Valuatie van argumenten in frames en het aanmaken van geprefereerde en stabiele extensies

Sjoerd Druiven Maarten Grachten

31 augustus 2000

Inhoudsopgave

1 Inleiding **1**

2 Argumentatie-frames **3**
2.1 Valuatie van argumenten 3
2.2 Relaties tussen valuaties en frame-eigenschappen 6
2.3 Extensies 8
2.4 Relaties tussen valuaties en extensies 10

3 Implementatie van de valuering en het aanmaken van extensies **15**
3.1 Het basis-algoritme 15
3.1.1 Ontoereikendheid van het basis-algoritme 16
3.2 Het aanmaken van de extensies 18

4 Toepassing van het programma **19**

A Broncode van het Prolog-programma **21**

1 Inleiding

In dit project hebben we ons ten doel gesteld het projectvoorstel van Gerard Vreeswijk over *defeasible reasoning* te realiseren. De basis van het project is de implementatie in Prolog van het argumentatie-theoretische begrip *defeasible*. In de argumentatieleer is een argument defeasible als het verslagen kan worden door één of meer andere argumenten. De interne structuur van argumenten doet in dit geval niet ter zake; het gaat slechts om een formele aanvalsrelatie tussen argumenten. Een samenhangend geheel van argumenten kan daarom worden weergegeven als een frame, met als domein de argumenten in kwestie en als relatie op dat domein de aanvalsrelatie tussen deze argumenten ('argument a valt argument b aan' wordt weergegeven als a→b of b←a). Op deze manier zou je bijvoorbeeld de dialoog:

Paul: 'Het broeikas-effect bestaat, want het wordt elk jaar warmer.' (a)

Ina: 'Dat het elk jaar warmer wordt zou ook aan klimatologische schommelingen kunnen liggen.' (b)

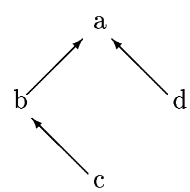
Paul: ‘Daarvoor is de stijgingsnelheid van de temperatuur te groot.’ (c)

Ina: ‘Het broeikas-effect is gewoon een verzinsel van de media.’ (d)

in een frame F kunnen weergeven als:

$$F = \langle D,R \rangle,\ D=\{a,b,c,d\},\ R=\{\langle b,a\rangle,\langle c,b\rangle,\langle d,a\rangle\}$$

of grafisch, zoals in figuur 1.



Figuur 1: Grafische weergave van frame F

Gegeven een dergelijk frame, kan de vraag gesteld worden welke argumenten uit het frame worden verslagen en welke onverslagen blijven. Daartoe moet allereerst een geschikte definitie van ‘verslagen’ (Engels: *defeated*) en ‘onverslagen’ (Engels: *undefeated*) gegeven worden. Op basis van deze definitie kan het frame worden ‘doorgerekend’ om zo tot een valuatie van de argumenten (met waarden verslagen en onverslagen) te komen.

Een nuttige taak is nu het vinden van zinvolle deelverzamelingen van het frame-domein, zogenaamde *extensies*. Wat precies onder zinvol verstaan moet worden, is afhankelijk van de toepassing. Een voordehandliggend gebruik van software die extensies van argumentatie-frames berekent, is om ten eerste de toestand van een discussie te representeren (door middel van het frame) en ten tweede bepaalde standpunten in deze discussie te bepalen. Hierbij geldt een standpunt als een verzameling argumenten. Een extensie is dus een zinvolle extensie in deze toepassing als hij, opgevat als standpunt in de discussie, een wenselijk standpunt oplevert. Ons programma zal naast een valuatie-algoritme ook verscheidene gangbare extensies kunnen bepalen bij een gegeven framework.

Er zijn meerdere toepassingen denkbaar van een programma dat standpunten in discussies kan bepalen. Eén daarvan is het bepalen van een status quo in discussies en debatten uit het dagelijks leven; maar zeker zo interessant is de toepassing in een autonome rationele agent die zelfstandig kan deelnemen in een debat met personen of met andere agents. In die gevallen moet de agent uit het geheel van argumenten in het debat een rationele keuze maken om zijn standpunt te vormen. De gekozen deelverzameling dient dan in ieder geval niet inconsistent te zijn en ook verslagen argumenten moeten gemeden worden voor een houdbaar standpunt. Naast deze algemeen aanvaarde eisen aan de gekozen extensie zijn er tal van minder onomstreden voorwaarden waaraan een goede extensie in de context van het debat zou kunnen voldoen.

Een interessant aspect van extensies als standpunten is dat verschillende soorten extensies overeen komen met verschillende soorten standpunten in discussies. Een extensie als de stabiele extensie correspondeert in het debat met een ‘overtuigd’ standpunt, dat wil zeggen dat elk argument ofwel behoort tot het standpunt of erdoor wordt aangevallen. Er zijn echter ook extensies (bv. de 'gegronde' (grounded) extensies) waarvan het corresponderende standpunt niet elk argument aanvalt dat niet tot het standpunt behoort. Men zou dit soort standpunten 'bescheiden' kunnen noemen. Een uiteenzetting over deze correspondenties tussen soorten extensies en soorten standpunten valt buiten het bestek van dit paper.

In de rest van dit paper zullen we de theorie omtrent argumentatieframes uiteenzetten (paragraaf 2) voor zover de implementatie daar aanleiding toe geeft; verder zullen we uitleggen op welke manier we de valuering van de argumenten hebben geïmplementeerd en hoe verschillende extensies bepaald worden (paragraaf 3). Ook zullen we de resultaten illustreren aan de hand van een concreet voorbeeld-debat (paragraaf 4).

2 Argumentatie-frames

2.1 Valuatie van argumenten

Allereerst moeten we benadrukken dat het domein van argumentatie-frames per definitie een eindige verzameling argumenten is. Deze eigenschap van frames is van belang in de bewijzen van sommige stellingen in de volgende paragrafen. Daarnaast zullen we ons voor het gemak beperken tot het bekijken van frames die 'uit één geheel bestaan', dat wil zeggen dat voor het frame geldt dat er voor twee willekeurige argumenten a en b (a $\neq$ b) een eindige rij argumenten a_0, ..., a_n is zodat geldt $a \circ a_0$, $a_0 \circ a_1$, ..., $a_{n-1} \circ a_n$, $a_n \circ b$ is, met $\circ \in \{\leftarrow, \rightarrow\}$. Dit is voor argumentatie-frames geen vreemde eis, want frames die deze eigenschap niet hebben, representeren eigenlijk meerdere onafhankelijke debatten.

Zojuist is al aan de orde gekomen dat argumenten in een frame een valuatie kunnen krijgen. We kunnen argumenten van een waarde voorzien door functies te definiëren die argumenten een waarde toekennen. Zulke functies noemen we 'valuatie-functies', of kortweg 'valuaties'.

Definitie 1 *Een* ***valuatie-functie*** *van een frame* $F=<D,R>$*, is een functie* $f{:}X \longrightarrow \{verslagen, onverslagen\}$*, waarbij geldt* $X \subseteq D$ *en* $X \neq \emptyset$.

De valuatie-functie is dus een partiële functie op het domein van het frame. We zijn voornamelijk geïnteresseerd in functies waarvan het domein $\subseteq$-maximaal is, omdat deze functies het meest informatief zijn met betrekking tot de valuaties van argumenten. We noemen deze functies voortaan 'maximale functies'.

Definitie 2 *Een valuatie-functie* $f{:}X \longrightarrow \{verslagen, onverslagen\}$ *van een frame* $F=<D,R>$*, is* ***maximaal*** *desda er geen valuatie-functie* $g{:}Y \longrightarrow \{verslagen, onverslagen\}$ *is (met* $Y \subseteq D$*), zodanig dat* $X \subset Y$.

Ten slotte beperken we de klasse van relevante valuatie-functies door een tweede eis te stellen; we willen alleen naar functies kijken die op een intuïtieve (samenhangende) manier de waarden 'verslagen' en 'onverslagen' toekennen. We hebben de definities van Vreeswijk, in [2, p.3] als uitgangspunt genomen.

Definitie 3 *Een valuatie-functie f van een frame* $F=<D,R>$ *is* ***samenhangend*** *desda*

$$f(a) = \begin{cases} onverslagen & \text{als geldt } \forall_{x \subseteq D}(x \rightarrow a \;\rightarrow\; f(x) = verslagen) \\ verslagen & \text{als geldt } \exists_{x \subseteq D}(x \rightarrow a \;\wedge\; f(x) = onverslagen) \end{cases}$$

Hiermee hebben de begrippen 'verslagen' en 'onverslagen' een éénduidige definitie gekregen:

Definitie 4a *Een argument a uit een frame F is onder valuatie f* ***onverslagen*** *desda alle aanvallers uit F van a onder f* verslagen *zijn.*

Definitie 4b *Een argument a uit een frame F is onder valuatie f* ***verslagen*** *desda er tenminste één aanvaller uit F van a onder f* onverslagen *is.*

Hierbij zijn de begrippen 'aanvaller' en 'verdediger' van belang:

Definitie 5a *Een argument a uit een fame F is een* ***aanvaller*** *van een argument b uit F desda* $a \rightarrow b$ *geldt in F.*

Definitie 5b *Een argument a uit een frame F is een* ***verdediger*** *van een argument b uit F desda er een argument c uit F is waarvoor* $c \rightarrow b$ *en* $a \rightarrow c$ *gelden in F.*

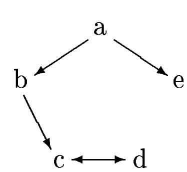
De circulariteit van de definitie van verslagen en onverslagen kan problemen opleveren in geval van *loops* in het frame. Een loop houdt in dat een argument door de pijlen van de aanvalsrelatie met zichzelf verbonden is (in dit geval geldt ofwel dat het argument zichzelf (indirect) aanvalt, ofwel dat het zichzelf (indirect) verdedigt).

Definitie 6 *In een frame F=<D,R> een vormt een rij argumenten* $a_1, \ldots, a_n$ *(*$a_i \in D$ *voor* $1 \leq i \leq n$*) een* ***loop*** *desda er geldt* $a_1 \rightarrow a_2, \ldots, a_{n-1} \rightarrow a_n$*,* $a_n \rightarrow a1$*.*

Definitie 7 *In een frame F=<D,R> een vormt een rij argumenten* $a_1, \ldots, a_n$ *(*$a_i \in D$ *voor* $1 \leq i \leq n$*) een* ***even loop*** *desda de rij een loop vormt en n even is.*

Definitie 8 *In een frame F=<D,R> een vormt een rij argumenten* $a_1, \ldots, a_n$ *(*$a_i \in D$ *voor* $1 \leq i \leq n$*) een* ***oneven loop*** *desda de rij een loop vormt en n oneven is.*

Het frame in figuur 2 bevat een oneven loop.

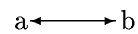


Figuur 2: Frame met een loop

In dit frame zijn argumenten b en e verslagen en is argument a onverslagen. Het opmerkelijke aan dit frame is de onbekendheid van de valuatie van de argumenten c en d. Het is met behulp van de definitie niet af te leiden dat argument c verslagen is, noch is het mogelijk af te leiden dat argument c onverslagen is.

Dit geldt ook voor argument d. Het is echter niet zo c en d daarom geen valuatie hebben. Als argument c verslagen zou zijn is argument d noodzakelijk onverslagen en vice versa.

Het fenomeen van onbekendheid van de valuatie komt het duidelijkst naar voren in het bekende Nixon-Diamond voorbeeld. Het frame is afgebeeld in figuur 3.



Figuur 3: Nixon-Diamond voorbeeld

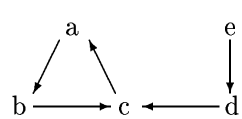
In dit voorbeeld staat argument a voor 'Nixon is een anti-pacifist, omdat hij een republikein is' en argument b staat voor 'Nixon is een pacifist aangezien hij een Quaker is'. In dit voorbeeld is goed in te zien dat iemand elk van de argumenten kan aanhangen, zonder dat hij in rechtvaardigingsproblemen komt: welk van beide argumenten men ook aanhoudt, het kan verdedigd worden met zichzelf (binnen dit frame). Op dit onderwerp komen we later terug bij toelaatbare verzamelingen.

De zojuist geïllustreerde notie van onbekendheid van een valuatie kan gepreciseerd worden.

Definitie 9 *Van een argument a uit een frame F is de valuatie* ***onbekend*** *desda er een maximaal samenhangende valuatie van F is waarin a verslagen is en er tevens een maximaal samenhangende valuatie van F is waarin a onverslagen is.*

De valuatie van een argument is onbekend wanneer die valuatie zodanig afhankelijk is van zichzelf dat het argument zichzelf (indirect) verdedigt. Het is makkelijk in te zien dat dit alleen het geval kan zijn wanneer het argument deel uitmaakt van een even loop in het frame. In paragraaf 2.2 zullen we een verband aantonen tussen het behoren tot een even loop en de onbekendheid van de valuatie van een argument.

Wanneer een loop uit een oneven aantal argumenten bestaat, vallen de argumenten uit de loop indirect zichzelf aan. Dat is het geval in het frame van figuur 4.



Figuur 4: Frame met een oneven loop

In dit frame is argument d verslagen en argument e onverslagen. De rest van de argumenten, a, b en c, vormen een oneven loop. De valuatie van deze argumenten is niet duidelijk. Als een aanname wordt gedaan over een valuatie van de argumenten ontstaan er inconsistenties; stel dat argument a onverslagen is, dan heeft b een onverslagen aanvaller en daaruit volgt dat b verslagen is. Argument c heeft nu alleen maar verslagen aanvallers en is daarom onverslagen. Omdat argument c argument a aanvalt moet a verslagen zijn, hetgeen in tegenspraak is met de aanname dat a onverslagen is. Stel nu dat argument a verslagen is, dan moet gelden dat c onverslagen is, en dus dat b verslagen is, waaruit volgt dat a onverslagen is; alweer een tegenspraak!

Hier blijkt dus dat uit het feit dat een argument niet onverslagen is, *niet* kan worden afgeleid dat het argument verslagen is (en vice versa). Met andere woorden: de wet van de uitgesloten derde gaat hier niet op. Deze eigenschap van een argument, niet verslagen en niet onverslagen zijn, noemen we de *onbepaaldheid* van de valuatie van dat argument.

Analoog aan de onbekendheid van een valuatie lijkt een INUS-conditie [1] voor de onbepaalde valuatie te zijn dat het betreffende argument deeluitmaakt van een oneven loop. Wanneer een argument deel uitmaakt van een oneven loop, impliceert dat nog niet dat de valuatie van het argument onbepaald is; het behoren tot een oneven loop is echter wel noodzakelijk voor de eigenschap dat het argument zichzelf (indirect) aanvalt. Dit is weer een voldoende voorwaarde (maar geen noodzakelijke, zoals we zometeen zullen zien) voor de onbepaaldheid van het argument. We zullen dit verband aantonen in 2.2.

Hieronder volgt de definitie van de onbepaaldheid van een valuatie:

Definitie 10 *Van een argument a uit een frame F is de valuatie* ***onbepaald*** *desda er geen maximaal samenhangende valuatie is waarin a verslagen is en ook geen maximaal samenhangende valuatie waarin a onverslagen is.*

Nu dringt zich de vraag op of, en zoja hoe, de valuatie van een argument beïnvloed wordt wanneer het wordt aangevallen door een argument met een onbepaalde valuatie. Wanneer we de valuatie bepalen aan de hand van de huidige noties verslagen en onverslagen, dan heeft een aanval door een onbepaald argument geen gevolgen als het aangevallen argument ook door één of meer onverslagen argumenten wordt aangevallen. Met dat laatste is immers voldaan aan de voorwaarde om verslagen te zijn.

Dan is er nog de vraag wat de valuatie is van een argument dat aangevallen wordt door een onbepaald argument en niet door onverslagen argumenten. Volgens de definities op pagina 4 geldt een argument als onverslagen als alle aanvallers verslagen zijn. Dat is hier niet geval, omdat er een onbepaalde aanvaller is; het argument is dus niet onverslagen. Maar er is geen onverslagen aanvaller, dus volgens de definitie geldt het argument ook niet als verslagen. Er zijn dus geen maximaal samenhangende valuaties waarin het argument verslagen is, noch zijn er dergelijke valuaties waarin het argument onverslagen is; daarmee is de valuatie onbepaald. Uit de definities van respectievelijk 'verslagen', 'onverslagen' en 'onbepaald', is dus af te leiden dat argumenten die tenminste één onbepaalde aanvaller hebben en geen onverslagen aanvallers, zelf ook onbepaald zijn.

2.2 Relaties tussen valuaties en frame-eigenschappen

Er zijn (weinig verbazend) duidelijke verbanden tussen de valuatie van argumenten en frame-eigenschappen; onder frame-eigenschappen verstaan we eigenschappen van de structuur in de aanvalsrelatie. Typische frame-eigenschappen zijn de even en oneven loops in de aanvalsrelatie.

[1] Een I(nsufficient)N(essecary condition for an)U(nnessecary)S(ufficient)-conditie is een voorwaarde die op zichzelf niet voldoende is, maar wel noodzakelijk voor een voldoende voorwaarde (die weer niet noodzakelijk is).

Met name twee verbanden springen in het oog, namelijk het verband tussen even loops en onbekende argumenten en, ten tweede, het verband tussen onbepaalde argumenten en oneven loops. Zoals in de vorige paragraaf al doorschemerde, is er geen één-op-één-relatie tussen onbekende valuatie en het behoren tot een even loop, respectievelijk tussen onbepaalde valuatie en het behoren tot een oneven loop. Dat heeft een aantal redenen:

1. Wanneer een argument behorend tot een loop (even of oneven) wordt aangevallen door een onverslagen argument, is er geen sprake meer van onbekende, noch van onbepaalde valuatie en zijn alle loop-elementen verslagen of onverslagen.
2. Argumenten die worden aangevallen door een argument uit een oneven loop, ook al behoren zij zelf niet tot de loop, zijn onbepaald.
3. Argumenten die worden aangevallen door een argument uit een even loop, ook al behoren zij zelf niet tot de loop, zijn onbekend.
4. Echter, wanneer een argument wordt aangevallen door twee argumenten uit dezelfde even loop, en deze twee argumenten zijn via een oneven aantal aanvalspijlen met elkaar verbonden, dan is het argument dat zij gezamenlijk aanvallen onverslagen.

Door deze factoren worden de verbanden enigszins gecompliceerd. Het eerste punt suggereert echter dat het vermoede verband onbekend/even loop en onbepaald/oneven loop wel bestaat, maar dan bij een specifiek soort loops, namelijk loops waarvan de argumenten niet worden aangevallen door onverslagen argumenten. Om die reden introduceren we nu een nieuw begrip, de *geïsoleerde loop*:

Definitie 11 *In een frame F vormen argumenten a_0, ..., a_n onder valuatie f een* ***geïsoleerde loop*** *desda deze argumenten een loop vormen en er geen argument b is dat onder f onverslagen is, waarvoor geldt dat $b \rightarrow a_0$, of ..., of $b \rightarrow a_n$* [2]*.*

Nu kunnen de volgende twee stellingen bewezen worden:

Stelling 1 *Als een argument deel uitmaakt van een geïsoleerde even loop, is de valuatie van dat argument onbekend.*

Bewijs: Stel een argument, zeg a, maakt deel uit van een geïsoleerde even loop. Er zijn dan argumenten a, a_0, ..., a_n met n even, zodanig dat $a \rightarrow a_0$, $a_0 \rightarrow a_1$, ..., $a_{n-1} \rightarrow a_n$, $a_n \rightarrow a$. Nu moeten we laten zien dat er een maximaal samenhangende valuatie van de argumenten a, a_0, ..., a_n is [3], waarbij a onverslagen is, en een maximaal samenhangende valuatie waarbij a verslagen is. Welnu, neem a onverslagen, en valueer vervolgens alle argumenten a_i met i even als verslagen en de argumenten a_i met i oneven als onverslagen. Dan ontstaat er een maximaal samenhangende valuatie waarin a onverslagen is. Nemen we a verslagen en vervolgens alle a_i met i even als onverslagen en de argumenten a_i met i oneven als verslagen, dan ontstaat er eveneens een maximaal samenhangende valuatie, maar nu met a als verslagen argument. Hieruit volgt direct dat de valuatie van a onbekend is. □

[2] b mag identiek zijn met a_i, $0 \leq i \leq n$.

[3] Hoe de andere argumenten in het frame gevalueerd zijn is hier niet relevant

Stelling 2 *Als een argument deel uitmaakt van een geïsoleerde oneven loop, is de valuatie van dat argument onbepaald.*

Bewijs: Stel een argument, we noemen het a, maakt deel uit van een geïsoleerde oneven loop. Er zijn dan argumenten a, a_0, ..., a_n met n oneven, zodanig dat a$\rightarrow a_0$, $a_0 \rightarrow a_1$, ..., $a_{n-1} \rightarrow a_n$, $a_n \rightarrow$a. Stel nu dat er een maximaal samenhangende valuatie f is waarin a onverslagen is a. Dan volgt vanwege de samenhangendheid van f dat a_0 verslagen is, a_1 is dan onverslagen, ..., a_{n-1} verslagen, a_n onverslagen, en daaruit: a verslagen. Maar er gold reeds dat a onverslagen was: tegenspraak! Er is dus geen maximaal samenhangende valuatie waarin a onverslagen is. Stel nu dat er een maximaal samenhangende valuatie f is waarin a verslagen is; a_0 is onverslagen in f, a_1 is verslagen in f, ..., a_{n-1} onverslagen, a_n verslagen, en daaruit volgt: a onverslagen. Wederom een tegenspraak. Er is dus geen maximaal samenhangende valuatie waarin a onverslagen is, noch is er een dergelijke valuatie waarin a verslagen is. Met definitie 10 volgt dat a onbepaald is. □

In beide stellingen vormt de frame-eigenschap een *voldoende* voorwaarde voor de valuatie. Het geven van noodzakelijke voorwaarden voor de onbekende en onbepaalde valuatie is lastiger, omdat dan rekening gehouden moet worden met de argumenten met een onbekende of onbepaalde valuatie, die niet zelf tot een loop behoren. We zullen bij het bespreken van de implementatie van de valuering in paragraaf 3 de valuatie van dit soort argumenten verder aan de orde stellen.

2.3 Extensies

Een rationele agent die deelneemt aan debatten, werkt met verzamelingen argumenten en daarom is het nuttig om wat (wenselijke) eigenschappen van argumenten en verzamelingen argumenten op een rijtje te zetten.

Allereerst definiëren we de begrippen 'aanvallen' en 'verdedigen' naast argumenten onderling, ook voor argumenten in relatie tot verzamelingen:

Definitie 12a *In een frame F=<D,R> geldt dat een argument $a \in D$* ***aangevallen*** *wordt door een verzameling $S \subseteq D$ desda er een argument $b \in S$ is zodanig dat b argument a aanvalt.*

Definitie 12b *In een frame F=<D,R> geldt dat een argument $a \in D$* ***verdedigd*** *wordt door een verzameling $S \subseteq D$ desda er voor elk argument $b \in D$ dat a aanvalt, een argument $c \in S$ is zodanig dat c argument b aanvalt.*

Zoals eerder opgemerkt zal een rationele agent geen verslagen argumenten aanhouden; ook argumenten met een onbepaalde valuatie zijn niet redelijk te verdedigen. De verzameling argumenten waaruit een deelverzameling als standpunt moet worden aangehouden kan dus worden beperkt tot de onverslagen argumenten. Verder zal een agent die argumenten aannemen die hij kan verdedigen. Dit doet hij door zoveel mogelijk de aanvallers van zijn argumenten aan te vallen met (andere) argumenten die hij al had aangenomen. Argumenten die verdedigd kunnen worden vanuit een verzameling van argumenten zijn 'acceptabel' (Engels: *acceptable*) met betrekking tot die verzameling.

Definitie 13 *In een frame $F=<D,R>$ geldt dat een argument $a \in D$* ***acceptabel*** *is met betrekking tot verzameling $S \subseteq D$ desda er voor een willekeurige aanvaller b van a een aanvaller c is waarvoor geldt $c \in S$.*

Op de notie van acceptabiliteit is eigenschap 'toelaatbaarheid' (Engels: *admissibility*) van verzamelingen gebaseerd:

Definitie 14 *In een frame $F=<D,R>$ geldt dat een verzameling $S \subseteq D$* ***toelaatbaar*** *is desda voor een elk argument $a \in S$ geldt dat a acceptabel is met betrekking tot S en S bovendien conflict-vrij is.*

Een verzameling die toelaatbaar is, valt dus elke aanvaller van zijn elementen aan. Verder is een toelaatbare verzameling conflict-vrij; dat wilt zeggen dat de argumenten uit de verzameling elkaar niet aanvallen:

Definitie 15 *In een frame $F=<D,R>$ geldt dat een verzameling $S \subseteq D$* ***conflict-vrij*** *is desda er geen argumenten $a \in S$ $b \in S$ zijn zodat $a \leftarrow b$ geldt.*

Ter illustratie: in het Nixon-Diamond voorbeeld is argument a acceptabel met betrekking tot de verzameling {a} en aangezien a het enige argument in {a} is, is {a} toelaatbaar. Verder is b acceptabel met betrekking tot {b} en dus is ook {b} toelaatbaar.

In [1] stelt Dung op basis van de begrippen acceptabel en toelaatbaar, een fundamenteel lemma op:

Lemma 1 *Stel dat S een toelaatbare verzameling argumenten is en twee argumenten, a en a' zijn acceptabel met betrekking tot S; dan geldt:*

1. *$S' = S \cup \{a\}$ is toelaatbaar, en*
2. *a' is acceptabel met betrekking tot S'*

(Zie voor het (eenvoudige) bewijs [1, p. 327])

Lemma 1 is van van belang bij het vinden van geprefereerde extensies; het rechtvaardigt namelijk een manier om toelaatbare verzamelingen uit te breiden tot geprefereerde extensies.

De geprefereerde extensie/verzameling is afgeleid van de toelaatbare verzameling:

Definitie 16 *Een verzameling S is een* ***geprefereerde extensie*** *van een frame $F=<D,R>$ desda $S \subseteq D$ een $\subseteq$-maximale toelaatbare verzameling is.*

We hebben net laten zien dat er in het Nixon-Diamond voorbeeld twee toelaatbare verzamelingen zijn, {a} en {b}. Beide zijn $\subseteq$-maximaal toelaatbaar, omdat er voor geen van beide verzamelingen een verzameling S bestaat zodat $\{a\} \subset$ S of $\{b\} \subset$ S en S is toelaatbaar. Zowel {a} als {b} zijn dus geprefereerde extensies van het Nixon-Diamond frame.

Voor de rationele agent is de geprefereerde extensie een geschikte extensie om in debatten aan te houden als standpunt. Omdat deze extensie gedefinieerd is als een verzameling die aanvallers van zichzelf aanvalt kan de agent zijn standpunt altijd verdedigen.

Een tweede gangbare extensie in de argumentatietheorie is de stabiele extensie. Deze extensie kan ook goed gebruikt worden om een sterk standpunt te bepalen in debatten.

Definitie 17 *Een verzameling S is een* ***stabiele extensie*** *van een frame F=<D,R> desda S conflict-vrij is en alle argumenten uit de verzameling* $S^{-1} = D/S$ *aanvalt.*

Zoals Dung in [1, 328] ook opmerkt, is er een eenvoudig verband tussen stabiele en geprefereerde extensies:

Stelling 3 *Elke stabiele extensie is een geprefereerde extensie.*

Bewijs We moeten ten eerste aantonen dat elke stabiele extensie een toelaatbare verzameling is en ten tweede dat die verzameling $\subseteq$-maximaal toelaatbare is. Welnu, neem van een willekeurige frame F = <D,R> (dat een stabiele extensie heeft) een willekeurige stabiele extensie S; neem $S^{-1} = D/S$. Volgens definitie 17 is S conflictvrij, met andere woorden, elk argumenten dat S aanvalt, is geen element van S en is dus element van S^{-1}. Verder geldt volgens definitie 17 dat S alle elementen in S^{-1} aanvalt, dus ook ook alle aanvallers van elementen van S. S is dus toelaatbaar. Stel dat S niet maximaal toelaatbaar is. Dan is er een toelaatbare verzameling $T \supset S$; T is zeker conflict-vrij. Er geldt $T \subseteq D$, dus $T \cap S^{-1} \neq \emptyset$. Neem $a \in T \cap S^{-1}$. Er is een element van S, zeg b, dat a aanvalt. Nu geldt $b \to a$, $a \in T$ en $b \in T$, dus T is *niet* conflictvrij. *Tegenspraak!* Dus S is een maximaal toelaatbare verzameling c.q. een geprefereerde extensie. □

Het omgekeerde van stelling 3 geldt niet. De lege verzameling kan bijvoorbeeld wel een geprefereerde extensie zijn (de lege verzameling is altijd toelaatbaar), maar nooit een stabiele extensie, omdat het daarvoor nodig is dat alle argumenten uit het frame aangevallen worden door argumenten uit de lege verzameling; dat is duidelijk nooit het geval, omdat het domein van een frame niet leeg mag zijn en de lege verzameling geen argumenten bevat. De eigenschap van een verzameling dat alle argumenten buiten de verzameling worden aangevallen door die verzameling, wordt *compleetheid* genoemd. Omdat een stabiele extensie moet zowel conflict-vrij als compleet moet zijn, bestaat er niet op alle frames een stabiele extensie: in sommige gevallen heeft een frame een zodanige structuur dat het niet lukt alle argumenten buiten de extensie aan te vallen zonder dat ook argumenten binnen de extensie elkaar aanvallen. In de volgende paragraaf zal een noodzakelijke en voldoende voorwaarde voor het bestaan van de stabiele extensie gegeven worden.

2.4 Relaties tussen valuaties en extensies

In deze subparagraaf komt een aantal verbanden aan de orde tussen de stabiele en geprefereerde extensies enerzijds en de valuaties van argumenten anderzijds. Op basis van deze verbanden kan dan, nadat de argumenten uit een frame gevalueerd zijn, een groot deel van de extensies bepaald worden. Deze methode is een alternatief voor de methode om geprefereerde extensies te vinden, die Vreeswijk in [2] beschrijft. Daarover gaat paragraaf 3.

Stelling 4 en lemma's 2 en 3 beschrijven respectievelijk de relatie tussen onverslagen, verslagen en onbepaalde argumenten enerzijds en geprefereerde extensies anderzijds.

Stelling 4 *Als een argument a onverslagen is onder valuering f, bestaat er altijd een geprefereerde extensie onder f waartoe a behoort.*

Bewijs: Neem in een willekeurig frame F = <D,R> onder een willekeurige maximaal samenhangende valuatie een willekeurig onverslagen argument, zeg a. Het volstaat te bewijzen dat a deel uitmaakt van een toelaatbare verzameling S, want ofwel S is zelf maximaal toelaatbaar, ofwel er is een T $\supset$ S die maximaal toelaatbaar en dus geprefereerd is. Er zijn voor a twee gevallen mogelijk: (geval 1) a heeft geen aanvallers of (geval 2) a heeft verslagen aanvallers (onder f). In geval 1 is a acceptabel met betrekking tot de lege verzameling. De lege verzameling is toelaatbaar en volgens lemma 1 is $\emptyset \cup \{a\} = \{a\}$ dus ook toelaatbaar. In geval 2: noem de (onder f) onverslagen verdedigers van a resp. $a_1^1 ... a_m^1$. Noem de (onder f) onverslagen verdedigers van die verdedigers (indien ze er zijn) resp. $a_1^2 ... a_n^2$; ga zo door totdat alle te benoemen elementen reeds een naam hebben gekregen [4]. De verzameling $\{a, a_1^1, ..., a_m^1, a_1^2, ..., a_n^2, ...\}$ verdedigt elk van zijn elementen en is dus een toelaatbare verzameling. □

Voor elk onverslagen argument, in elke maximaal samenhangende valuatie van een frame bestaat er dus tenminste één geprefereerde extensie waartoe dat element behoort. De volgende lemma's beweren dat geprefereerde extensies nooit verslagen argumenten, noch onbepaalde argumenten kunnen bevatten.

Lemma 2 *Een verslagen argument behoort nooit tot een geprefereerde extensie.*

Bewijs: Neem in een willekeurig frame F = <D,R> onder een willekeurige maximaal samenhangende valuatie een willekeurig verslagen argument, zeg a. Stel a behoort tot een geprefereerde extensie, zeg S. Argument a is verslagen en heeft dus een onverslagen tegenstander, zeg b. Aangezien S geprefereerd is, moet er een verslagen argument $a_1 \in S$ zijn dat b aanvalt. Het kan niet zo zijn dat a_1=a, omdat a en b dan een geïsoleerde even loop zouden vormen en dientengevolge de valuatie van a onbekend zou zijn (volgens stelling 1). Deze a_1 moet op zijn beurt een verdedigend doch verslagen argument a_2 in S hebben. Er kan niet gelden a_2=a_1, noch a_2=a, omdat in beide gevallen een geïsoleerde even loop gevormd zou worden, en de valuatie van a onbekend zou zijn. Er is dus een verslagen verdediger van a_2 in S, etc. S moet oneindig veel verschillende verslagen argumenten bevatten, hetgeen in strijd is met de eis dat het domein van een argumentatie-frame een eindige verzameling argumenten is. Argument a behoort dus niet tot een geprefereerde extensie. □

Lemma 3 *Een onbepaald argument behoort nooit tot een geprefereerde extensie.*

Bewijs: Neem in een willekeurig frame F = <D,R> onder een willekeurige maximaal samenhangende valuatie een willekeurig onbepaald argument, zeg a. Stel dat a behoort tot een geprefereerde extensie, zeg S. Argument a heeft een aanvaller b met onbepaalde valuatie. Aangezien S geprefereerd is, heeft b een aanvaller a_1 in S die niet verslagen is (volgens lemma 2) en niet onverslagen kan zijn, omdat b dan verslagen zou zijn; dus is a_1 onbepaald. Het kan niet zo

[4] Loops in het frame zijn hierbij geen probleem: de te benoemen argumenten zullen geen deel uitmaken van oneven loops, omdat in dat geval de valuatie onbepaald zou zijn; in het geval het te benoemen element deel uitmaakt van een even loop, kom je bij de benoeming vroeg of laat weer bij hetzelfde element uit, omdat je met de benoeming steeds sprongen van twee aanvals-pijlen maakt.

zijn dat a_1=a, omdat a en b dan een geïsoleerde even loop zouden vormen en dientengevolge de valuatie van a onbekend zou zijn (volgens stelling 1). Deze a_1 moet op zijn beurt een verdedigend doch onbepaald argument a_2 in S hebben. Er kan niet gelden a_2=a_1, noch a_2=a, omdat in beide gevallen een geïsoleerde even loop gevormd zou worden, en de valuatie van a onbekend zou zijn. Er is dus een onbepaalde verdediger van a_2 in S, etc. S moet oneindig veel verschillende verslagen argumenten bevatten, hetgeen in strijd is met de eis dat het domein van een argumentatie-frame een eindige verzameling argumenten is. Argument a behoort dus niet tot een geprefereerde extensie. □

Nu geldt ook het volgende:

Lemma 4 *Slechts onverslagen argumenten kunnen behoren tot geprefereerde extensies.*

Lemma 4 volgt uit lemma's 2 en 3.

Met behulp van stelling 3 en kan hetzelfde voor stabiele extensies aangetoond worden:

Lemma 5 *Slechts onverslagen argumenten kunnen behoren tot stabiele extensies.*

Lemma 5 volgt direct uit lemma 4 en stelling 3.

Stelling 5 *Een frame F heeft een stabiele extensie desda F geen argumenten bevat met een onbepaalde valuatie.*

Bewijs:
⇒: Stel een stabiele extensie, bijvoorbeeld S, bestaat voor een frame F. Stel verder dat F wel een argument met onbepaalde valuatie bevat, zeg a. Argument a kan niet tot S behoren, volgens lemma 5; hieruit volgt uit met de definitie van stabiele extensies op bladzijde 10, dat a aangevallen wordt door S. Omdat S alleen onverslagen argumenten bevat (lemma 5), moet a wel verslagen zijn. Dat is in tegenspraak met de aanname dat a onbepaald is. F bevat dus geen onbepaalde argumenten.
⇐: Stel dat frame F geen onbepaalde argumenten bevat. F kan onder een gegeven valuatie, slechts onverslagen en verslagen argumenten bevatten. De verzameling van alle onverslagen argumenten is in dit geval de stabiele extensie. Die verzameling is conflict-vrij, omdat er geen aanvalsrelatie kan bestaan tussen twee onverslagen argumenten. De verzameling is ook compleet, omdat de argumenten die niet tot de extensie behoren allen verslagen zijn; een argument kan alleen verslagen zijn indien het wordt aangevallen door een onverslagen argument en alle onverslagen argumenten behoren tot de extensie. □

We bewijzen nu het complementaire stelling van stelling 3:

Stelling 6 *Indien een frame F een stabiele extensie heeft, zijn alle geprefereerde extensies stabiele extensies.*

Bewijs: We moeten, op grond van stelling 5, bewijzen dat elke geprefereerde extensie op een frame zonder onbepaalde argumenten compleet is. Daartoe zullen we aantonen dat elke toelaatbare extensie uit te breiden is tot een verzameling die zowel stabiel als geprefeerd is. Stel dat op een dergelijk frame F, S een willekeurige toelaatbare verzameling is. S bestaat alleen uit onverslagen argumenten, omdat S een deelverzameling van een geprefereerde extensie is (zie lemma 4). Stel dat F onverslagen argumenten, a_0, ..., a_n, bevat die niet tot S behoren. De verzameling $S' = S \cup \{a_0, ..., a_n\}$ is conflict-vrij, want zoniet, dan zouden niet alle elementen van S′ onverslagen zijn. S′ is $\subseteq$-maximaal toelaatbaar, want de verzameling is niet uit te breiden zonder verslagen argumenten toe te voegen (en zijn niet acceptabel m.b.t. S′). S′ is dus geprefereerd. Verder is S′ ook compleet, want alle argumenten buiten $S \cup \{a_0, ..., a_n\}$ zijn verslagen en worden dus aangevallen door onverslagen argumenten, en die zijn alle bevat in $S \cup \{a_0, ..., a_n\}$. S′ is dus ook stabiel. □

De geprefereerde extensies vallen dus samen met de stabiele extensies op frames die geen onbepaalde argumenten bevatten.

We kunnen ook iets zeggen over het aantal stabiele en geprefereerde extensies dat een frame heeft. Het aantal geprefereerde extensies is direct verbonden met het aantal maximaal samenhangende valuaties. Het aantal stabiele extensies is direct verbonden met het aantal maximaal samenhangende valuaties dat is gedefinieerd op het gehele domein van een frame. We definiëren de eigenschap 'volledigheid' van een valuatie-functie dus als volgt:

Definitie 18a *Een valuatie-functie* $f{:}X \longrightarrow \{verslagen, onverslagen\}$ *op een frame* $F = <D,R>$ *is* ***volledig*** *desda* $X = D$.

Definitie 18b *Een valuatie-functie* $f{:}X \longrightarrow \{verslagen, onverslagen\}$ *op een frame* $F = <D,R>$ *is* ***onvolledig*** *desda* $X \subset D$.

We bekijken eerst het eenvoudigste geval: stel dat het frame geen maximaal samenhangende valuaties heeft. Hieruit volgt direct dat het frame slechts onbepaalde argumenten bevat. In dit geval heeft het frame geen stabiele extensie (volgens stelling 5). De geprefereerde extensie kan geen onbepaalde argumenten bevatten, maar kan wel leeg zijn, dus heeft het frame in dit geval de lege verzameling als enige extensie.

Een andere mogelijkheid is dat het frame wel onvolledige maximaal samenhangende valuaties heeft, maar geen volledige. In dat geval zijn er dus zowel onbepaalde als niet-onbepaalde argumenten in het frame. Er is wederom geen stabiele extensie (stelling 5), maar er zijn wel geprefereerde extensies. Met behulp van lemma 6 is het makkelijk te laten zien dat een frame evenveel geprefereerde extensies als maximaal samenhangende valuaties heeft (mits er maximaal samenhangende valuaties van het frame zijn).

Lemma 6 *In een frame F met een maximaal samenhangende valuatie f is de verzameling van alle onverslagen argumenten (onder f) een geprefereerde extensie van F onder f en tevens de enige geprefereerde extensie onder f.*

Bewijs: De verzameling van alle onverslagen argumenten, zeg S, is geprefereerd, want S is ten eerste conflict-vrij (een onverslagen argument wordt niet aangevallen door andere onverslagen argumenten) en ten tweede geldt dat alle aanvallers van elementen van S verslagen zijn en dus tenminste één onverslagen aanvaller hebben (die tot S behoort); S is dus ook toelaatbaar. S is duidelijk maximaal toelaatbaar want er zijn geen acceptabele argumenten meer met betrekking tot S. S is dus geprefereerd. Elke andere toelaatbare verzameling T bevat enkel onverslagen argumenten (dat volgt uit lemma 4) en is dus geldt T $\subseteq$ S. T is dus niet maximaal toelaatbaar. □

Stelling 7 *Indien een frame F maximaal samenhangende valuaties heeft, zijn er evenveel geprefereerde extensies als maximaal samenhangende valuaties.*

Bewijs: Uit lemma 6 volgt dat er tenminste evenveel geprefereerde extensies als maximaal samenhangende valuaties zijn. Er zijn ook tenminste evenveel maximaal samenhangende valuaties als geprefereerde extensies; voor een gegeven geprefereerde extensie S kan namelijk als volgt een maximaal samenhangende valuatie f gevonden worden: geef alle argumenten die tot S behoren de waarde onverslagen en alle argumenten die door S worden aangevallen de waarde verslagen. De onverslagen argumenten hebben nu geen onverslagen tegenstanders, omdat S conflict-vrij is. Ook geldt dat alle verslagen argumenten tenminste één onverslagen aanvaller hebben. Valuatie f is dus samenhangend. Stel dat f niet de grootste samenhangende valuatie is; er zijn dan onverslagen argumenten die niet tot S behoren. Noem de verzameling die precies deze argumenten bevat T. De verzameling S $\cup$ T bevat alle onverslagen argumenten, en dus zijn alle aanvallers van S $\cup$ T verslagen; deze verslagen argumenten worden noodzakelijkerwijs aangevallen door S $\cup$ T; S $\cup$ T is dus toelaatbaar, waaruit volgt dat S niet maximaal toelaatbaar is. Dat is in tegenspraak met de aanname dat S een geprefereerde extensie is. Valuatie f is dus een maximaal samenhangende valuatie. □

Voor stabiele extensies geldt een overeenkomstig verband, maar dan in relatie tot volledige maximaal samenhangende valuaties. Om dit aan te tonen is lemma 7 nodig.

Lemma 7 *In een frame F met een volledige maximaal samenhangende valuatie f is de verzameling van alle onverslagen argumenten een stabiele extensie van F onder f en tevens de enige onder f.*

Bewijs: De verzameling van alle onverslagen argumenten, zeg S, onder f, is maximaal toelaatbaar (zie het bewijs van lemma 6). S is dus conflict-vrij en omdat f volledig is, bevat het frame geen argumenten met onbepaalde valuatie. Omdat alle onverslagen argumenten in S zitten, zijn alle resterende argumenten verslagen. De argumenten worden aangevallen door onverslagen argumenten en dus door S. S is dus ook compleet. Hieruit volgt dat S een stabiele extensie is. S is ook de enige stabiele extensie is onder valuatie f want S is tevens een geprefereerde extensie, volgens stelling 3, en tevens de enige, volgens lemma 6. Aangezien elke stabiele extensie een geprefereerde extensie is, zijn alle stabiele extensies identiek met S. □

Stelling 8 *Een frame F heeft evenveel stabiele extensies als volledige maximaal samenhangende valuaties.*

Voor elke volledige maximaal samenhangende valuatie van het frame is er precies één stabiele extensie volgens lemma 7. Er zijn geen andere stabiele extensies, omdat die zouden moeten gelden onder een onvolledige valuatie. De onvolledigheid van de valuatie impliceert de aanwezigheid van onbepaalde argumenten en in dat geval is er geen stabiele extensie, volgens stelling 5. Dat impliceert dat een frame precies evenveel stabiele extensies heeft als volledige maximaal samenhangende valuaties.

3 Implementatie van de valuering en het aanmaken van extensies

Het valueren van argumenten is niet altijd eenvoudig. Tot op zekere hoogte is het weliswaar gemakkelijk, maar het wordt gecompliceerd indien de valuatie van argumenten onbekend of onbepaald is. Om die argumenten te kunnen valueren is het handig te weten welke argumenten in geïsoleerde loops voorkomen. Dat is in het geval van kleine frames in een (goed opgestelde) grafische weergave makkelijk te zien, maar geïsoleerde loops detecteren is een taak van aanzienlijke complexiteit. In ons algoritme worden de loops dan ook niet gedetecteerd, maar de valuatie geschiedt in eerste instantie voorzover die volgens een eenvoudig algoritme kan worden uitgevoerd. Dit algoritme is niet in alle gevallen in staat elk argument uit het frame te valueren. Daarom worden de resterende, ongevalueerde argumenten op een andere (omslachtiger) manier gevalueerd. Hierbij wordt overigens opnieuw van het algoritme gebruik gemaakt. Nadat op deze manier van alle argumenten de valuatie is vastgesteld (er kunnen meerdere maximaal samenhangende valuaties zijn gegenereerd), kunnen de geprefereerde en stabiele extensies eenvoudig worden gegenereerd, dankzij de stellingen uit paragraaf 2.4. We zullen de manier van aanpak laten zien aan de hand van een aantal voorbeelden.

3.1 Het basis-algoritme

Allereerst bekijken we nog eens het reeds besproken frame in figuur 2 op bladzijde 4. Uit definitie 4a volgt direct dat argument a, omdat het niet wordt aangevallen, onverslagen is. Argumenten b en e worden aangevallen door a en zijn daarom, volgens definitie 4b, verslagen. De argumenten c en d vallen elkaar aan en verdedigen zodoende zichzelf tegen de aanval van de ander. Argument c wordt ook nog aangevallen door b, maar dit argument is verslagen en daarom is de valuatie van c en d in dit voorbeeld onbekend [5].

Argumenten zoals a, b en e, die noch deel uitmaken van geïsoleerde loops, noch enkel vanuit geïsoleerde loops aangevallen worden, kunnen dus gemakkelijk gevalueerd worden. We geven nu een basis-algoritme, dat in staat is om dergelijke argumenten te valueren

stap 1 vind de argumenten die geen aanvallers hebben, deze argumenten zijn onverslagen; worden in deze stap geen argumenten gevonden, stop dan

[5] Was b echter onverslagen geweest, dan was c verslagen geweest en lag zodoende de valuatie van d ook vast. In dat geval waren er geen argumenten met onbekende valuatie geweest

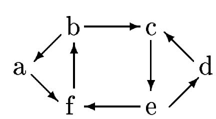
stap 2 vind de argumenten die worden aangevallen door onverslagen argumenten, deze zijn verslagen; worden in deze stap geen argumenten gevonden, stop dan

stap 3 negeer verslagen argumenten als aanvallers en ga weer naar stap 1

In het voorbeeld van figuur 2 wordt in de eerste stap argument a als onverslagen gekwalificeerd. In stap 2 worden b en e als verslagen argumenten gevonden, in stap 3 wordt b niet meegeteld als aanvaller van c. In stap 1 worden vervolgens geen argumenten zonder aanvallers meer gevonden (a was al gevalueerd en telt nu dus niet mee), waardoor de argumenten c en d overblijven na beëindiging van het algoritme.

3.1.1 Ontoereikendheid van het basis-algoritme

Zoals ook in dit voorbeeld, blijft in sommige gevallen na het doorlopen van dit algoritme een aantal argumenten ongevalueerd; deze argumenten zullen we hierna 'resterende argumenten' noemen. De resterende argumenten hebben gemeenschappelijk dat ze worden aangevallen door argumenten die deel uitmaken van een loop (daaronder vallen in ieder geval alle argumenten die zelf tot een loop behoren). Men zou vermoeden dat alleen argumenten die aangevallen worden vanuit geïsoleerde loops overblijven, aangezien de argumenten van niet-geïsoleerde loops allen verslagen dan wel onverslagen zijn. Toch zijn er ook niet-geïsoleerde loops te construeren die niet zonder meer met het valuerings-algoritme te valueren zijn. Een treffend voorbeeld van een frame dat dergelijke loops bevat is het frame in figuur 5.



Figuur 5: Frame met niet-geïsoleerde loops

Het lijkt hier alsof de twee oneven loops, bestaande uit respectievelijk a, b, f en c, d, e, beide geïsoleerde loops zijn; er zijn weliswaar argumenten die de loops aanvallen (e valt de linker loop aan en b de rechter), maar deze argumenten lijken niet direct onverslagen. Indien de valuatie van b en e onbepaald zou zijn, was er geen maximaal samenhangende valuatie van de argumenten mogelijk waarin b of e onverslagen is. Toch is zo'n valuatie wel mogelijk. De valuatie waarin b en e onverslagen zijn en de andere argumenten, a, c, d en f, verslagen, is maximaal samenhangend. Wanneer er ook een valuatie zou zijn waarin b en e verslagen zijn, zou de valuatie van b en e onbekend zijn, maar een dergelijke valuatie is niet maximaal samenhangend. In dat geval zouden de beide oneven loops geïsoleerd zijn en dientengevolge de loop-elementen (waaronder b en e) onbepaald, hetgeen in tegenspraak is met de aanname dat b en e verslagen zijn. Dit impliceert dat b en e onverslagen zijn en daarom zijn de beide oneven loops niet-geïsoleerd.

Het is makkelijk in te zien dat met het huidige valuerings-algoritme geen valuatie van de argumenten gevonden kan worden; geen enkel argument heeft immers geen aanvallers. Om in dergelijke gevallen het algoritme toch te kunnen gebruiken, kunnen we een hypothese over de valuatie van een argument stellen. Vanuit die hypothese kan dan het valuerings-algoritme toegepast worden om de valuatie van de andere argumenten te berekenen. Deze procedure kan op elk van de resterende argumenten worden toegepast en zo kan worden bekeken of er een maximaal samenhangende valuatie mogelijk is.

Wij stellen slechts de hypothese van een onverslagen argument. De situatie waarin het argument verslagen is komt overeen met een situatie waarin een van de aanvallers van het argument onverslagen is. Als er meerdere aanvallers zijn zijn er dus ook meerdere situaties waarin het argument verslagen is. Verder gebruiken wij slechts het basis-algoritme om implicaties van de hypothese te bepalen. Hierdoor kunnen slechts implicaties voor argumenten die in richting van de aanvalsrelaties liggen worden bepaald.

Hier volgt een beknopte weergave van de werking van het valuerings algoritme en ter illustratie wordt elke stap toegepast op argumentatie-frame uit figuur 5 en figuur 6.

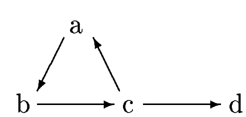
Als eerste wordt voor elk apart argument bekeken wat de implicaties zijn van de hypothese dat het argument onverslagen is, wanneer onder deze hypothese het basis-algoritme wordt toegepast. Zo is in figuur 5 onder de hypothese dat a onverslagen is f verslagen, b onverslagen en a verslagen.

Als een argument een onbepaalde valuatie heeft is het noodzakelijk dat kan worden afgeleid door middel van het basisalgoritme dat onder de hypothese van een onverslagen valuatie de valuatie van het argument eigelijk verslagen behoort te zijn. Dit is bijvoorbeeld het geval voor argument c uit figuur 6 en argument a uit figuur 5. Helaas is de noodzakelijke voorwaarde geen voldoende voorwaarde; een argument heeft ook een onbepaalde valuatie als de hypothese van het verslagen argument impliceert dat het argument eigenlijk onverslagen behoort te zijn. Wij veronderstellen voor de eenvoud van het programma alleen onverslagen valuaties van argumenten. Willen we elke mogelijke situatie vinden waarin het argument verslagen is, dan moeten we voor elke aanvaller stellen dat deze onverslagen is. Dit impliceert automatisch dat het ter discussie staande argument verslagen is. Als in elk van deze situaties blijkt dat de aanvaller onverslagen is onder de hypothese dat de aanvaller verslagen is kan werkelijk worden geconcludeerd dat het oorspronkelijke argument een onbepaalde valuatie heeft. Argument c in figuur 5 heeft argument b als aanvaller. Als argument b onverslagen is, is c verslagen, a onverslagen en b verslagen. Uit deze tegenspraak kan worden afgeleidt dat argument een onbepaalde valuatie heeft. Argument a in figuur 5 heeft echter niet een onbepaalde valuatie aangezien onder de hypothese dat b onverslagen is geen tegenspraak kan worden afgeleidt.

Als een argument een onbepaalde valuatie heeft hebben alle argumenten die worden aangevallen door dit argument ook een onbepaalde valuatie, dit volgt direct uit de definitie van onverslagen en verslagen. Argumenten die worden aangevallen door argumenten die worden aangevallen door het argument met de onbepaalde valuatie hebben dus ook een onbepaalde valuatie. Dus alle argumenten die zijn verbonden door aanvalsrelaties die allen georienteerd zijn richting deze argumenten met een argument met een onbepaalde valuatie hebben dus ook een onbepaalde valuatie. Uit figuur 6 blijkt dan direct dat alle argumenten in dit figuur een onbepaalde valuatie hebben. In figuur 5 heeft geen van de argumenten een onbepaalde valuatie.

Als alle argumenten die een onbepaalde valuatie hebben zijn bepaald kunnen de maximaal samenhangende valuaties worden bepaald. Een maximaal samenhangende valuatie komt overeen met een combinatie van hypothesen waarbij in deze combinatie de implicaties van deze hypothese consistent met elkaar zijn. Binnen figuur 5 zou dit ongeveer alsvolgt kunnen gaan: De hypothese dat argument a onverslagen is is inconsistent met zichzelf, de hypothese van het verslagen argument a door te stellen dat argument b onverslagen is geeft in eerste instantie geen inconsistentie. Deze wordt gecombineerd met de implicatie van de hypothese dat argument b onverslagen is, c verslagen, d verslagen, e onverslagen en verslagen. Aangezien slechts van deze combinatie de implicaties van de verschillende hypotheses consistent met elkaar zijn, de implicaties zijn compatible.

Een mogelijk probleem zou kunnen worden dat er bij het samenvoegen van de hypotheses de inconsistentie als gevolg van de combinatie van hypothese niet wordt gevonden aangezien deze niet wordt afgeleid, omdat het basis-algoritme slechts een gedeelte van alle implicaties afleidt. Gelukkig wordt door middel van het algoritme elke inconsistentie afgeleidt. Als er een inconsistentie is, is een argument zowel verslagen als onverslagen. Deze inconsistentie wordt gevonden aangezien ten eerste voor dit argument wordt gesteld dat hij onverslagen is en voor de aanvaller van het argument dat hij onverslagen is en dus volgt dat het argument dat een problematische valuatie heeft verslagen is. Het problematische argument heeft altijd een aanvaller aangezien hij anders al in eerste instantie als onverslagen was gevaluaeerd.



Figuur 6: Frame met slechts onbepaalde argumenten

Wanneer er maar één maximaal samenhangende valuatie gevonden is, is de valuatie van alle argumenten conform de maximaal samenhangende valuatie. Wanneer er meerdere maximaal samenhangende valuaties gevonden zijn, duidt dat op de aanwezigheid van argumenten met een onbekende valuatie. Het identificeren van die argumenten is eenvoudig: het zijn precies die argumenten die in tenminste één van de maximaal samenhangende valuaties onverslagen zijn en in tenminste één verslagen. De argumenten die in elke maximaal samenhangende valuatie dezelfde valuatie hebben, zijn vanzelfsprekend niet onbekend. Dan is er ook nog de mogelijkheid dat er wel maximaal samenhangende valuaties gevonden worden, maar dat er resterende argumenten zijn die nooit als verslagen en nooit als onverslagen worden gevalueerd in die valuaties. Deze argumenten zijn klaarblijkelijk onbepaald.

3.2 Het aanmaken van de extensies

Nadat alle argumenten gevalueerd zijn, kunnen de geprefereerde extensies en, zo ze bestaan, de stabiele extensies worden aangemaakt met behulp van de stellingen die verbanden tussen valuatie en extensies beschrijven (in paragraaf 2.4).

Als eindresultaat van het valuerings-algoritme wordt elke maximaal samenhangende valuatie gegeven. Vanuit dit resultaat kunnen de verschillende extensie worden aangemaakt. De verzameling van alle onverslagen argumenten in een maximaal samenhangende valuatie vormt een geprefereerde extensie en vormt tevens een stabiele extensie als de valuatie volledig is. De gegronde (Engels: *grounded*) extensie is de verzameling van argumenten die waarvoor geldt dat de valuatie van deze argumenten in elke maximaal samenhangende valuatie onverslagen is.

4 Toepassing van het programma

We zullen nu illustreren hoe de resultaten van het programma in debatten geïnterpreteerd kunnen worden. Daartoe passen we eerst het programma toe op het debat-voorbeeld over het broeikas effect, uit de inleiding. Het frame is afgebeeld in figuur 1 op bladzijde 2. Hier kan direct het valuerings-algoritme worden toegepast zonder hypothesen. Na beëindiging van het algoritme zijn alle argumenten gevalueerd en zijn er geen resterende argumenten. Er is dus precies één volledige maximaal samenhangende valuatie. In deze valuatie is {b,a} de verzameling verslagen argumenten en {c,d} de verzameling onverslagen argumenten. Er is dus een stabiele extensie en die is identiek met de geprefereerde extensie, de verzameling {c,d}. In het debat zou het best te verdedigen standpunt dus uit twee argumenten bestaan, namelijk: 'de stijgingssnelheid van de temperatuur is te groot om veroorzaakt te worden door klimatologische schommelingen' en 'het broeikas-effect is gewoon een verzinsel van de media'.

Dit voorbeeld toont een onverwacht aspect van het gebruik van geprefereerde extensies als manier om standpunten te bepalen: de standpunten die op deze manier bepaald worden kunnen weliswaar alleen reeds bestaande argumenten bevatten, waardoor het lijkt alsof elk geprefereerd standpunt 'achter de feiten aanloopt'; maar uit dit voorbeeld blijkt dat er door de combinatie van argumenten toch een aanzet gegeven kan worden tot de formulering van een geheel nieuw standpunt. In het voorbeeld suggereert het geprefereerde standpunt duidelijk de mogelijkheid van een derde mogelijke verklaring, naast de klimatologische schommelingen en het broeikas-effect.

Deze progressieve neiging is natuurlijk te herleiden tot eigenschappen van de geprefereerde extensie. De geprefereerde extensie bevat de argumenten die onverslagen zijn, ongeacht de intentie waarmee deze argumenten naar voren zijn gebracht. Van de argumenten die de geprefereerde extensie bevat was er één bedoeld ter confirmatie van het bestaan van het broeikas-effect en één ter ontkrachting ervan. Omdat de argumenten zelf niet in tegenspraak zijn met elkaar, kunnen ze toch verenigd worden in de geprefereerde extensie.

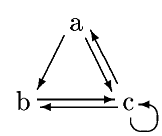
Het valt zeer te bezien of alle besproken frames uit de vorige paragrafen reële debat-situaties representeren. In werkelijkheid zijn frames die echte discussies en debatten representeren vaak minder gecompliceerd en vaak heeft het frame een boomstructuur: er wordt een stelling (de wortel van de boom) verdedigd en aanvallers van die stelling vallen ofwel de stelling zelf aan, ofwel de verdedigende argumenten van die stelling (de takken van de boom). Toch zijn er paradoxale debatten te vinden die je min of meer kunt representeren als een frame met oneven loops. We zullen hiervan een (enigszins gekunsteld) voorbeeld geven:

Ail: 'Immigranten moeten de Nederlandse cultuur overnemen, want behoud van eigen cultuur leidt tot isolement en is dus een teken van slechte integratie, terwijl ze juist moeten integreren.' (a)

Lia: 'Immigranten moeten wel integreren, maar het recht op behoud van eigen cultuur is een aspect van de Nederlandse cultuur en daarom is behoud van eigen cultuur een teken van goede integratie. Immigranten moeten dus hun eigen cultuur behouden.' (b)

Ali: 'Immigranten moeten integreren, maar het behouden van eigen cultuur en daarmee ook het overnemen van de Nederlandse cultuur (wat betreft behoud van eigen cultuur) leidt tot isolement. Immigranten moeten dus hun cultuur niet behouden en niet de Nederlandse cultuur overnemen.' (c)

Ails stelling is dat immigranten de Nederlandse cultuur moeten overnemen en verdedigt die met het argument dat behoud van eigen cultuur een teken van slechte integratie is. Dit argument wordt aangevallen door Lia's argument dat behoud van eigen cultuur een typisch Nederlands verschijnsel is. Zij is daarom van mening dat immigranten hun eigen cultuur moeten behouden. Ali wordt de dupe van Lia's paradoxale inzicht en concludeert dat zowel het behouden van eigen cultuur als het overnemen van de Nederlandse cultuur tot isolement leidt. Immigranten moeten volgens haar dus geen van beide doen. Met dit argument valt ze zowel Ail, als Lia, als zichzelf aan. De argumenten van Ail en Lia vallen beide op hun beurt Ali's argument aan. In een frame ziet dit debat er als volgt uit:

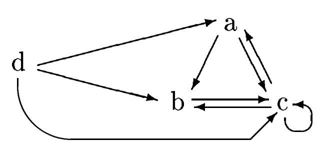


Figuur 7: Frame van het integratie-debat

Er zijn geen argumenten zonder aanvallers, dus wanneer we dit frame proberen te valueren met onze implementatie, moeten we direct beginnen met het uitvoeren van het algoritme onder de hypothetische valuatie van een argument. Wanneer we a als onverslagen nemen, leiden we af dat b en c verslagen zijn; op dit punt kan er geen stap meer uitgevoerd worden en is er een maximaal samenhangende valuatie gevonden met als verzameling onverslagen argumenten {a} en als verzameling verslagen argumenten {b,c}. We zoeken verder naar andere valuaties door b onverslagen te nemen; dat levert in de volgende stap c verslagen; daaruit volgt a onverslagen en vervolgens b verslagen. Er is ook geen valuatie met b onverslagen. Argument c valt zichzelf aan dus de hypothese dat c onverslagen is, leidt in één stap tot een tegenspraak. De enige maximaal samenhangende valuatie van het frame is dus die met a onverslagen en b en c verslagen. Deze valuatie is volledig, dus is er een stabiele extensie, namelijk {a}. Deze verzameling is ook gelijk de (enige) geprefereerde extensie. Uit het feit dat de argumenten niet onbepaald zijn, volgt tevens dat deze loop niet geïsoleerd is. Hieruit blijkt dat het niet altijd in één oogopslag te zien is of een loop geïsoleerd is.

Wanneer we de extensies terugvertalen naar het debat, zou Ail de enige houdbare positie innemen. Haar posistie is stabiel, omdat haar argument beide andere argumenten aanvalt en geen tegenspraak met zichzelf vormt. Haar positie is ook geprefereerd, omdat het de grootst mogelijke verzameling argumenten is die niet in tegenspraak is met zichzelf en al zijn elementen verdedigt.

Ten slotte willen we nog een argumentatie-theoretische evaluatie geven van de tegenwoordig alom gehoorde dooddoener: 'Waar zijn we nou helemaal mee bezig?'. Een dergelijke opmerking in een discussie is vaak bedoeld om de gehele discussie te relativeren. Ons inziens kan deze opmerking in termen van frames het beste geïnterpreteerd worden als een argument dat alle andere argumenten aanvalt. In het bovengenoemde integratie-debat zou de opmerking 'waar zijn we nou helemaal mee bezig ?' (d) (die het meest waarschijnlijk door Lia of Ali geplaatst zal worden) resulteren in het frame dat is afgebeeld in figuur 8.



Figuur 8: Frame van het integratie-debat

Zolang niemand de spelbreker gebrek aan coöperativiteit (of iets anders) verwijt, is het argument onverslagen en is de valuatie van het frame uiterst eenvoudig: alle argumenten zijn verslagen behalve d. De geprefeerde en stabiele extensie zijn beide de verzameling {d}. In tegenstelling tot de geprefereerde extensie in het broeikas-debat, biedt deze extensie geen uitzicht op nieuwe standpunten. Een ander gevolg van de opmerking is dat het formeel gezien geen zin heeft argumenten tegen de oorspronkelijke argumenten (a, b en c) te vinden zolang d niet verslagen is; hiermee krijgt het begrip 'dooddoener' een mogelijke interpretatie in argumentatie-frames.

A Broncode van het Prolog-programma

De broncode kan ook worden ge-download, vanaf de URL:
http://tcw2.ppsw.rug.nl/ faine/project/val.pl

```
framework(1,[[d]/a,[a,e]/b,[b]/c,[c]/d,[a]/e]).
framework(2,[[b]/a,[c]/b,[a]/c]).
framework(3,[[c,d,h]/a,[d,e]/b,[p,m]/c,[c,g]/d,[d,h,i]/e,[n]/f,
[]/g,[]/h,[j]/i,[i]/j,[m]/k,[k,p]/l,[l]/m,[i,j]/n,[g,h,n]/p,
[p]/q,[z]/x,[x]/z]).
framework(4,[[b,d]/a,[c]/b,[]/c,[c]/d]).
framework(5,[[d]/a,[a]/b,[b]/c,[c]/d]).
framework(6,[[b]/a,[f]/b,[b,d]/c,[e]/d,[c]/e,[a,e]/f]).
framework(7,[[b,c]/a,[d]/b,[e]/c,[f]/d,[g]/e,[d]/f,[e]/g]).

prefex(Frame,Pref,UnPref):-
valuation(Frame,Undef1,Def1,_,Complete,0),
pick(Def2/Undef2,Complete),
append(Def2,Def1,UnPref),
```

```
append(Undef2,Undef1,Pref)
 .
stable(Frame,Stable,UnStable):-
(
    valuation(Frame,Undef1,Def1,[],Complete,0)
->
pick(Def2/Undef2,Complete),
append(Def2,Def1,UnStable),
append(Undef2,Undef1,Stable)
;
    write('No Stable Extension'),nl
)
 .

grounded(Frame,Grounded):-
(
    valuation(Frame,Undef1,_,_,[_/Undef3|Complete],0),
    get_undef_out_complete(Undef2,Undef3,Complete),
    append(Undef2,Undef1,Grounded)
;
    valuation(Frame,Grounded,_,_,[],0)
)
.

get_undef_out_complete(Undef1,Undef2,[Def/_|Complete]):-
get_out(Def,Undef2,Undef3),
get_undef_out_complete(Undef1,Undef3,Complete)
 .
get_undef_out_complete(Undef,Undef,[]).

get_out([H|T1],[H|T2],L):-
!,
get_out(T1,T2,L)
 .
get_out([H1|T1],[H2|T2],L1):-
name(H1,[H1Val]),
name(H2,[H2Val]),
(
    H1Val < H2Val
->
get_out(T1,[H2|T2],L1)
;
    L1 = [H2|L2],
    get_out([H1|T1],T2,L2)
)
 .
get_out([],L,L):-!.
get_out(_,[],[]).

%%%  valuation  %%%
```

```
valuation(Frame,Undefeated,Defeated,Undecided,Complete,PO):-
framework(Frame,AttRel),
first_valuation([],AttRel,Undefeated,Defeated,Rest),
hypo_valuation(Rest,Rest,QuatroS),
make_undecided(QuatroS,QuatroS,NewQuatroS,Undecided),
findall(Pair,complete_valuation(NewQuatroS,NewQuatroS,[]/[],Pair),Complete),
(
    PO == 1
->
pretty_print(AttRel,Undefeated,Defeated,Undecided,Complete)
;
    true
)
 .

%%% first_valuation %%%

first_valuation(Def,AttRel,U1,D1,Rest):-
make_undef(UnDef,Def,AttRel,NewAttRel),
(
    UnDef == []
->
    U1 = [],
    D1 = Def,
    Rest = NewAttRel
;
    make_def(NewDef,UnDef,NewAttRel,NewNewAttRel),
    first_valuation(NewDef,NewNewAttRel,U2,D2,Rest),
    append(UnDef,U2,U1),
    append(Def,D2,D1)
)
 .

make_def([Arg|NewDef],UnDef,[AttS/Arg|AttRel],NewAttRel):-
list_member(UnDef,AttS),
!,
make_def(NewDef,UnDef,AttRel,NewAttRel)
 .
make_def(NewDef,UnDef,[AttRel|AttRelS],[AttRel|NewAttRel]):-
make_def(NewDef,UnDef,AttRelS,NewAttRel)
 .
make_def([],_,[],[]).


list_member([UnDef|_],[UnDef|_]):-!.
list_member([UnDef|UnDefS],[Att|AttS]):-
(
    name(UnDef,[UnDefVal]),
    name(Att,[AttVal]),
```

```
    UnDefVal > AttVal,
    !,
    list_member([UnDef|UnDefS],AttS)
;
    list_member(UnDefS,[Att|AttS])
)
  .


make_undef(NewUnDef,[Def|DefS],AttRel,NewAttRel):-
!,
remove_def(Def,AttRel,BetAttRel),
make_undef(NewUnDef,DefS,BetAttRel,NewAttRel)
  .
make_undef([Arg|NewUnDef],[],[[]/Arg|AttRel],NewAttRel):-
!,
make_undef(NewUnDef,[],AttRel,NewAttRel)
  .
make_undef(NewUnDef,[],[AttPair|AttRel],[AttPair|NewAttRel]):-
!,
make_undef(NewUnDef,[],AttRel,NewAttRel)
  .
make_undef([],[],[],[]).


remove_def(Def,[AttS/Arg|AttRel],[AttSMinDef/Arg|BetAttRel]):-
!,
remove(Def,AttS,AttSMinDef),
remove_def(Def,AttRel,BetAttRel)
  .
remove_def(_,[],[]).


remove(Def,[Def|AttS],AttS):-!.
remove(Def,[Att|AttS],[Att|AttSMinDef]):-
!,
(
    name(Def,[DefVal]),
    name(Att,[AttVal]),
    DefVal > AttVal,
    !,
    remove(Def,AttS,AttSMinDef)
;
    AttS = AttSMinDef
)
  .
remove(_,[],[]).

%%% end firstvaluation %%%
```

```
%%% hypo_valuation %%%

hypo_valuation(AttRelS,[Atts/Arg|AttRel],[Def/UnDef1/Atts/Arg|QuatroS]):-
hypo_undef(Arg,AttRelS,Def,UnDef2),
append([Arg],UnDef2,UnDef1),
hypo_valuation(AttRelS,AttRel,QuatroS)
 .

hypo_valuation(_,[],[]).

hypo_undef(Arg,AttRelS,Def,UnDef):-
make_def(NewDef,[Arg],AttRelS,NewAttRelS),
first_valuation(NewDef,NewAttRelS,UnDef,Def,_)
 .
%%% end hypo_valuation %%%


%%% make_undecided %%%

make_undecided(QuatroS1,[Def/_/Atts/Arg|_],EndQuatroS,Undecided1):-
list_member([Arg],Def),
check_out_att(QuatroS1,Atts),
!,
find_undecided(Arg,[Arg],QuatroS1,QuatroS3,Undecided2),
find_tail_and_get_out(Arg,QuatroS3,QuatroS5,QuatroS4),
make_undecided(QuatroS5,QuatroS4,EndQuatroS,Undecided3),
append(Undecided2,Undecided3,Undecided4),
append([Arg],Undecided4,Undecided1)
 .
make_undecided(QuatroS1,[_|QuatroS2],EndNewQuatroS,Undecided):-
make_undecided(QuatroS1,QuatroS2,EndNewQuatroS,Undecided)
 .
make_undecided(EndNewQuatroS,[],EndNewQuatroS,[]).

check_out_att(QuatroS,[Arg|Atts]):-
find_arg(Def/_/_/Arg,QuatroS),
list_member([Arg],Def),
check_out_att(QuatroS,Atts)
 .
check_out_att(_,[]).
%%% find_undecided %%%

find_undecided(_,[],EndQuatroS,EndQuatroS,[]):-!.
find_undecided(Arg,UDMakers,QuatroS,EndQuatroS1,Undecided1):-
do_it(Arg,UDMakers,QuatroS,EndQuatroS2,Undecided2),
find_undecided(Arg,Undecided2,EndQuatroS2,EndQuatroS1,Undecided3),
append(Undecided2,Undecided3,Undecided1)
 .
do_it(Arg,UDMakerS,[_/_/Atts/Arg2|QuatroS1],QuatroS2,[Arg2|Undecided]):-
Arg \== Arg2,
```

```
list_member(UDMakerS,Atts),
!,
do_it(Arg,UDMakerS,QuatroS1,QuatroS2,Undecided)
 .
do_it(Arg,UDMakerS,[Quatro|QuotraS1],[Quatro|QuatroS2],Undecided):-
do_it(Arg,UDMakerS,QuotraS1,QuatroS2,Undecided)
 .
do_it(_,_,[],[],[]).

find_tail_and_get_out(Arg,[_/_/_/Arg|QuatroS],QuatroS,QuatroS):-!.
find_tail_and_get_out(Arg,[Quatro|QuatroS1],[Quatro|QuatroS3],QuatroS2):-
find_tail_and_get_out(Arg,QuatroS1,QuatroS3,QuatroS2)
 .
%%% end find_undecided %%%

%%% end make_undecided %%%

%%% complete_valuation %%%

complete_valuation(QuatroS1,[Def1/UnDef1/AttS/_|QuatroS2],Def2/UnDef2,Pair):-
(
    consistent(Def1/UnDef1,Def2/UnDef2),
    append(Def1,Def2,Def3),
    append(UnDef1,UnDef2,UnDef3)
;
    pick(Arg2,AttS),
    find_arg(Def4/UnDef4/_/Arg2,QuatroS1),
    consistent(Def4/UnDef4,Def2/UnDef2),
    append(Def4,Def2,Def3),
    append(UnDef4,UnDef2,UnDef3)
),
complete_valuation(QuatroS1,QuatroS2,Def3/UnDef3,Pair)
 .

complete_valuation(_,[],Pair,Pair).

find_arg(Def/UnDef/_/Arg,[Def/UnDef/_/Arg|_]):-!.
find_arg(Quatro,[_|QuatroS]):-
find_arg(Quatro,QuatroS)
 .

consistent(Undef1/Def1,Undef2/Def2):-
\+ list_member(Def1,Undef2),
\+ list_member(Undef1,Def2)
 .

%%% end complete_valuation %%%

%%% pretty print %%%
```

```
pretty_print(AttRel,UnDef,Def,UnDecided,Complete):-
write('Framework        :'),write(AttRel),nl,nl,
write('First Valuation :'),nl,
write('   Verslagen     :'),write(Def),nl,
write('   Onverslagen  :'),write(UnDef),nl,nl,
write('Onbepaalde       :'),nl,
write(' Valuaties van  :'),write(UnDecided),nl,nl,
write('Completions      :'),nl,
complete_print(Complete)
 .

complete_print([Def/UnDef|CompleteS]):-
write('   Verslagen     :'),write(Def),nl,
write('   Onverslagen  :'),write(UnDef),nl,nl,
complete_print(CompleteS)
 .
complete_print([]).
%%%   trivial predicates   %%%
append([H|T1],[H|T2],[H|T3]):-
!,
append(T1,T2,T3)
 .
append([H1|T1],[H2|T2],[H3|T3]):-
(
    name(H1,[H1Val]),
    name(H2,[H2Val]),
    H1Val < H2Val,
    !,
    H3 = H1,
    append(T1,[H2|T2],T3)
;
    H3 = H2,
    append([H1|T1],T2,T3)
)
 .
append([],L,L):-!.
append(L,[],L).

pick(H,[H|_]).
pick(H,[_|T]):-pick(H,T).

member(H,[H|_]):-!.
member(H,[_|T]):-member(H,T).
```

Referenties

[1] P. M. Dung. On the acceptibility of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-persons games. *Artificial Intelligence*, 77(2):321–357, 1995.

[2] G. Vreeswijk. A simple procedure to test membership of preferred extensions, 2000.