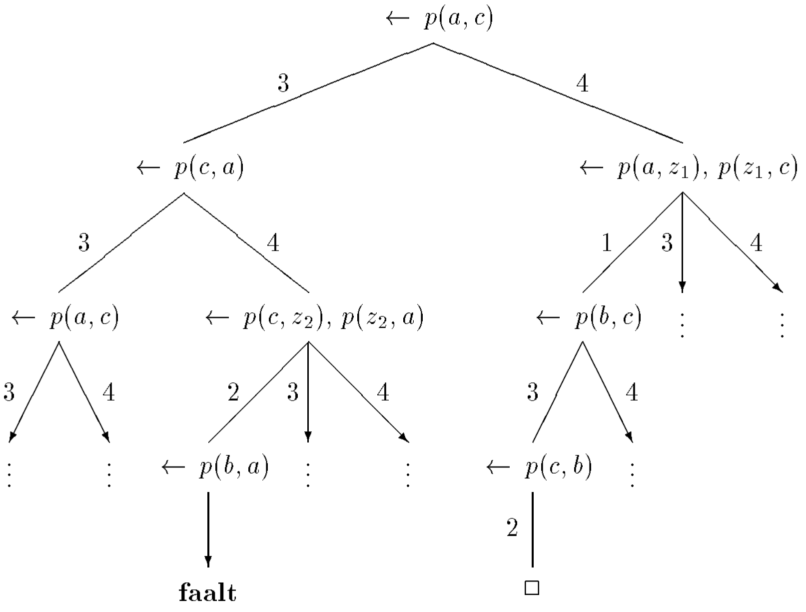
Er zijn dus twee besluiten te nemen: welke literaal en welke program-clause met unificeerbare head te gebruiken.

- De SLD-afleiding is een lineaire resolutie-afleiding waarbij de afgeleide clauses steeds goal-clauses zijn. Bepaald moet worden welke literaal geresolveerd zal worden. Daarvoor gebruiken we een *berekeningsregel.*
- Als bekend is welke literaal geresolveerd zal worden, is er nog een keuze omtrent welke program-clause zal worden gebruikt. Daarvoor gebruiken we een *keuzeregel.*

17.4.1 DEFINITIE **Berekeningsregel en keuzeregel**
Een berekeningsregel *is een functie die toegepast op een goal-clause een literaal uit die goal-clause oplevert. Een* keuzeregel *is een functie die toegepast op een literaal en een eindige verzameling program-clauses een program-clause oplevert waarvan de head unificeerbaar is met de literaal. Als zo'n program-clause niet bestaat levert de functie $\perp$ op.*

De berekeningsregel is onbelangrijk, dat zagen we al in het voorbeeld. En inderdaad, SLD-resolutie blijft correct en volledig onafhankelijk van de gebruikte berekeningsregel.

17.4.2 STELLING **Correctheid en volledigheid SLDr-resolutie**
Zij P een verzameling program-clauses, $\leftarrow A_1, A_2, \ldots, A_n$ een goal-clause, σ een substitutie,



Figuur 17.9: Een SLD-boom.

en r een berekeningsregel. Zij verder SLDr de SLD-resolutie aangevuld met het voorschrift dat berekeningsregel r moet worden gebruikt om te bepalen welke literaal zal worden geresolveerd. De substitutie σ is een correcte antwoordsubstitutie dan en slechts dan als er een SLDr-weerlegging uit $P \cup \{\leftarrow A_1, A_2, \ldots, A_n\}$ met antwoordsubstitutie α en een substitutie β is zodanig dat $\sigma = \alpha\beta$.

De berekeningsregel is niet volkomen onbelangrijk: de lengte van een SLD-weerlegging kan variëren afhankelijk van de berekeningsregel, zodat het uit efficiëntie-overwegingen zinvol kan zijn een slimme regel te bedenken. De meeste Prolog implementaties hebben geen slimme berekeningsregel, men gebruikt de in onze notatie meest linkse literaal.

Als de berekeningsregel vastgelegd is, blijft nog de keuze van de program-clause die zal worden gebruikt, over. Een overzichtelijke manier om alle mogelijkheden weer te geven is in een zogenaamde SLD-boom. Neem bijvoorbeeld het programma gegeven door de volgende verzameling P van Horn-clauses:

1. $p(a,b)$,
2. $p(c,b)$,
3. $p(x,y) \leftarrow p(y,x)$,
4. $p(x,y) \leftarrow p(x,z), p(z,y)$.

waarin x, y en z variabelen zijn en a, b en c constanten. In figuur 17.9 staat een deel van de SLD-boom bepaald door de goal $\leftarrow p(a,c)$. In de SLD-boom heeft elke knoop een etiket, namelijk een goal-clause, en verder heeft elke knoop een aantal opvolgers en wel één voor iedere program-clause waarvan de head unificeerbaar is met de literaal die voor deze resolutiestap wordt gebruikt. Welke literaal uit de goal-clause dat is, wordt dus bepaald door de berekeningsregel. In figuur 17.9 is steeds de meest linkse literaal van de goal-clause genomen. Langs de takken staat aangegeven welke program-clause is gebruikt.

Zoals men in de SLD-boom kan zien, zijn er de volgende mogelijkheden:

- De berekening —dat wil zeggen de resolutiedeductie— termineert niet. Dit doet zich voor in de meest linkse tak van de boom.
- De berekening kan niet verder worden voortgezet, terwijl $\Box$ niet is afgeleid. Men zegt dat in dat geval dat de berekening *eindig faalt.*
- De berekening leidt tot $\Box$. Men zegt in dat geval dat de berekening *succesvol* is.

De kunst is natuurlijk om zo'n succesvolle berekening te vinden, als die tenminste bestaat. Een slimme keuzeregel, die zo snel mogelijk een succesvolle berekening vindt en in elk geval vermijdt dat het zoeken niet termineert terwijl er een succesvolle berekening bestaat, verdient de voorkeur.

In Prolog implementaties houdt men het eenvoudig en probeert men de program-clauses waarvan de head-literaal unificeerbaar is met de goal-literaal, in de volgorde die ze in het programma hebben. De keuzeregel heeft op verschillende manieren invloed. Een belangrijk aspect is de hoeveelheid herstelwerk dat moet worden gedaan als een keuze van een program-clause verkeerd blijkt te zijn geweest, zodat deze moet worden ingetrokken en vervangen door een andere keuze. Het gebruikte procédé noemt men *backtracking.* Er zijn verschei-dene slimme backtracking-methoden ontwikkeld, maar die vinden tenslotte toch niet zo veel aftrek in Prolog implementaties; men heeft zich geconcentreerd op het zo efficiënt mogelijk implementeren van backtracking zelf.

De keuzeregel die Prolog gebruikt leidt soms tot niet-terminerende berekeningen terwijl er wel een SLD-weerlegging bestaat. Beschouw het programma:

$$p(x,y) \leftarrow p(y,x),$$
$$p(a,b).$$

en de goal $\leftarrow p(a,b)$. De resolvent van deze goal-clause en de tweede program-clause levert $\Box$ op. Volgens de Prolog keuzeregel echter wordt steeds de resolvent bepaald met de eerste program-clause. Dat leidt nooit tot $\Box$. Een beginstuk van de berekening staat in figuur 17.10.

17.5 Opgaven

1. Bereken met behulp van resolutie $2+3$. Dat wil zeggen, laat door middel van resolutie zien dat **N3, N5'** $\models \exists x[\mathrm{s}(\mathrm{s}(\mathbf{0})) + \mathrm{s}(\mathrm{s}(\mathrm{s}(\mathbf{0}))) = x]$. Wat is de antwoordsubstitutie?

2. Geef *alle* bewijzen van **N3, N5'** $\models \exists x \exists y[x + y = \mathrm{s}(\mathrm{s}(\mathrm{s}(\mathrm{s}(\mathbf{0}))))]$ door middel van resolutieweerleggingen. Welke antwoordsubstituties worden hierbij verkregen?

3. Beschouw het programma P (de theorie) om lijsten achter elkaar te schakelen, dat bestaat uit de volgende twee clauses.

$$\mathit{schakel}(\mathit{nil}, x, x)$$
$$\neg \mathit{schakel}(x, y, z) \vee \mathit{schakel}(a + x, y, a + z)$$

$$
\begin{array}{ll}
\leftarrow p(a,b), & \\
& \left[\begin{array}{l} p(x_1,y_1) \leftarrow p(y_1,x_1), \\ {[}x_1{:=}a, y_1{:=}b{]} \end{array}\right] \\
\leftarrow p(b,a), & \\
& \left[\begin{array}{l} p(x_2,y_2) \leftarrow p(y_2,x_2), \\ {[}x_2{:=}b, y_2{:=}a{]} \end{array}\right] \\
\leftarrow p(a,b), & \\
& \left[\begin{array}{l} p(x_3,y_3) \leftarrow p(y_3,x_3), \\ {[}x_3{:=}a, y_3{:=}b{]} \end{array}\right] \\
\leftarrow p(b,a), & \\
\vdots &
\end{array}
$$

Figuur 17.10: Een niet-terminerende berekening.

waarin a, x, y en z variabelen zijn, en lijsten gerepresenteerd zijn als termen opgebouwd met de binaire operator $+$. Bereken het resultaat van het aaneenschakelen van de lijsten $a+(b+nil)$ en $c+(d+(a+nil))$ door een resolutieweerlegging te geven uit:

$$P \cup \{\neg schakel(a+(b+nil), c+(d+(a+nil)), x)\}.$$

Is deze weerlegging lineair? Input? Geef een weerlegging die input-resolutie gebruikt en ook een weerlegging die wel lineair is, maar niet input.

4. Beschouw de theorie van lidmaatschap van lijsten zoals geformaliseerd door volgende verzameling P van Horn-clauses

$$
\begin{array}{l}
element(x, x+y) \\
element(x, y+z) \leftarrow element(x, z)
\end{array}
$$

Er zijn drie SLD-weerleggingen uit $P \cup \{\leftarrow element(x, a+(b+(c+nil)))\}$. Welke zijn de antwoordsubstituties?

5. Beschouw de volgende verzameling P van Horn-clauses:

$$
\begin{array}{l}
p(a,b) \\
p(c,b) \\
p(x,y) \leftarrow p(y,x) \\
p(x,y) \leftarrow p(x,z), p(z,y)
\end{array}
$$

waarin x, y en z variabelen zijn en a, b en c constanten. In figuur 17.9 staat een SLD-weerlegging van $P \cup \{ \leftarrow p(a, c)\}$. Is deze weerlegging toegestaan bij de keuzeregel die Prolog hanteert? (Dat wil zeggen als steeds de eerste clause met een unificeerbaar head wordt genomen in de weerlegging). Is het mogelijk de volgorde van de clauses zo te kiezen dat er een SLD-weerlegging bestaat die deze keuzeregel respecteert? Zo ja, welke volgorde(n) is (zijn) dat? Zo nee, bewijs dat zo'n volgorde niet bestaat.

6. Koop een Prolog systeem voor uw PC.

Index