Università degli Studi di Pisa

Facoltà di Scienze Matematiche Fisiche e Naturali
Corso di Laurea in Scienze dell'Informazione

Tesi di Laurea

Proprietà del reticolo degli osservabili in programmazione logica

Candidato
Gianluca Amato

Relatore
Giorgio Levi

Controrelatore
Laura Ricci

Anno Accademico 1995-96

Sommario

La metodologia dell'interpretazione astratta [CC77] ha trovato ormai applicazione in molti campi, non solo limitatamente all'analisi statica di programmi, ma più in generale come mezzo per derivare automaticamente semantiche che mettano in evidenza osservabili interessanti a partire da una semantica dettagliata e completa di partenza. In questo schema è possibile inserire il framework semantico introdotto per la prima volta in [CLM95].

Molti studi sono stati portati avanti, inoltre, a partire da [CC79], per derivare automaticamente domini astratti più o meno concreti partendo da altri domini astratti di base. Oltre ai "banali" operatori di estremo superiore ed inferiore (prodotto ridotto) sul reticolo delle astrazioni, si ricordano il completamento disgiuntivo e il powerset [FR95], le dipendenze funzionali [GR95], la complementazione [CFG^{+}95, GPR96] e la minima base disgiuntiva [GR96].

L'obiettivo della tesi è mostrare tutta una serie di proprietà di cui gode il quadro semantico di [CLM95, Meo96]. Particolare riguardo è posto nell'analizzare il comportamento delle varie classi di osservabili relativamente a tali operatori sulle astrazioni, *lub* e *glb* in primo luogo.

Indice

Capitolo 1

Introduzione

L'eleganza dell'approccio classico alla sua semantica è stata per molto tempo la caratteristica più apprezzata della programmazione logica e la ragione principale della sua popolarità. L'equivalenza della procedura di prova basata sulla risoluzione SLD e delle caratterizzazioni model-theoretic e di punto fisso è stato sicuramente uno dei contributi più importanti alla materia da parte di van Emden e Kowalski nel loro articolo [vEK76].

Questa visione dei fatti, che tra l'altro ha portato più volte ad affermare che la semantica della programmazione logica è stata scritta dai logici prima ancora della sua nascita, è, tuttavia, piuttosto limitante. Essa si basa, infatti, sull'idea che l'unica cosa di rilevante nell'esecuzione di un programma logico sia il successo o meno di una query, facendo così passare in secondo piano la differenza fondamentale che c'è tra un theorem prover e un interprete PROLOG, tra logica e programmazione, che è essenzialmente il concetto di calcolo. Un interprete per un linguaggio logico non solo ci dice se una clausola è o meno soddisfacibile nel contesto del programma ma, qualora la risposta sia affermativa, ci calcola anche la soluzione che la soddisfa. Tutto ciò sparisce nella semantica classica basata sull'insieme di successo.

1.1 Semantica, osservabili, composizionalità

Il problema, inoltre, non è soltanto accademico come si potrebbe pensare in un primo momento. La semantica di un linguaggio infatti, oltre ad essere da ausilio per la comprensione del significato, anche intuitivo, di un programma, ha conseguenze dirette nello sviluppo e nelle funzionalità di tutta una serie di tools, detti appunto *guidati dalla semantica*, basati sulla analisi e trasformazione dei programmi. La nozione fondamentale in questo caso è quella di *equivalenza tra programmi* [Mah88]. Quando un programma viene trasformato in un altro, in maniera magari automatica, ad esempio per ottimizzarne la velocità d'esecuzione, si desidera che il nuovo codice ottenuto sia in un certo senso "equivalente" a quello di partenza.

La nozione di equivalenza può poi essere quasi sempre modellata su quella di *osservabile*, che rappresenta tutto ciò di una computazione su cui vogliamo focalizzare la nostra attenzione. Per la programmazione logica gli osservabili di interesse sono quelli che hanno a che fare con riposte calcolate, risultanti, call-patterns, ecc … Ogni osservabile X definisce una equivalenza tra programmi $\approx_X$ in modo tale che $P_1 \approx_X P_2$ quando tutto ciò che osserviamo sulle computazioni di P_1 lo osserviamo anche in quelle di P_2 e viceversa. Ad esempio, se *ca* è l'osservabile delle risposte calcolate, $P_1 \approx_{ca} P_2$ se per ogni goal $\boldsymbol{G}$, $\boldsymbol{G}$ ha le stesse risposte calcolate, modulo varianza, sia in P_1 che in P_2.

Una semantica $\mathcal{S}$ si dice *corretta* rispetto a una relazione di equivalenza $\approx$ quando $\mathcal{S}(P_1) = \mathcal{S}(P_2)$ implica $P_1 \approx P_2$ per tutti i programmi P_1 e P_2. Il problema fondamentale della semantica basata sul modello minimo di Herbrand è che, come viene mostrato per la prima volta in [FLMP88, FLMP89], non è corretta rispetto alla maggior parte delle equivalenze interessanti tra cui, ad esempio, proprio quella delle risposte calcolate, che di solito è quella che ogni trasformazione di programmi dovrebbe rispettare, in quanto rappresenta il legame tra input ed output di un programma logico. Per tale motivo la semantica di van Emden e Kowalski non è utilizzabile come riferimento per i sistemi di analisi e trasformazione di programmi.

Ancora un altro problema della semantica standard è che non è composizionale rispetto all'unione di programmi. In genere, dato un operatore sintattico $\circ$, una semantica $\mathcal{S}$ si dice composizionale rispetto a $\circ$ se la semantica di $P_1 \circ P_2$ è ottenibile come funzione delle semantiche di P_1 e P_2. Nel caso della programmazione logica l'operatore sintattico sicuramente più importante è $\cup$

dell'unione tra clausole, rispetto al quale abbiamo già detto che la semantica del modello minimo di Herbrand non è composizionale. Anche in questo caso non si tratta di una questione puramente di eleganza teorica dato che la $\cup$-composizionalità diventa essenziale nella definizione di versione modulari dei linguaggi logici.

Nasce dunque l'esigenza di sviluppare nuove semantiche che siano corrette rispetto alle equivalenze tra programmi più importanti [Cla79, Fer87, YYD87, DF87] e/o composizionali rispetto a vari operatori sintattici [O'K85, BGLM94, BGLM93, MP88, GS89, BLM91]. Tutti questi lavori portano ad una più profonda comprensione del funzionamento dei linguaggi logici ma causano la nascita di un proliferare di nuove definizioni semantiche. In [Gab92, GLM92] viene invece presentato un framework semantico uniforme con il quale definire varie semantiche corrette e minimali rispetto a diversi osservabili. I domini semantici utilizzati sono opportune classi di equivalenza di insiemi di clausole definite.

1.2 Interpretazione astratta

Un altro problema per cui la semantica assume un ruolo determinante è quello delle analisi statiche di groundness, sharing, goal independency [GG94], ecc ... L'approccio standard per trattare questo tipo di analisi è basato sull'uso della teoria dell'interpretazione astratta [CC77, CC92a] il cui scopo è formalizzare, in maniera indipendente dal linguaggio, la relazione tra semantiche a vari livelli di astrazione.

Una semantica basta su di un osservabile che dà informazioni sul fatto che certe risposte siano ground o meno può essere vista come una astrazione della semantica delle risposte calcolate. Analogamente per le altre semantiche coinvolte in questo tipo di analisi. L'idea delle tecniche di interpretazione astratta è quella di avere una semantica dettagliata di partenza a valori in un certo *dominio concreto* che volta per volta viene proiettata in una semantica più astratta, basata su di un *dominio astratto* di valori. In [Gia92], ad esempio, questo metodo viene usato per trattare le interpretazioni astratte di una semantica che modella i vincoli di risposta nel paradigma CLP.

Il legame tra dominio concreto e astratto, quando entrambi siano reticoli completi, viene generalmente stabilito in termini di una *connessione di Galois*, che consente di ricavare sempre la migliore semantica astratta per ogni astrazione del dominio. Nel caso particolare di *iniezioni di Galois*, che poi sono isomorfe a operatori di chiusura sul dominio concreto, l'insieme di tutte le astrazioni forma un reticolo completo, acquisendo quindi una forte struttura matematica.

Peraltro, appoggiandosi alla teoria della interpretazione astratta, si hanno a disposizione tutti i risultati della ricerca in questo campo, volti principalmente a derivare da domini astratti di partenza nuovi domini, più o meno concreti, tramite operatori quali il prodotto ridotto, le (auto)dipendenze funzionali, il completamento disgiuntivo, la complementazione, la minima base disgiuntiva, ecc ... o a velocizzare le analisi tramite operatori di narrowing e/o widening [CC92a].

1.3 La tesi

La teoria dell'interpretazione astratta può essere usata, però, anche per derivare semantiche standard, tipo quella delle risposte calcolate, partendo da semantiche più concrete quale quella dei risultanti; ciò consente di giungere ad uno schema uniforme col quale trattare sia semantiche per la specifica del funzionamento di un programma che per l'analisi statica [CL94, CLM95].

Nel corso della tesi riprenderemo il framework semantico introdotto in [CLM95, Meo96] fondato proprio su quest'idea. In esso il dominio concreto è scelto particolarmente a *basso livello*, si tratta direttamente degli alberi SLD, e vengono presentate due semantiche, una denotazionale, l'altra operazionale, definite grazie agli stessi operatori di base. Ciò consente di trattare facilmente problemi come l'equivalenza delle due semantiche, la composizionalità rispetto ai vari operatori sintattici, l'esistenza e la definizione di una denotazione goal-independent.

Le astrazioni (osservabili) sono poi formalizzate come inserzioni di Galois tra il dominio degli alberi SLD e un dominio astratto oppure, ed è quello che faremo prevalentemente noi, tramite un operatore di chiusura sul reticolo degli alberi SLD. Fissato un osservabile è possibile derivare tutta una serie di proprietà (ancora una volta composizionalità, equivalenza delle semantiche top-down e bottom-up astratte, esistenza di una denotazione astratta, precisione della semantica astratta) semplicemente esaminando le proprietà di cui sono dotate gli operatori astratti ottimali corrispodenti a quelli di base concreti. È possibile così classificare gli osservabili in varie categorie secondo le proprietà di cui godono i rispettivi operatori di base e, di conseguenza, le semantiche corrispondenti.

In questo framework è possibile inserire sia semantiche *standard*, che abbiano almeno una delle due versioni top-down o bottom-up precisa, sia quelle utilizzate per le analisi statiche che, in genere, per garantire la terminazione, devono perdere informazione. Nella tesi ci occuperemo però quasi esclusivamente delle prime.

Per esse le proprietà da considerare sugli operatori astratti di base sono essenzialmente la precisione o meno dei vari operatori. Studiando allora come tale precisione si mantiene o viene persa applicando agli osservabili le diverse trasformazioni tra domini proprie della teoria dell'interpretazione astratta e di cui abbiamo già accennato in precedenza, sapremo anche come tali trasformazioni si comportano rispetto alle nostre categorie di osservabili.

Sarà questo uno dei problemi affrontati nella tesi. Nel contempo si tenterà di studiare la struttura interna del reticolo degli osservabili, per individuare sottoreticoli significativi ed elementi "estremali" per le varie categorie. Estenderemo anche il framework originario considerando una nuova classe di osservabili, per i quali esiste una semantica top-down precisa, che non è inclusa nei lavori sopra citati.

Lo schema della tesi è il seguente:

Nel capitolo 2 introduciamo le notazioni che utilizziamo nel resto dello scritto e diamo un breve cenno alle basi della programmazione logica e della teoria dell'interpretazione astratta.

Nel capitolo 3 studiamo come vari operatori su domini astratti si comportano rispetto alla precisione di operatori definiti sui domini stessi. La trattazione qui è del tutto generale, indipendente dal framework che definiremo e utilizzeremo in seguito, per cui è applicabile anche in ambiti che con la programmazione logica non hanno nulla a che fare. Studieremo in particolare gli operatori di lub e glb nel reticolo degli osservabili, le dipendenze funzionali e il completamento disgiuntivo.

Nel capitolo 4 introduciamo il framework su cui sono basati i nostri risultati successivi. Esso è essenzialmente simile a quello presentato originariamente in [CLM95] ma sono state apportate alcune modifiche marginali, più che altro, ma non solo, a livello di notazione, per renderlo più maneggevole ai nostri fini. Vengono ricordati tutti i risultati più importanti già noti e le classi degli osservabili perfetti e denotazionali.

Nel capitolo 5 estendiamo il framework appena presentato. Dopo aver brevemente discusso del problema della varianza tra collezioni, introduciamo una particolare classe di osservabili, in un certo senso trasversale alla classificazione standard del capitolo 4, quella degli osservabili che hanno l'operatore di chiusura associato additivo. Vediamo come in tali casi sia possibile definire l'osservabile tramite una opportuna relazione di astrazione sulle derivazioni, cosa che facilita spesso le dimostrazioni di precisione per gli operatori semantici di base. Successivamente vengono presentati gli osservabili operazionali, cui abbiamo già accennato, caratterizzati dal fatto di disporre di una semantica top-down precisa.

Il capitolo 6 è dedicato all'analisi delle proprietà del reticolo degli osservabili. Si dimostra dapprima che le classi dei denotazionali, operazionali e perfetti sono chiuse rispetto agli operatori di lub e glb del reticolo degli osservabili, dopo di chè si verificano, con degli esempi, dei risultati negativi nati dai nostri tentativi di dimostrare da una parte che l'osservabile dei risultanti era il più astratto osservabile perfetto, dall'altra che le classi dei perfetti e dei denotazionali erano le uniche a godere delle proprietà di precisione e composizionalità che hanno. Il problema degli elementi estremali viene quindi riaffrontato restringendoci ai soli osservabili che conservano l'informazione sul successo di un albero SLD ottenendo questa volta dei risultati in positivo.

Nel capitolo 7 si ritratta l'argomento delle dipendenze funzionali e del completamento disgiuntivo questa volta applicati al caso concreto del nostro framework dimostrando, ad esempio, che l'osservabile della c-semantica è il più astratto osservabile additivo che concretizza l'osservabile delle sostituzioni di risposta ground.

Infine, il capitolo 8 contiene alcune osservazioni sui possibili sviluppi e sui risultanti raggiunti.

In appendice sono riportate alcune dimostrazioni piuttosto tecniche su sostituzioni e derivazioni SLD che abbiamo preferito separare per non appesantire il corpo della tesi.

Capitolo 2

Preliminari

Nel corso della tesi assumeremo familiarità con le nozioni base della teoria dei reticoli e dell'interpretazione astratta, oltre che con quelle della programmazione logica. Qui ricordiamo solamente i risultati più importanti e introduciamo le notazioni che utilizzeremo nel seguito.

Per una trattazione più approfondita e dedicata si può consultare [Bir67] per quanto riguarda la teoria dei reticoli, [CC92b, CC92a] per la teoria dell'interpretazione astratta e [Apt90, Llo87] per ciò che riguarda la programmazione logica.

2.1 Insiemi ordinati e reticoli

Una relazione $\preceq$ su un insieme C è detta *preordine* se gode delle proprietà riflessiva e transitiva. Ad ogni preordine si può associare una relazione di equivalenza $\approx$ tale che $x \approx y \Leftrightarrow x \preceq y \wedge y \preceq x$ e un *ordinamento* $\sqsubseteq$ su $C_{/\approx}$ per cui $[x] \sqsubseteq [y] \Leftrightarrow x \preceq y$ dove con $[x]$ e $[y]$ si sono indicate le classi di equivalenza dei due elementi x ed y.

Sia dato un insieme $C \neq \emptyset$ e una relazione di ordinamento che indichiamo con $\sqsubseteq_C$. La coppia $\langle C, \sqsubseteq_C \rangle$ viene detta insieme (parzialmente) ordinato o *poset*, abbreviazione di *partially ordered set*. Nel seguito indicheremo un poset solo con l'insieme dei suoi elementi e spesso, quando non nascano ambiguità, scriveremo semplicemente $\sqsubseteq$ al posto di $\sqsubseteq_C$. Lo stesso faremo per tutti gli altri operatori e relazioni indicizzati da un insieme.

Un *limite superiore* o *upper bound* di un $S \subseteq C$ non vuoto è un elemento x di C tale che $\forall y \in S.\ y \sqsubseteq x$. In maniera simmetrica si definisce il *limite inferiore* o *lower bound*. Dato $S \subseteq C$ non vuoto si indica con $\bigsqcup_C S$, se esiste, il *minimo limite superiore* di S o *lowest upper bound* (*lub*). Analogamente useremo $\bigsqcap_C S$ per il *massimo limite inferiore* o *greatest lower bound* (*glb*).

Un *reticolo completo* C è un poset in cui ogni sottoinsieme non vuoto ha un glb e un lub. Ne segue che il reticolo ha un elemento minimo $\bot_C$ e uno massimo $\top_C$. Si noti, inoltre, che C è un reticolo completo sse ha un elemento massimo e ogni sottoinsieme ha il glb. Basta infatti definire $\bigsqcup_C X = \bigsqcap_C \{y \in C \mid \forall x \in X.\ y \sqsupseteq_C x\}$ per ogni $X \subseteq C$. Analogamente nel caso in cui esista un elemento minimo e ogni sottoinsieme abbia un lub. D'ora in poi, a meno che diversamente specificato, insiemi dai nomi C, D o derivati denoteranno implicitamente dei reticoli completi.

Una funzione $f : C \to D$ si dice *monotona* se $x \sqsupseteq y$ implica $f(x) \sqsupseteq f(y)$, *additiva* se $f(\bigsqcup_{i \in I} x_i) = \bigsqcup_{i \in I} f(x_i)$ per ogni $\{x_i\}_{i \in I} \subseteq C$, *continua* se è additiva su catene. Chiaramente l'additività implica la continuità e questa a sua volta la monotonia.

Quando ci farà comodo applicheremo una $f : C \to D$ anche ad elementi di C^n. In tal caso f è da intendere come la funzione da C^n a D^n definita da $\lambda\langle x_1, \ldots, x_n\rangle.\ \langle f(x_1), \ldots, f(x_n)\rangle$. Lo stesso faremo anche per funzioni che non sono definite da/a reticoli.

Due reticoli completi C e D si dicono *isomorfi* se esiste $f : C \to D$ monotona e invertibile e lo si indica con la notazione $C \cong D$; f viene detta *isomorfismo*.

Si dice che $x \in C$ è punto fisso di un operatore $T : C \to C$ se $T(x) = x$. Se T è monotono allora ha un reticolo completo di punti fissi. Indichiamone con $\mathrm{lfp}(T)$ e $\mathrm{gfp}(T)$ il minimo e massimo.

Per ogni ordinale α si definiscono le *potenze ordinali* di T all'insù e all'ingiù come:

$$
\begin{aligned}
&T \uparrow 0 = \bot && T \downarrow 0 = \top \\
&T \uparrow (\alpha+1) = T(T \uparrow \alpha) && T \downarrow (\alpha+1) = T(T \downarrow n) \\
&T \uparrow \beta = \bigsqcup \{T \uparrow n \mid n < \beta\} && T \downarrow \beta = \bigsqcap \{T \downarrow n \mid n < \beta\}
\end{aligned}
$$

per β ordinale limite.

Le potenze ordinali consentono di ricavare il massimo e minimo punto fisso di T in maniera quasi costruttiva. Esistono infatti due ordinali γ_1 e γ_2 tali che $\mathrm{lfp}(T) = T \uparrow \gamma_1$ e $\mathrm{gfp}(T) = T \downarrow \gamma_2$. Se T è continua risulta $\gamma_1 = \omega$ primo ordinale transfinito.

Utilizzeremo anche la seguente notazione standard:

$$
\begin{aligned}
&T^0(x) = x \\
&T^{\alpha+1}(x) = T(T^\alpha(x)) \\
&T^\alpha(x) = \bigsqcup \{T^\beta \mid \beta < \alpha\} && \text{(per } \alpha \text{ ordinale limite)}
\end{aligned}
$$

che permette di esprimere $T \uparrow \alpha$ come $T^\alpha(\bot)$. Si dimostra, più in generale, che se $T(x) \sqsupseteq x$, esiste un ordinale γ per cui $T^\gamma(x)$ è il minimo punto fisso di T maggiore o uguale ad x, che è ottenibile anche come $\mathrm{lfp}(\lambda y.\ T(y \sqcup x))$.

Definizione 2.1.1 (Operatore di chiusura) *Un* operatore di chiusura *(superiore) [War42] (in breve* uco *per* upper closure operator*) su C è un operatore $\rho : C \to C$ monotono, idempotente ed estensivo (ovvero $\forall x \in C.\ \rho(x) \sqsupseteq x$).*

Si dimostrano le seguenti proprietà:

1. Ogni operatore di chiusura ρ su C è univocamente identificato dall'insieme dei suoi punti fissi, uguale a $\rho(C)$. Sarà in particolare $\rho(x) = \bigsqcap\{y \sqsupseteq x \mid y \in \rho(C)\}$.

2. $X \subseteq C$ è detto *famiglia di Moore* se $\top \in X$ ed X è chiuso rispetto al glb. L'insieme dei punti fissi di ρ è una famiglia di Moore e, viceversa, data una famiglia di Moore X, esiste un uco ρ tale che $\rho(C) = X$.

3. L'insieme dei punti fissi di un operatore di chiusura è un reticolo completo ma non è detto sia un sottoreticolo di C, a meno che l'uco non sia additivo, poiché, in linea di massima, il lub differisce.

L'insieme degli operatori di chiusura su C, che indicheremo con $\mathrm{uco}(C)$, forma a sua volta un reticolo completo tale che, per ogni $\rho, \eta \in \mathrm{uco}(C)$, $\{\rho_i\}_{i\in I} \subseteq \mathrm{uco}(C)$ ed $x \in C$:

1. $\rho \sqsubseteq \eta$ sse $\forall x \in C.\ \rho(x) \sqsubseteq \eta(x)$ o, in maniera equivalente, $\eta(C) \subseteq \rho(C)$
2. $\left(\bigsqcap_{i\in I} \rho_i\right)(x) = \bigsqcap_{i\in I} \rho_i(x)$
3. $\left(\bigsqcup_{i\in I} \rho_i\right)(C) = \bigcap_{i\in I} \rho_i(C)$

2.2 Interpretazione astratta

Definizione 2.2.1 (Connessione/inserzione di Galois) *Siano C e D due reticoli completi, $\gamma : D \to C$ e $\alpha : C \to D$ due funzioni monotone. La tupla $G = \langle \gamma, C, D, \alpha \rangle$ è una* connessione di Galois *(G.c.) se $\forall c \in C.\ c \sqsubseteq_C \gamma(\alpha(c))$ e $\forall d \in D.\ d \sqsupseteq_D \alpha(\gamma(d))$. Si ha una* inserzione *o* iniezione di Galois *(G.i.) se, in più, $\forall d \in D.\ d = \alpha(\gamma(d))$.*

Valgono le seguenti proprietà:

1. Siano C e D due reticoli completi, $\gamma : D \to C$ e $\alpha : C \to D$ due funzioni monotone. La tupla $G = \langle \gamma, C, D, \alpha \rangle$ è una *aggiunzione* se $\forall c \in C.\ \forall d \in D.\ \alpha(c) \sqsubseteq_D d \Leftrightarrow c \sqsubseteq_C \gamma(d)$. α (γ) è chiamato l'*aggiunto sinitro (destro)* di γ (α). $\langle \gamma, C, D, \alpha \rangle$ è una aggiunzione sse è una connessione di Galois.
2. Una connessione di Galois è una inserzione sse γ è iniettiva. D'altronde γ è iniettiva sse α è surgettiva.
3. $\gamma \circ \alpha$ è un operatore di chiusura su C.

Nell'ambito dell'interpretazione astratta C e D sono detti rispettivamente *dominio concreto* e *dominio astratto*, mentre γ ed α sono le funzioni di *concretizzazione* e di *astrazione*. Le relazioni di ordinamento rappresentano una misura del contenuto informativo di ogni elemento del dominio. Si dice che $y \in D$ è *astrazione* di $x \in C$ se $\gamma(y) \sqsupseteq x$ o, equivalentemente, $\alpha(x) \sqsubseteq y$. Ogni elemento del dominio concreto ha una astrazione più piccola della altre, quindi più precisa, data da $\alpha(x)$.

Dato un operatore $\mathrm{op} : C^n \to C$ sul dominio concreto e uno $\tilde{\mathrm{op}} : D^n \to D$ sul dominio astratto, si dice che $\tilde{\mathrm{op}}$ è *corretto* rispetto a op se $\forall \boldsymbol{x} \in C^n$, $\mathrm{op}(\boldsymbol{x}) \sqsubseteq \gamma(\tilde{\mathrm{op}}(\alpha(\boldsymbol{x})))$. Si ha che la composizione di operatori corretti è ancora corretta e che, se $\tilde{T} : D \to D$ è corretto rispetto a T, allora $\alpha(\mathrm{lfp}(T)) \sqsubseteq \mathrm{lfp}(\tilde{T})$, ovvero $\mathrm{lfp}(\tilde{T})$ è una astrazione di $\mathrm{lfp}(T)$.

Nell'ambito che a noi interessa avremo di solito una semantica definita come minimo punto fisso di un operatore T, a sua volta definito come composizione di un certo numero di operatori di base. Allora, disponendo di una versione astratta corretta di tali operatori è possibile derivare da essi, automaticamente, una semantica astratta corretta.

Tra tutte le astrazioni corrette di un operatore op ve ne è una *ottimale*, quella definita da $\tilde{\mathrm{op}}(x) = \alpha(\mathrm{op}(\gamma(x)))$ per ogni $x \in D$. Purtroppo non è vero in generale che la composizione di operatori ottimale è anch'essa ottimale.

In genere il passaggio dalla semantica concreta a quella astratta provoca una perdita di precisione, a meno che tutti gli operatori coinvolti siano precisi. Un operatore astratto $\tilde{\mathrm{op}}$ si dice *preciso* rispetto a op se $\forall \boldsymbol{x} \in C^n$, $\alpha(\mathrm{op}(\boldsymbol{x})) = \tilde{\mathrm{op}}(\alpha(\boldsymbol{x}))$. Non sempre data una connessione di Galois ed op esiste un operatore astratto preciso per op ma, quando ciò accade, sicuramente tale operatore è quello ottimale, per cui si deve avere $\alpha(\mathrm{op}(\boldsymbol{x})) = \alpha(\mathrm{op}((\gamma \circ \alpha)\boldsymbol{x}))$. Quando per una G.c. esiste un operatore astratto preciso per op diremo che la G.c. è precisa rispetto ad op o, viceversa, che op è preciso rispetto alla connessione di Galois.

La composizione di operatori precisi è precisa. Inoltre, se $\tilde{T} : C \to C$ è preciso per l'operatore T, allora $\alpha(\mathrm{lfp}(T)) = \mathrm{lfp}(\tilde{T})$ e quindi la nostra semantica astratta viene computata senza perdita di informazione, essendo uguale a quella che si ottiene astraendo direttamente dalla semantica concreta. La precisione è di solito gradita ma spesso occorre farne a meno quando si vuole che il calcolo del minimo punto fisso di $\tilde{T}$ termini, ad esempio nei problemi di analisi statica.

2.2.1 Iniezioni di Galois ed operatori di chiusura

Si noti che è possibile [CC79] associare ad ogni inserzione di Galois l'uco $\gamma \circ \alpha$ e viceversa ad ogni uco ρ su C le iniezioni di Galois $\langle \iota^{-1}, C, D, \iota \circ \rho \rangle$ dove ι è un isomorfismo tra $\rho(C)$ e D. Scegliendo $D = \rho(C)$ si ottiene la G.i. $\langle \lambda x.x, C, \rho(C), \rho \rangle$ che può quindi essere considerata il rappresentante canonico della classe di equivalenza di tutte le G.i. che hanno dominio astratto D isomorfo a $\rho(C)$. Tutto ciò fornisce una elegante caratterizzazione delle G.i. che si rivela, tra l'altro, particolarmente utile per snellire le dimostrazioni.

Inoltre, da questo punto di vista, le operazioni di lub e glb sul reticolo degli operatori di chiusura assumono un nuovo significato. Siano $\{\rho_i\}_{i \in I} \subseteq \mathrm{uco}(C)$, con ogni ρ_i associato ad una inserzione di Galois GI_i. Il glb dei ρ_i è allora associato ad una G.i. che è la più astratta tra quelle più concrete delle GI_i, ovvero è associato al *prodotto ridotto* delle GI_i. In maniera simmetrica il lub dei ρ_i è associato alla G.i. più concreta tra quelle che astraggono tutte le GI_i.

I concetti di operatore corretto, ottimale e preciso hanno anche loro una adeguata corrispondenza in questo quadro. Un operatore $\tilde{\mathrm{op}} : \rho(C)^n \to \rho(C)$ rispetto a $\mathrm{op} : C^n \to C$ è:

- corretto se $\forall \boldsymbol{x} \in C^n.\, \mathrm{op}(\boldsymbol{x}) \sqsubseteq \tilde{\mathrm{op}}(\rho(\boldsymbol{x}))$
- ottimale se $\forall \boldsymbol{x} \in \rho(C)^n.\, \tilde{\mathrm{op}}(\boldsymbol{x}) = \rho(\mathrm{op}(\boldsymbol{x}))$
- preciso se $\forall \boldsymbol{x} \in C^n.\, \tilde{\mathrm{op}}(\rho(\boldsymbol{x})) = \rho(\mathrm{op}(\boldsymbol{x}))$

e l'operatore astratto ottimale di op è preciso sse $\forall \boldsymbol{x} \in C^n$, $\rho(\mathrm{op}(\rho(\boldsymbol{x}))) = \rho(\mathrm{op}(\boldsymbol{x}))$. In tal caso si dirà che ρ è preciso rispetto a op o che op lo è rispetto a ρ.

2.3 Programmazione logica

Sia $S = \langle \Phi, \Pi, V \rangle$ la segnatura di un linguaggio del prim'ordine, dove:

- Φ è un insieme di *simboli di funzione* di arità fissata,
- Π è un insieme di *simboli di predicato* di arità fissata,
- V è un insieme numerabilmente infinito di *variabili*.

Simboli di funzione di arità 0 sono detti anche simboli di *costante*. L'insieme dei termini del nostro linguaggio lo si indica con $\tau(\Phi \cup V)$ mentre con $\tau(\Phi)$ si indica l'insieme dei termini *ground* (ovvero non contenenti alcuna variabile) e con $\mathcal{L}(S)$ l'insieme delle formule del prim'ordine.

In generale chiamiamo *espressione* un termine, una formula o qualunque altro oggetto sintattico costruito a partire dalla segnatura S. Data una espressione E indichiamo con $\mathrm{var}(E)$ l'insieme di tutte le variabili che occorrono in essa.

In tutto quanto diremo nel seguito è da intendere fissata implicitamente una segnatura.

2.3.1 Teoria dell'unificazione

Una *sostituzione* è una applicazione $\theta : V \to \tau(\Phi \cup V)$ tale che $\mathrm{dom}(\theta) = \{x \in V \mid \theta(x) \neq x\}$ (il *domininio* di θ) è finito. Per indicare una sostituzione θ t.c. $\mathrm{dom}(\theta) = \{x_1, \ldots, x_n\}$ e $\theta(x_i) = t_i$ per ogni $i \in [1..n]$ utilizzeremo la notazione:

$$\theta = \{x_1/t_1, \ldots, x_n/t_n\}$$

dove $x_1, \ldots, x_n$ sono variabili distinte.

Indichiamo con $\mathrm{range}(\theta)$ l'insieme $\{y \in V \mid \exists x \in \mathrm{dom}(\theta) \text{ t.c. } y \in \mathrm{var}(\theta(x))\}$ e poniamo $\mathrm{var}(\theta) = \mathrm{dom}(\theta) \cup \mathrm{range}(\theta)$. Una sostituzione è detta *ground* se $\mathrm{range}(\theta) = \emptyset$ mentre ϵ denota la sostituzione che ha come dominio l'insieme vuoto.

Se $W \subseteq V$ indicheremo con $\theta|_W$ la *restrizione* di θ alle variabili di W, così definita:

$$\theta|_W\,(x) = \begin{cases} \theta(x) & \text{se } x \in W \\ x & \text{se } x \notin W \end{cases}$$

Data un'espressione E useremo spesso la notazione $\theta|_E$ al posto di $\theta|_{\mathrm{var}(E)}$.

Date due sostituzioni θ e σ indichiamo con $\theta\sigma$ la loro *composizione* funzionale, ovvero la funzione $\lambda x.\sigma(\theta(x))$. La composizione di sostituzioni gode della proprietà associativa ed ϵ è l'elemento neutro.

Una *ridenominazione* o *renaming* è una sostituazione σ tale che esiste la sostituzione inversa σ^{-1} per cui $\sigma\sigma^{-1} = \sigma^{-1}\sigma = \epsilon$. Una sostituzione θ è detta *idempotente* se $\theta\theta = \theta$ o, equivalentemente, se $\mathrm{var}(\theta) \cap \mathrm{range}(\theta) = \emptyset$. Indichiamo con Subst l'insieme di tutte le sostituzioni idempotenti.

Data un'espressione E e una sostituazione θ con $E\theta$ si indica l'oggetto ottenuto da E sostituendo simultaneamente ad ogni occorrenza della variabile v il termine $\theta(v)$. θ è *grounding* per E se $E\theta$ è ground.

Sulle sostituzioni è possibile porre la relazione $\leq$ così definita: $\theta \leq \theta'$ sse $\exists\gamma$ t.c. $\theta'\gamma = \theta$. In maniera simile per le espressioni: $E \leq E'$ sse $\exists\theta$ t.c. $E'\theta = E$. E e θ sono dette *istanza* di E' e θ' rispettivamente. Si dice anche che E' (θ') è *più generale* di E (θ). Due espressioni E ed E' sono *varianti* se esiste una rinominazione σ t.c. $E = E'\sigma$ e lo si indica con $E \equiv E'$.

Si dice *unificatore* delle espressoni $E_1, \ldots, E_n$ una sostituzione θ t.c. $E_1\theta = \cdots = E_n\theta$. Esso è detto *unificatore più generale* (o mgu) se è il più generale tra gli unificatori di $E_1, \ldots, E_n$ ed int tal caso è unico a meno di ridenominazioni. Dato un insieme espressioni, esse si dicono unificabili se esiste per loro un unificatore. Si noti che l'esistenza di un unificatore implica l'esistenza di un mgu idempotente. Con $\mathrm{mgu}(E_1, \ldots, E_n)$ denotiamo l'insieme degli mgu delle espressioni $E_1, \ldots, E_n$.

Una trattazione esauriente della teoria dell'unificazione si può trovare in [LMM88].

2.3.2 Il linguaggio

Sulla segnatura S di cui sopra definiamo il nostro linguaggio di programmazione logica:

$$\begin{aligned} GOAL &::= \square \mid ATOM, GOAL \\ CLAUSE &::= ATOM \leftarrow GOAL \\ PROG &::= \{CLAUSE\} \cup PROG \end{aligned}$$

Esaminiamo più da vicino le varie categorie sintattiche:

- $ATOM$ è una formula atomica della logica del prim'ordine, ovvero una formula del tipo $p(\boldsymbol{t})$ con $p \in \Pi$ simbolo di predicato di arità n e $\boldsymbol{t}$ n-upla di termini. Nel seguito indicheremo un atomo con i simboli H ed A, eventualmente con apici e pedici.

- Le stringhe generate da $GOAL$ sono dette, appunto, *goals*, e verranno indicate con il simbolo $\boldsymbol{G}$ o derivati.

- Le stringe generate da $CLAUSE$ sono dette *clausole* (definite) e verranno nel seguito indicate con c o derivati. Ogni clausola del tipo $H \leftarrow A_1, \ldots, A_n$, dove H è la *testa* e $A_1, \ldots, A_n$ il *corpo*, può essere pensata come una rappresentazione della formula del prim'ordine $\vec{\forall}.(A_1 \wedge \cdots \wedge A_n \Rightarrow H)$ con $\vec{\forall}E$ che indica la chiusura universale di E. Se il corpo di una clasuola è il goal vuoto $\square$, la clausola è detta unitaria ed è la rappresentazione della formula $\vec{\forall}.H$ in logica dei predicati.

- Per *programma logico* (definito) si intende una stringa generata da $PROG$ ed indicata, nel seguito, con P. Ogni programma è isomorfo ad un insieme finito di clausole e quindi, come tale, anche ad un insieme finito di formule del prim'ordine, per cui è possibile parlare dei suoi modelli e delle sue conseguenze logiche nel senso classico della semantica di Tarski.

Talvolta si utilizzerà anche la categoria sintattica $QUERY$ e la produzione:

$$QUERY ::= GOAL \ \texttt{in} \ PROG$$

che serve a specificare in un unico blocco programma e goal di cui si richiede il "calcolo".

Indicheremo con Atoms, Goals, Clauses, Progs e Queries i linguaggi generati dalle omonime categorie sintattiche.

Dato un goal $\boldsymbol{G} = A_1, \ldots, A_n$, una sostituzione θ si dice (sostituzione di) *risposta corretta* per $P \cup \boldsymbol{G}$ se $\vec{\forall}.(A_1 \wedge \cdots \wedge A_n)\theta$ è una conseguenza logica di P.

Semantica operazionale

Ad un *programma logico* si può attribuire un significato operazionale utilizzando la *risoluzione SLD*, una specializzazione della procedura di risoluzione di Robinson per clausole definite, che dà luogo alle derivazioni SLD.

Definizione 2.3.1 (Derivazione SLD) *Dato un programma P ed un goal $\boldsymbol{G}$ si chiama* derivazione SLD *di $P \cup \boldsymbol{G}$ una sequenza possibilmente infinita di goal $\boldsymbol{G}_0$, $\boldsymbol{G}_1$, ... , $\boldsymbol{G}_n$, ... detti* risolventi *insieme ad una sequenza c_1, c_2, ... , c_n, ... di clausole* rinominate apart, *ovvero tali che $\forall i \geq 1$, $\mathrm{var}(c_i) \cap \mathrm{var}(\boldsymbol{G}_0, c_1, \ldots, c_{i-1}) = \emptyset$, e ad una sequenza $\theta_1, \theta_2, \ldots, \theta_n, \ldots$ di mgu idempotenti tali che $\boldsymbol{G}_0 = \boldsymbol{G}$ e $\forall i \geq 1$, $\boldsymbol{G}_i$ è il risolvente di $\boldsymbol{G}_{i-1}$ e c_i usando θ_i. Una* refutazione SLD *è una derivazione SLD che ha $\Box$ come ultimo goal.*

Indicheremo nel seguito una derivazione SLD con la notazione:

$$\boldsymbol{G}_0 \xrightarrow[c_1]{\theta_1} \boldsymbol{G}_1 \xrightarrow[c_2]{\theta_2} \ldots \xrightarrow[c_n]{\theta_n} \boldsymbol{G}_n \xrightarrow[\theta_{n+1}]{c_{n+1}} \ldots$$

omettendo gli mgu o le clausole quando non interessi esplicitarne il valore. Se d è finita e termina col goal $\boldsymbol{G}_n$, si definiscono:

- $\mathrm{first}(d) = \boldsymbol{G}_0$, il primo goal di una derivazione d,
- $\mathrm{last}(d) = \boldsymbol{G}_n$, l'ultimo goal di d,
- $\mathrm{length}(d) = n$, lunghezza della derivazione d,

- answer$(d) = (\theta_1 \ldots \theta_n)|_{\boldsymbol{G}_0}$, detta (sostituzione di) *risposta calcolata parziale*. Se $\boldsymbol{G}_n = \Box$ si parla di *risposta calcolata*,

- result$(d) = (\boldsymbol{G}_0 \, \text{answer}(d) \leftarrow \boldsymbol{G}_n)$, detto *risultante*. L'insieme di tutti i risultanti viene indicato con Resultants, quelli relativi al solo goal inziale $\boldsymbol{G}_0$ con $\text{Resultants}_{\boldsymbol{G}_0}$.

- clauses$(d) = (c_1, \ldots, c_n)$, la sequenza delle clausole usate nella derivazione.

Talvolta si userà anche il concetto di *regola di selezione*. Essa è una funzione che quando applicata ad una derivazione SLD finita d, seleziona un particolare atomo in last(d). Una derivazione SLD via la regola di selezione R è come una normale derivazione in cui però $\boldsymbol{G}_i$ è il risolvente di c_i e $\boldsymbol{G}_{i-1}$ nel quale l'atomo su cui si risolve è quello selezionato da R sulla derivazione $\boldsymbol{G}_0 \to \ldots \to \boldsymbol{G}_i$.

Una regola di selezione molto usata è quella sinistra del PROLOG, indicata con L, che seleziona sempre il primo atomo dell'ultimo goal, indipendentemente dal resto della derivazione.

Una categoria molto importante di regole di selezione è quella delle *regole locali*.

Definizione 2.3.2 (Regole locali) *Una regola di selezione R si dice* locale *sse esiste una bigezione ϕ su $\mathbb{N}$ tale che valgano le seguendi condizioni:*

1. *se $\boldsymbol{G} = A_1, \ldots, A_n$ è il goal iniziale, allora l'atomo selezionato da R in $\boldsymbol{G}$ è l'atomo A_s tale che $\phi(s) < \phi(i)$ per ogni $i \leq n, i \neq s$;*

2. *se $\boldsymbol{G}$ è un generico risolvente, sia $A_1, \ldots, A_n$ la sequenza di atomi in $\boldsymbol{G}$ aggiunta dall'ultimo passo di derivazione. Allora, come prima, l'atomo selezionato è A_s tale che $\phi(s) < \phi(i)$ per ogni $i \leq n, i \neq s$.*

Dato un programma P, l'insieme delle derivazioni SLD via R per un goal $\boldsymbol{G}$ forma un albero detto *albero-SLD*. Spesso si sintetizzano le informazioni date da tutti gli alberi SLD per un programma P nel cosidetto *insieme di successo* di P, ovvero l'insieme di tutti gli atomi ground per cui esiste una refutazione SLD via R, definito come:

$$\mathcal{O}_R(P) = \{A \text{ ground} \mid \exists d \text{ refutazione SLD via } R \text{ di } P \cup A\}$$

Valgono i seguenti risultati:

- Sia P un programma, $\boldsymbol{G} = A_1, \ldots, A_n$ un goal e θ la risposta calcolata di una derivazione SLD di $P \cup \boldsymbol{G}$. Allora θ è una risposta corretta per $P \cup \boldsymbol{G}$.

- Sia P un programma, $\boldsymbol{G}$ un goal ed R una regola di selezione. Se θ è una risposta corretta per $P \cup \boldsymbol{G}$, allora esiste una refutazione SLD via R di $P \cup \boldsymbol{G}$ con risposta calcolata γ tale che $\boldsymbol{G}\gamma \geq \boldsymbol{G}\theta$.

- L'insieme di successo di un programma P non dipende dalla regola di selezione ed è l'insieme delle conseguenze logiche atomiche ground di P. Per tale motivo si può scrivere semplicemente $\mathcal{O}(P)$ invece di $\mathcal{O}_R(P)$.

Semantica denotazionale

Data la segnatura S, chiamiamo *universo di Herbrand* U_S l'insieme $\tau(\Phi)$ dei termini ground del nostro linguaggio e con *base di Herbrand* B_S l'insieme di tutti gli atomi ground. Una *interpretazione di Herbrand* è un sottoinsieme della base di Herbrand. Ad essa si può fare corrispondere una interpretazione nel senso classico della semantica della logica del prim'ordine alla Tarski. Chiaramente con *modello di Herbrand* di un programma P si intenderà una interpretazione di Herbrand per la quale l'interpretazione di Tarski associata è modello di P .

Un importante risultato è che, supponendo che la nostra segnatura abbia almeno un simbolo di costante, un programma logico ha un modello se e solo se ha un modello di Herbrand. Inoltre, l'intersezione di modelli di Herbrand è ancora tale. Ne segue l'esistenza di un modello di Herbrand minimo per ogni programma P, denominato $\mathcal{M}(P)$.

Esso può essere calcolato come minimo punto fisso di un operatore sulle intepretazioni di Herbrand così definito:

$$T_P(I) = \big\{A \in B_L \mid A \leftarrow B_1, \ldots, B_n \text{ è una istanza ground di una clausola in } P \text{ e } \{B_1, \ldots, B_n\} \subseteq I\big\}.$$

Si dimostra che il minimo modello di Herbrand di un programma P ed il suo insieme di successo coincidono, ovvero che

$$\mathcal{O}_R(P) = \mathcal{M}(P)$$

per ogni regola di selezione R.

2.4 Ulteriori notazioni

Vediamo qualche altra notazione che utilizzeremo più volte nel corso della tesi.

Dato un insieme S indicheremo con $\#S$ la sua cardinalità. Se f è una funzione su insiemi semplificheremo spesso la notazione scrivendo $f\{x \mid P(x)\}$ invece del più corretto $f\big(\{x \mid P(x)\}\big)$.

Sia $\approx$ una relazione di equivalenza su un insieme S e $\mathrm{op} : S^n \to S$ tale che $\forall x_1, \ldots, x_n, y_1, \ldots, y_n.\ \big(\forall i \leq n.\ x_i \approx y_i\big) \Rightarrow \big(\mathrm{op}(x_1, \ldots, x_n) \approx \mathrm{op}(x_1, \ldots, x_n)\big)$. Allora $\approx$ si dice *conguenza* rispetto a op. In tal caso è possibile definire su $S_{/\approx}$, ovvero l'insieme delle classi di equivalenza di S, un *lifting* di op, l'operatore $\hat{\mathrm{op}}$ tale che:

$$\hat{\mathrm{op}}([x_1], \ldots, [x_n]) = [\mathrm{op}(x_1, \ldots x_n)]$$

Spesso si tenderà ad utilizzare sempre op come nome sia per l'operatore originario che per quello definito su $S_{/\approx}$. Analogo discorso si può fare per le relazioni definite su S.

È possibile inoltre estendere la relazione $\approx$ a sottoinsiemi di S ponendo, per ogni $X, X' \subseteq S$, $X \approx X'$ sse $\forall x \in X.\ \exists x' \in X'$ t.c. $x \approx x'$ e viceversa. La $\approx$ estesa rimane una relazione di equivalenza e si ha un isomorfismo tra le classi di sottoinsiemi di S presi modulo $\approx$ e i sottoinsiemi di $S_{/\approx}$.

Capitolo 3

Proprietà degli operatori di chiusura

Come abbiamo già accennato in precedenza, quando parleremo d osservabili e astrazioni lo faremo preferendo sempre formalizzare la discussione tramite operatori di chiusura piuttosto che con le iniezioni di Galois. A nostro avviso ciò semplifica notevolmente il formalismo e le dimostrazioni senza danneggiare eccessivamente l'intuitività.

A tal proposito, andremo a vedere alcune delle proprietà di cui godono gli uco, con particolare attenzione al modo in cui l'eventuale precisione rispetto ad un operatore op si comporta nei riguardi di varie forme di composizione degli uco stessi.

3.1 Semplici proprietà di base

Cominciamo ad esaminare alcune proprietà per molti versi banali ma delle quali è bene essere pienamente coscienti, visto che si riveleranno più volte utili nel seguito.

Teorema 3.1.1 *Sia ρ un uco su C e $x, y \in C$. Allora:*

$$\rho(x) \sqsupseteq \rho(y) \Leftrightarrow \rho(x) \sqsupseteq y$$

Dimostrazione. Si verificano separatamente i due versi dell'implicazione:

- $\rho(x) \sqsupseteq y \Rightarrow \rho^2(x) \sqsupseteq \rho(y) \Rightarrow \rho(x) \sqsupseteq \rho(y)$

- $\rho(x) \sqsupseteq \rho(y) \Rightarrow \rho(x) \sqsupseteq \rho(y) \sqsupseteq y$ ■

Teorema 3.1.2 *Sia $op : C \to C$ una funzione monotona su C e $\forall i \in I.\ x_i \in C$. Allora:*

$$\mathrm{op}\left(\bigsqcap_{i\in I} x_i\right) \sqsubseteq \bigsqcap_{i\in I} \mathrm{op}(x_i)$$

Dimostrazione. Per l'ipotesi di monotonia segue $\mathrm{op}\left(\bigsqcap_{i\in I} x_i\right) \sqsubseteq \mathrm{op}(x_i)$ per ogni $i \in I$ e, dalle proprietà del glb, la tesi. ■

Dedichiamoci ora ad alcune particolarità degli operatori di chiusura additivi.

Definizione 3.1.3 (Base disgiuntiva) *Dato un reticolo completo C, un $S \subseteq C$ è detto* base disgiuntiva *di C se per ogni $x \in C$ esiste $S' \subseteq S$ tale che $x = \bigsqcup S'$.*

Teorema 3.1.4 *Sia ρ un operatore di chisura su C additivo e* $\mathrm{op} : C^n \to C$ *un operatore anch'esso additivo. Allora, per ogni base disgiuntiva S di C, ρ è preciso rispetto a* op *sse per ogni $s_1, \ldots, s_n \in S$,*

$$\rho(\mathrm{op}(s_1, \ldots, s_n)) = \rho(\mathrm{op}(\rho(s_1), \ldots, \rho(s_n)))$$

Dimostrazione. Chiaramente se ρ è preciso per op la proprietà di cui sopra è vera, per cui un lato dell'implicazione è banale. D'altro canto, siano $x_1, \ldots, x_n$ elementi di C e $x_i = \bigsqcup S'_i$ con $S'_i = \{s_{i,j} \mid j \in I_i\} \subseteq S$ per ogni $i \in [1..n]$,

$$\begin{array}{lcl}
\rho(\mathrm{op}(\rho(x_1), \ldots, \rho(x_n))) & = & [\,\text{definizione degli } x_i\,] \\
\rho(\mathrm{op}(\rho(\bigsqcup S'_1), \ldots, \rho(\bigsqcup S'_n))) & = & [\,\text{additività di } \rho \text{ ed op}\,] \\
\bigsqcup\{\rho(\mathrm{op}(\rho(s_{1,i_1}), \ldots, \rho(s_{n,i_n})))\}_{i_1\in I_1,\ldots,i_n\in I_n} & = & [\,\text{ipotesi del teorema}\,] \\
\bigsqcup\{\rho(\mathrm{op}(s_{1,i_1}, \ldots, s_{n,i_n}))\}_{i_1\in I_1,\ldots,i_n\in I_n} & = & [\,\text{additività di } \rho \text{ ed op}\,] \\
\rho(\mathrm{op}(x_1, \ldots, x_n)) & &
\end{array}$$

come volevasi dimostrare. ■

3.2 Il lub degli operatori di chiusura

A questo punto possiamo passare ad analizzare più da vicino l'operatore di minimo limite superiore sul reticolo uco(C). La definizione di lub che abbiamo dato nella sezione 2.1 non è neanche lontanamente costruttiva e ciò si riflette in una certa difficoltà nel trattare il reticolo degli operatori di chiusura.

Qui vogliamo rimediare in parte al problema evidenziando alcune proprietà di cui gode il lub per arrivare a delle definizioni alternative un pò più intuitive e facili da manipolare. Un risultato analogo si trova anche in [CC79].

Teorema 3.2.1 *Sia* $\rho = \bigsqcup_{i \in I} \rho_i$, *con* $\{\rho_i\}_{i \in I} \subseteq \mathrm{uco}(C)$. *Allora* $\rho = \rho \circ \rho_i = \rho_i \circ \rho$ *per ogni* $i \in I$

Dimostrazione. Per ogni $i \in I$ e $x \in C$ abbiamo $x \sqsubseteq \rho_i(x) \sqsubseteq \rho(x)$ e applicando ρ a tutti e tre i termini si ottiene, per monotonia ed idempotenza degli uco, $\rho(x) \sqsubseteq \rho(\rho_i(x)) \sqsubseteq \rho(x)$. Segue $\rho \circ \rho_i = \rho$.

D'altronde $\rho_i(\rho(x)) \sqsupseteq \rho(x)$ per estensionalità ma $\rho_i(\rho(x)) \sqsubseteq \rho^2(x) = \rho(x)$ da cui $\rho_i \circ \rho = \rho$. ■

Teorema 3.2.2 *Sia* $\rho = \bigsqcup_{i \in I} \rho_i$, *con* $\{\rho_i\}_{i \in I} \subseteq \mathrm{uco}(C)$, $x, y \in C$. *Abbiamo:*

$$x \sqsupseteq y \wedge \rho(x) \sqsupset \rho(y) \Rightarrow \forall i \in I.\ \rho_i(x) \sqsupset \rho_i(y)$$

Dimostrazione. Chiaramente per tutti gli $i \in I$ è $\rho_i(x) \sqsupseteq \rho_i(y)$ (monotonia degli uco). Se fosse $\rho_i(x) = \rho_i(y)$ si avrebbe, grazie al teorema 3.2.1, $\rho(x) = \rho(\rho_i(x)) = \rho(\rho_i(y)) = \rho(y)$ che è assurdo. ■

Veniamo quindi ad una caratterizzazione del lub che, in un certo senso, può dirsi costruttiva, anche se è ben lungi dal poter essere utilizzata come algoritmo di calcolo. Ricordiamo che per ogni uco ρ sul reticolo completo C vale

$$\rho(x) = \bigsqcap \{y \sqsupseteq x \mid y \in \rho(C)\} \tag{3.2.1}$$

per cui, se $\rho = \bigsqcup_{i \in I} \rho_i$, allora $\rho(C) = \bigsqcap_{i \in I} \rho_i(C)$ e quindi

$$\rho(x) = \bigsqcap \{y \sqsupseteq x \mid \forall i \in I.\ \rho_i(y) = y\} \tag{3.2.2}$$

Consideriamo ora l'operatore $T(y) = \bigsqcup_{i \in I} \rho_i(y)$. Si noti che y è punto fisso di T sse lo è di tutti i ρ_i. Il teorema di sopra può allora essere riletto affermando che $(\bigsqcup_{i \in I} \rho_i)(x)$ è il minimo punto fisso di T maggiore o uguale ad x. Segue:

Corollario 3.2.3 (Caratterizzazione costruttiva del lub) *Siano ρ_i con $i \in I$ degli uco su C. Allora:*

$$\Big(\bigsqcup_{i \in I} \rho_i\Big)(x) = \mathrm{lfp}\, \lambda y.\ T(x \sqcup y) = T^{\alpha}(x)$$

per un opportuno ordinale α.

Dimostrazione. Segue da quanto visto nella sezione 2.1, poiché T è un operatore monotono ed estensivo, per cui $T(x) \sqsupseteq x$ qualunque sia x. ∎

3.3 Operatori di chiusura precisi

Il problema che ci poniamo ora è quello di stabilire se il glb e il lub di uco precisi rispetto ad un certo operatore op rimangono o no tali. Quello che dimostreremo, che è già di per sè un importante risultato della teoria dell'interpretazione astratta, verrà poi utilizzato per dimostrare facilmente che il lub e il glb di osservabili denotazionali, perfetti e operazionali sono ancora osservabili dello stesso tipo.

Teorema 3.3.1 *Siano* op : $C^n \to C$ *una funzione monotona e* $\{\rho_i\}_{i \in I} \subseteq \mathrm{uco}(C)$ *precisi rispetto ad* op*. Allora anche* $\rho = \bigsqcap_{i \in I} \rho_i$ *è preciso rispetto ad* op.

Dimostrazione. In pratica dobbiamo dimostrare che $\rho(\mathrm{op}(\rho(\boldsymbol{v}))) = \rho(\mathrm{op}(\boldsymbol{v}))$ per ogni $\boldsymbol{v} \in C^n$. La direzione $\sqsupseteq$ è banale per monotonia di ρ. Nell'altro verso, tenendo conto del teorema 3.1.1, basta dimostrare $\mathrm{op}(\rho(\boldsymbol{v})) \sqsubseteq \rho(\mathrm{op}(\boldsymbol{v}))$. Sia $\boldsymbol{v} = \langle v_1, \ldots, v_n \rangle$, si ricava:

$$\begin{array}{lcl} \rho(\mathrm{op}(v_1, \ldots, v_n)) & = & [\text{per definizione di glb}] \\ \bigsqcap_{i \in I} \rho_i(\mathrm{op}(v_1, \ldots, v_n)) & \sqsupseteq & [\text{per le ipotesi e teorema 3.1.1}] \end{array}$$

$$
\begin{array}{lll}
\prod_{i\in I} \mathrm{op}(\rho_i(v_1),\ldots,\rho_i(v_n)) & \sqsupseteq & [\text{per il teorema 3.1.2}] \\
\mathrm{op}(\prod_{i\in I} \rho_i(v_1),\ldots,\prod_{i\in I} \rho_1(v_n)) & = & [\text{per def. di glb}] \\
\mathrm{op}(\rho(v_1),\ldots,\rho(v_n)) & &
\end{array}
$$

proprio come desiderato. ∎

Osservazione 3.3.2 Si noti che nella dimostrazione usiamo la proprietà che i ρ_i sono precisi rispetto ad op in maniera *locale*. Ciò vuol dire che fissato un singolo $\boldsymbol{v} \in C^n$, $\rho_i(\mathrm{op}(\rho_i(\boldsymbol{v}))) = \rho_i(\mathrm{op}(\boldsymbol{v}))$ per ogni i implica $\rho(\mathrm{op}(\boldsymbol{v})) = \rho(\mathrm{op}(\rho(\boldsymbol{v})))$, senza bisogno della precisione di op su tutto C^n. ∎

Lemma 3.3.3 *Sia* $\mathrm{op} : C \to C$ *una funzione monotona e additiva e* $\{\rho_i\}_{i\in I} \subseteq \mathrm{uco}(C)$ *precisi rispetto ad* op. *Allora anche* $\rho = \bigsqcup_{i\in I} \rho_i$ *è preciso rispetto ad* op.

Dimostrazione. Analogamente a quanto visto per il teorema 3.3.1 una delle diseguaglianze che si ottengono è ovvia. Supponiamo allora, per assurdo, che esista $x \in C$ tale che $\rho(\mathrm{op}(\rho(x))) \sqsupset \rho(\mathrm{op}(x))$. Si consideri l'insieme:

$$Z = \{z_j\}_{j\in J} = \{y \in C \mid x \sqsubseteq y \sqsubseteq \rho(x),\ \rho(\mathrm{op}(y)) = \rho(\mathrm{op}(x))\}$$

che è non vuoto, dato che $x \in Z$. Si possono verificare le seguenti proprietà:

1. $\rho_i(z_j) \sqsubseteq \rho(x)$, per tutti gli $i \in I$ e $j \in J$,

2. $\rho(\bigsqcup Z) = \rho(x)$, $\rho_i(\bigsqcup Z) \sqsubseteq \rho(x)$ per tutti gli $i \in I$,

3. $\bigsqcup Z \in Z$, ovvero $\rho(\mathrm{op}(\sqcup Z)) = \rho(\mathrm{op}(x))$. La direzione $\sqsupseteq$ è banale perché $\bigsqcup Z \sqsupseteq x$. D'altronde $\rho(\mathrm{op}(x)) \sqsupseteq \mathrm{op}(z_j)$ per ogni $j \in J$ implica $\rho(op(x)) \sqsupseteq \bigsqcup_j \mathrm{op}(z_j) = \mathrm{op}(\sqcup Z)$ per additività di op. Dal teorema 3.1.1 segue la diseguaglianza cercata.

A questo punto verifichiamo che esiste almeno un ρ_i per cui op non è preciso. Intanto esiste $j \in J$ t.c. $\sqcup Z \notin \rho_j(C)$, altrimenti sarebbe $\rho(x) \sqsubseteq \sqcup Z$ e di conseguenza $\rho(\mathrm{op}(\rho(x))) \sqsubseteq \rho(\mathrm{op}(\sqcup Z)) = \rho(\mathrm{op}(x))$ contro la nostra ipotesi. Per tale j si ha che $\rho_j(\mathrm{op}(\rho_j(\sqcup Z))) = \rho_j(\mathrm{op}(k))$ con $k \sqsupseteq \sqcup Z$. Ma, per definizione di Z, si ha allora $\rho(\mathrm{op}(k)) \sqsupseteq \rho(\mathrm{op}(\sqcup Z))$ e poiché $k \sqsupset \sqcup Z$, per il teorema 3.2.2, $\rho_j(\mathrm{op}(\rho_j(\sqcup Z))) = \rho_j(\mathrm{op}(k)) \sqsupset \rho_j(op(\sqcup Z))$. ■

Osservazione 3.3.4 Si noti che l'unico utilizzo dell'additività di op si ha nella dimostrazione del punto 3. È possibile allora alleggerire leggermente le ipotesi richiedendo l'addititività solo sugli insiemi $\rho^{-1}(x)$ al variare di $x \in \rho(C)$. ■

Teorema 3.3.5 *Siano* op : $C^n \to C$ *una funzione monotona e additiva,* $\{\rho_i\}_{i\in I} \subseteq \mathrm{uco}(C)$ *precisi rispetto ad* op. *Allora anche* $\rho = \bigsqcup_{i\in I} \rho_i$ *è preciso rispetto ad* op.

Dimostrazione. Lo si dimostra per induzione su n. Il lemma 3.3.3 è equivalente al teorema per $n = 1$. Se $n > 1$, fissato $\boldsymbol{v} = \langle v_1, \ldots, v_n\rangle \in C^n$, sia $\mathrm{op}^n(x) = \mathrm{op}(\rho(v1), \ldots, \rho(v_{n-1}), x)$. Chiaramente op^n è additivo e i ρ_i sono precisi rispetto ad esso. Infatti:

$$\begin{array}{lcl}
\rho_i(\mathrm{op}_n(x)) & = & [\text{per def. di } \mathrm{op}^n] \\
\rho_i(\mathrm{op}(\rho(v_1), \ldots, \rho(v_{n-1}), x)) & = & [\text{per precisione di } \rho_i] \\
\rho_i(\mathrm{op}(\rho_i(\rho(v_1)), \ldots, \rho_i(\rho(v_{n-1})), \rho_i(x))) & = & [\text{per il teorema 3.2.1}] \\
\rho_i(\mathrm{op}(\rho(v_1), \ldots, \rho(v_{n-1}), \rho_i(x))) & = & [\text{per def. di } \mathrm{op}^n] \\
\rho_i(\mathrm{op}^n(\rho_i(x))) & &
\end{array}$$

Per il lemma 3.3.3 segue

$$\rho(\mathrm{op}(\rho(v_1), \ldots, \rho(v_{n-1}), v_n)) = \rho(\mathrm{op}(\rho(v_1), \ldots, \rho(v_{n-1}), \rho(v_n))). \qquad (1)$$

Adesso poniamo $\mathrm{op}^1(x_1, \ldots, x_{n-1}) = \mathrm{op}(x_1, \ldots, x_{n-1}, v_n)$. Ancora una volta op^1 è additivo e preciso rispetto ai ρ_i. Infatti:

$$\begin{array}{lcl}
\rho_i(\mathrm{op}^1(x_1, \ldots, x_{n-1})) & = & [\text{per def. di } \mathrm{op}^1] \\
\rho_i(\mathrm{op}(x_1, \ldots, x_{n-1}, v_n)) & = & [\text{per la precisone dei } \rho_i] \\
\rho_i(\mathrm{op}(\rho_i(x_1), \ldots, \rho_i(v_{n-1}), \rho_i(v_n))) & \sqsupseteq & [\text{estensionalità e monotonia}] \\
\rho_i(\mathrm{op}(\rho_i(x_1), \ldots, \rho_i(v_{n-1}), v_n)) & = & [\text{per def. di } \mathrm{op}^1]
\end{array}$$

$$\rho_i(\mathrm{op}^1(\rho_i(x_1), \ldots, \rho_i(x_{n-1})))$$

mentre la diseguaglianza inversa è banale. Si ottiene allora

$$\rho(\mathrm{op}^1(v_1, \ldots, v_{n-1})) = \rho(\mathrm{op}^1(\rho(v_1), \ldots, \rho(v_{n-1}))) \tag{2}$$

ovvero

$$\rho(\mathrm{op}(v_1, \ldots, v_{n-1}, v_n)) = \rho(\mathrm{op}(\rho(v_1), \ldots, \rho(v_{n-1}), v_n)) \tag{3}$$

che, affiancata con la dimostrazione di prima, ci dà la tesi. ■

La dimostrazione appena vista, in particolare il lemma 3.3.3, sfrutta la caratterizzazione non costruttiva dell'operazione di upper bound su operatori di chisura. Vediamo qui un'altra dimostrazione del lemma che utilizza l'induzione transfinita e la caratterizzazione del corollario 3.2.3.

Lemma 3.3.6 *Sia* $\mathrm{op} : C \to C$ *una funzione monotona e additiva e* $\{\rho_i\}_{i\in I} \subseteq \mathrm{uco}(C)$ *precisi rispetto ad* op. *Allora anche* $\rho = \bigsqcup_{i\in I} \rho_i$ *è preciso rispetto ad* op.

Dimostrazione. Come al solito, poiché una diseguaglianza è banale, per il teorema 3.1.1, basta dimostrare che $\rho(\mathrm{op}(x)) \sqsupseteq \mathrm{op}(\rho(x))$ per ogni $x \in C$. Per il corollario 3.2.3, ponendo $T(y) = \bigsqcup_{i\in I} \rho_i(y)$, si ottiene, per opportuni ordinali γ_1, γ_2:

$$\rho(\mathrm{op}(x)) = T^{\gamma_1}(\mathrm{op}(x))$$
$$\mathrm{op}(\rho(x)) = \mathrm{op}(T^{\gamma_2}(x))$$

A questo punto basta dimostrare che $T^\alpha(\mathrm{op}(x)) \sqsupseteq \mathrm{op}(T^\alpha(x))$ per ogni α. Lo si fa per induzione transfinita:

passo base

$$T^0(\mathrm{op}(x)) = \mathrm{op}(x) = \mathrm{op}(T^0(x))$$

passo induttivo

$$T^{\alpha+1}(\mathrm{op}(x)) \quad = \quad [\text{per def. delle potenze ordinali}]$$

$$
\begin{array}{lll}
T(T^{\alpha}(\mathrm{op}(x))) & \sqsupseteq & [\text{per ipotesi induttiva}] \\
T(\mathrm{op}(T^{\alpha}(x))) & = & [\text{per def. di } T] \\
\bigsqcup_{i\in I} \rho_i(\mathrm{op}(T^{\alpha}(x))) & \sqsupseteq & [\text{per le ipotesi del lemma}] \\
\bigsqcup_{i\in I} \mathrm{op}(\rho_i(T^{\alpha}(x))) & = & [\text{per additività di op}] \\
\mathrm{op}\left(\bigsqcup_{i\in I} \rho_i(T^{\alpha}(x))\right) & = & [\text{per def. di } T] \\
\mathrm{op}(T(T^{\alpha}(x))) & = & [\text{per def. delle potenze ordinali}] \\
\mathrm{op}(T^{\alpha+1}(x)) & &
\end{array}
$$

passo di induzione transfinita

$$
\begin{array}{lll}
T^{\alpha}(\mathrm{op}(x)) & = & [\text{per def. delle potenze ordinali}] \\
\bigsqcup_{\beta<\alpha} T^{\beta}(\mathrm{op}(x)) & \sqsupseteq & [\text{per ipotesi induttiva}] \\
\bigsqcup_{\beta<\alpha} \mathrm{op}(T^{\beta}(x)) & = & [\text{per additività di op}] \\
\mathrm{op}\left(\bigsqcup_{\beta<\alpha} T^{\beta}(x)\right) & = & [\text{per def. delle potenze ordinali}] \\
\mathrm{op}(T^{\alpha}(x)) & &
\end{array}
$$

■

3.4 Operatori di chiusura trasparenti

Esaminiamo ora una proprietà degli uco che si affianca alla precisione.

Definizione 3.4.1 *Sia ρ un uco su C e consideriamo* $\mathrm{op} : C^n \to C$. *Si dice che ρ è* trasparente *rispetto a* op *(o, viceversa, che* op *è trasparente rispetto a ρ) se $\forall x \in C.\ \rho(\mathrm{op}(\rho(\boldsymbol{x}))) = \mathrm{op}(\rho(\boldsymbol{x}))$.*

Si ricorda che l'operatore astratto ottimale si definisce come $\tilde{\mathrm{op}}(\boldsymbol{x}) = \rho(\mathrm{op}(\boldsymbol{x}))$ per ogni $\boldsymbol{x} \in \rho(C)^n$. Quando ρ è trasparente rispetto ad op si ottiene l'eguaglianza tra $\tilde{\mathrm{op}}$ e $\mathrm{op}|_{\rho(C)^n}$.

Il fatto che ρ sia preciso rispetto ad op può essere letto, invece, introducendo la seguente relazione di equivalenza su C:

$$x \approx_\rho y \Leftrightarrow \rho(x) = \rho(y)$$

e dicendo che $\approx_\rho$ è una congruenza rispetto ad op, ovvero che op è ben definito su $C_{/\approx_\rho}$.

Se un uco ρ è sia preciso che trasparente rispetto a op, si ha $\mathrm{op}(\rho(\boldsymbol{x})) = \rho(\mathrm{op}(\boldsymbol{x}))$, ovvero op assume gli stessi valori su tutti gli elementi $\boldsymbol{x} \in C^n$ appartenenti alla stessa classe di equivalenza.

Teorema 3.4.2 *Siano* op_1 *e* op_2 *due operatori su* C *e* ρ *un uco trasparente rispetto ad essi. Allora* ρ *è trasparente anche rispetto a* $\mathrm{op}_2 \circ \mathrm{op}_1$.

Dimostrazione. Dato un qualunque $x \in C$ si ha:

$$\begin{array}{lcl} \rho(\mathrm{op}_2(\mathrm{op}_1(\rho(x)))) & = & [\text{trasparenza di } \mathrm{op}_1] \\ \rho(\mathrm{op}_2(\rho(\mathrm{op}_1(\rho(x))))) & = & [\text{trasparenza di } \mathrm{op}_2] \\ \mathrm{op}_2(\rho(\mathrm{op}_1(\rho(x)))) & = & [\text{trasparenza di } \mathrm{op}_1] \\ \mathrm{op}_2(\mathrm{op}_1(\rho(x))) & & \end{array}$$

come volevasi dimostrare. ■

Corollario 3.4.3 *Dati* ρ *un uco su* C *trasparente rispetto a* op_1 *e* op_2*, abbiamo* $\tilde{\mathrm{op}}_1 \circ \tilde{\mathrm{op}}_2 = (\mathrm{op}_1 \circ \mathrm{op}_2)|_{\rho(C)} = \widetilde{\mathrm{op}_1 \circ \mathrm{op}_2}$

Questi ultimi risultati si estendono ovviamente in maniera banale al caso di operatori su C di arità maggiore di uno.

3.5 Completamento disgiuntivo

Abbiamo visto come gli operatori lub e glb sul reticolo degli operatori di chiusura consentano di costruire uco componendo altri uco di base. Esistono vari operatori di questo tipo nell'ambito della teoria dell'interpretazione astratta, uno di essi è il *completamento disgiuntivo* [GR96, CC79].

Definizione 3.5.1 (Completamento disgiuntivo) *Sia C un reticolo completo e ρ un uco su di esso. Si definisce completamente disgiuntivo di ρ l'operatore di chiusura:*

$$\mho_C \rho = \bigsqcup \{\rho' \sqsubseteq \rho \mid \rho' \text{ è additivo}\}$$

È possibile trovare una caratterizzazione alternativa del completamento disgiuntivo, che sia parzialmente costruttiva, ricorrendo ancora una volta all'induzione transfinita.

Teorema 3.5.2 *Sia C un reticolo completo e $T(S) = \mathrm{lms}\{\bigsqcup S' \mid S' \subseteq S\}$ un operatore sulle famiglie di Moore di C, con $\mathrm{lms}(X)$ ad indicare la più piccola famiglia di Moore che contiene X. Allora, se ρ è un uco su C tale che $S = \rho(C)$, si ha:*

1. *T è monotono ed estensivo,*
2. *$(\mho_C \rho)(C) = \bigcap \{S' \supseteq S \mid T(S') = S'\}$*

ovvero $(\mho_C \rho)(C) = T^\alpha(S)$ per un opportuno ordinale α. Se inoltre $\rho(C)$ è completamente inf-distributivo, allora $\alpha = 1$.

Dimostrazione. Cominciando dal punto 1, data una qualsiasi famiglia di Moore S di C, verifichiamo che $T(S) \supseteq S$. Prendendo $S' = \{x\}$ con $x \in S$ si ottiene che $\{\bigsqcup S' \mid S' \subseteq S\} \supseteq S$ da cui $T(S) \supseteq S$ per estensionalità di lms. La monotonia di T è anch'essa banale considerando che se $S_1 \supseteq S_2$ ogni sottoinsieme di S_2 è anche sottoinsieme di S_1 e sfruttando la monotonia di lms.

poiché T è un operatore monotono, ha un reticolo completo di punti fissi. Se verifichiamo che un uco è additivo se e solo se la sua immagine è punto fisso di $T(S)$ la segue segue banalmente. Sappiamo che se ρ è additivo, l'immagine $\rho(C)$ è un sottoreticolo completo di C per cui $\{\bigsqcup S' \mid S' \subseteq \rho(C)\} = \rho(C)$ che è una famiglia di Moore da cui $T(\rho(C)) = \rho(C)$. Se, invece, ρ non è additivo, esiste $S' \subseteq \rho(C)$ tale che $\bigsqcup S' \notin \rho(C)$ e quindi $T(\rho(C)) \supset \rho(C)$.

Infine, supponiamo $S = \rho(C)$ completamente inf-distributivo. Vuol dire che $S'' = \{\bigsqcup S' \mid S' \subseteq S\}$ è già un sottoreticolo completo additivo di C, oltre che famiglia di Moore, per cui $T(T(S)) = T(S)$. Infatti:

- Sia $\{x_i\}_{i\in I} \subseteq S''$, allora $\bigsqcap_C\{x_i\}_{i\in I} = \bigsqcap_C\{\bigsqcup_C S_i' \mid x_i = \bigsqcup_C S_i',\ S_i' \subseteq S, i \in I\}$. poiché S è completamente inf-distributivo, si può riscrivere il tutto come

$$\bigsqcup\Big\{\bigsqcap_{i\in I}\phi(i) \mid \phi : I \to S \text{ t.c. } \forall i \in I.\ \phi(i) \in S_i\Big\}$$

  che è un elemento di S'' in quanto $\bigsqcup_{i\in I}\phi(i)$ è in S, che è una famiglia di Moore. Dunque S'' è chiuso per intersezione e contiene evidentemente anche il $\top_C$, dato che S lo contiene, per cui è anch'esso una famiglia di Moore.

- Analogamente S'' è chiuso per lub. Sia infatti $x = \bigsqcup_{i\in I}\{x_i\}$ con $x_i \in S''$. poiché $x_i = \bigsqcup S_i$ con $S_i \subseteq S$, si ha $x = \bigsqcup_{i\in I}\bigsqcup S_i \in S''$ come desiderato.

■

In genere se un certo uco op è preciso rispetto a op, nulla ci assicura che anche $\mho\rho$ sia preciso. Le cose cambiano, però, quando ci si limita a considerare operatori additivi su C.

Lemma 3.5.3 *Sia C un reticolo completo, ρ' un uco su C e $\rho = \mho_C\rho'$. Sia ancora* op *un operatore su C additivo e preciso rispetto a ρ'. Allora* op *è preciso rispetto a ρ.*

Dimostrazione. Al solito basta dimostrare che $\mathrm{op}(\rho(x)) \sqsubseteq \rho(\mathrm{op}(x))$. Lo si fa per induzione transfinita. Si pone ρ^α tale che $\rho^\alpha(C) = T^\alpha(\rho'(C))$ per ogni α e si dimostra che op è preciso per ρ^α qualunque sia α.

passo base) Banale perche $\rho^0(C) = \rho(C)$ per cui $\rho^0 = \rho$.

passo induttivo) Ci possiamo limitare a dimostrare che T trasforma uco precisi in uco precisi. Sia allora $\eta(C) = T(\eta'(C))$ con η' preciso per op.

$$\begin{aligned}
&\eta(x) && =\\
&\qquad [\text{def. di operatore di chiusura}] &&\\
&\textstyle\bigsqcap\{y \in T(\eta'(C)) \mid x \sqsubseteq y\} && =\\
&\qquad [\text{def. di T}] &&
\end{aligned}$$

$$\bigsqcap\Big\{y \mid \exists\Delta \subseteq \mathcal{P}(\eta'(C)) \text{ t.c. } y = \bigsqcap_{S'\in\Delta}\{\sqcup S' \mid x \sqsubseteq y\}\Big\} \quad =$$

[per la teoria degli insiemi]

$$\bigsqcap\{\sqcup S' \mid S' \subseteq \eta'(C),\ x \sqsubseteq \sqcup S'\} \quad =$$

[perché $x \sqsubseteq \sqcup S'$ implica $\sqcup(S' \cup \{x\}) = \sqcup S'$]

$$\bigsqcap\{\sqcup S' \mid S' \subseteq \eta'(C),\ x \in S'\} \quad =$$

[per definizione di uco]

$$\bigsqcap\{\sqcup\eta'(S') \mid S' \subseteq C, x \in \eta'(S')\} \quad =$$

[perché $x \in S' \Rightarrow x \in \eta'(S')$ e $\eta'(S') \subseteq C$]

$$\bigsqcap\{\sqcup\eta'(S') \mid S' \subseteq C, x \in S'\}$$

A questo punto si ottiene

$$\mathrm{op}(\eta(x)) \quad =$$

[per quanto visto sopra]

$$\mathrm{op}\Big(\bigsqcap\{\sqcup\eta'(S') \mid S' \subseteq C, x \in S'\}\Big) \quad \sqsubseteq$$

[per il teorema 3.1.2]

$$\bigsqcap\{\mathrm{op}(\sqcup\eta'(S') \mid S' \subseteq C, x \in S'\}) \quad \sqsubseteq$$

[per additività e precisione di op]

$$\bigsqcap\{\sqcup\eta'(\mathrm{op}(S')) \mid S' \subseteq C, x \in S'\}) \quad \sqsubseteq$$

[perché se $\mathrm{op}(x) \in S'$ allora $x \in \mathrm{op}^{-1}(S')$]

[e $\sqcup\eta'(\mathrm{op}(\mathrm{op}^{-1}(S'))) = \sqcup\eta'(S')$]

$$\bigsqcap\{\sqcup\eta'(S') \mid S' \subseteq C, \mathrm{op}(x) \in S'\} \quad =$$

[per quanto visto sopra]

$$\eta(\mathrm{op}(x))$$

come volevasi dimostrare.

passo di induz. transfinita) Basta dimostrare che se $\{\rho_i\}_{i\in I}$ è un insieme di uco precisi per op, allora l'uco ρ tale che $\rho(C) = \bigcup\{\rho_i(C)\}_{i\in I}$, ovvero $\rho = \bigsqcap\{\rho_i\}_{i\in I}$ è perfetto per op. Ma questo è già stato dimostrato nel teorema 3.3.1. ∎

Teorema 3.5.4 *Siano* op : $C^n \to C$ *un operatore additivo e* $\rho \in \text{uco}(C)$ *preciso rispetto ad* op. *Allora anche* $\mho\rho$ *è preciso rispetto ad* op.

Dimostrazione. La dimostrazione procede come quella del teorema 3.3.5 utilizzando il lemma 3.5.3 invece del 3.3.3. ∎

3.6 Dipendenze funzionali

Esaminiamo ora un altro operatore sugli operatori di chiusura, quello delle *dipendenze funzionali* [GR95].

Definizione 3.6.1 *Sia* ρ_1 *e* ρ_2 *due uco sul reticolo completo* C *e* $\odot$ *un operatore binario su* C *additivo a sinitra. Allora, poniamo:*

$$(\rho_1 \to^{\odot} \rho_2)(x) = \bigsqcup\{x' \in C \mid \rho_2(x' \odot y) \sqsubseteq \rho_2(x \odot y) \forall y \in \rho_1(C)\}$$

e si dimostra che $\rho_1 \to^{\odot} \rho_2$ *è anch'esso un operatore di chiusura su* C*, detto delle* dipendenze funzionali tra ρ_1 e ρ_2 relativamente ad op. *Se* $\rho_1 = \rho_2 = \rho$ *si parla di* autodipendenza su ρ *e si indica con* $Dep^{\odot}\,\rho'$.

Come da noi definita, la nozione di dipendenza funzionale è analoga a quella in [GR95], semplicemente riadattata per utilizzare gli operatori di chiusura invece delle iniezioni di Galois. Si noti che se $\rho = \rho_1 \to^{\odot} \rho_2$, vale

$$\rho_2(\rho(x) \odot y) = \rho_2(x \odot y) \tag{3.6.1}$$

per ogni $y \in \rho_1(C)$ e $x \in C$.

Teorema 3.6.2 *Nelle ipotesi della definizione 3.6.1, sia* $\odot$ *preciso sul primo argomento rispetto a* ρ_2*. Allora* $\rho = \rho_1 \to^{\odot} \rho_2 \sqsupseteq \rho_2$*.*

Dimostrazione. Sotto queste ipotesi si ha $\rho_2(\rho_2(x) \odot y) = \rho_2(x \odot y)$ per cui $\rho_2(x)$ fa parte dell'insieme su cui si fa il lub per determinare $\rho(x)$, da cui la tesi. ∎

Teorema 3.6.3 *Nelle ipotesi della definizione 3.6.1, se esiste* $y \in \rho_1(C)$ *tale che* $x \odot y = x$ *per tutti gli* $x \in C$*, allora* $\rho_1 \to^{\odot} \rho_2 \sqsubseteq \rho_2$*.*

Dimostrazione. Fissato x, sia $x' \in C$ ed $x' \not\sqsubseteq \rho_2(x)$; abbiamo che $\rho_2(x' \odot y) = \rho_2(x') \not\sqsubseteq \rho_2(x) = \rho_2(x \odot y)$ per cui x' non fa parte dell'insieme su cui si fa il lub nella def. di dipendenza funzionale. Tale insieme è composto solo da elementi minori di $\rho_2(x)$ da cui la tesi. ■

Teorema 3.6.4 *Nelle ipotesi della definizione 3.6.1,*

$$\rho_1 \rightarrow^{\odot} \rho_2 \ \sqsupseteq\ \bigsqcup\{\rho' \sqsubseteq \rho_2 \mid \rho' \textit{ è preciso per } \odot \textit{ sul primo argomento}\} \tag{3.6.2}$$

Dimostrazione. Sia $\rho' \sqsubseteq \rho_2$ preciso rispetto a $\odot$ ristretto al primo argomento. Allora $\rho_2 = \eta \circ \rho'$ con η uco su $\rho'(C)$, da cui, per ogni $x, v \in C$:

$$\begin{array}{lcl}
\rho_2(\rho'(x) \odot v) & = & \\
\eta(\rho'(\rho'(x) \odot v)) & = & [\text{precisione di } \rho'\,] \\
\eta(\rho'(x \odot v)) & = & \\
\rho_2(x \odot v) & &
\end{array}$$

Dalla definizione delle dipendenze funzionali segue direttamente la tesi. ■

Teorema 3.6.5 *Sia $\rho' = Dep^{\odot}\rho$ con ρ operatore di chiusura su C. Sia $\odot$ additivo a sinistra e preciso sul secondo argomento relativamente a ρ e $\rho' \sqsupseteq \rho$. Allora $\odot$ è preciso sul primo argomento rispetto a ρ'.*

Dimostrazione. Sotto queste ipotesi, per ogni $y \in C$ abbiamo:

$$\begin{array}{lcl}
\rho(\rho'(x) \odot y) & = & [\text{per la precisione di } \odot\,] \\
\rho(\rho'(x) \odot \rho(y)) & = & [\text{per l'equazione (3.6.1)}\,] \\
\rho(x \odot \rho(y)) & = & [\text{per la precisione di } \odot\,] \\
\rho(x \odot y) & &
\end{array}$$

A questo punto, supposto $\rho' = \eta \circ \rho$, si applica η alla catena di eguaglianze e si ottiene

$$\rho'(\rho'(x) \odot y) = \rho'(x \odot y) \tag{1}$$

che completa la dimostrazione. ■

Teorema 3.6.6 *Sia* $\rho' = Dep^{\odot}\,\rho$ *con* ρ *uco sul reticolo* C *e* $\odot$ *operatore binario additivo a sinistra. Sia poi* op *un altro operatore di arità* n *sullo stesso reticolo completo* C *e additivo. Allora se per ogni* $y_1, z_1, \ldots, y_n, z_n \in C$ *vale*

$$\big(\forall x \in \rho(C).\ \rho(y_1 \odot x) \sqsubseteq \rho(z_1 \odot x) \wedge \cdots \wedge \rho(y_n \odot x) \sqsubseteq \rho(z_n \odot x)\big) \Longrightarrow$$
$$\big(\forall x \in \rho(C).\ \rho(\mathrm{op}(y_1, \ldots, y_n) \odot x) \sqsubseteq \rho(\mathrm{op}(z_1, \ldots, z_n) \odot x)\big)$$

segue che op *è preciso rispetto a* ρ'*.*

Dimostrazione. Siano $x_1, \ldots, x_n \in C$. Si ha:

$$\begin{array}{ll}
\mathrm{op}(\rho'(x_1), \ldots, \rho'(x_n)) & = \\
\quad [\,\text{definizione di } \rho'\,] & \\
\mathrm{op}\big(\sqcup\{x'_1 \in C \mid \rho(x'_1 \odot y) \sqsubseteq \rho(x_1 \odot y)\ \forall y \in \rho(C)\}, \ldots, & \\
\quad \sqcup\,\{x'_n \in C \mid \rho'(x'_n \odot y) \sqsubseteq \rho(x_n \odot y)\ \forall y \in \rho(C)\}\big) & = \\
\quad [\,\text{additività di op}\,] & \\
\bigsqcup\{\mathrm{op}(x'_1, \ldots, x'_n) \mid \forall i \in [1..n], y \in C.\ \ \rho(x'_i \odot y) \sqsubseteq \rho(x_i \odot y)\} & \sqsubseteq \\
\quad [\,\text{per le ipotesi del teorema}\,] & \\
\bigsqcup\{x \mid \rho(x \odot y) \sqsubseteq \rho(\mathrm{op}(x_1, \ldots, x_n) \odot y)\ \forall y \in C\} & = \\
\quad [\,\text{definizione di } \rho\,] & \\
\rho(\mathrm{op}(x_1, \ldots, x_n)) &
\end{array}$$

A questo punto per il teorema 3.1.1 si ha la tesi. ■

Corollario 3.6.7 *Sia* $\rho' = Dep^{\odot}\,\rho$ *con* ρ *preciso sul secondo argomento di* $\odot$ *e* $\odot$ *goda della proprietà associativa. Allora* ρ' *è preciso rispetto a* $\odot$*.*

Dimostrazione. Siano y_1, z_1, y_2 e z_2 tali che

$$\forall x \in \rho(C).\ \rho(y_1 \odot x) \sqsubseteq \rho(z_1 \odot x) \wedge \rho(y_2 \odot x) \sqsubseteq \rho(z_2 \odot x) \qquad (1)$$

otteniamo, per ogni $x \in \rho(C)$:

$$\begin{array}{lll}
\rho((y_1 \odot y_2) \odot x) & = & [\,\text{associtatività di } \odot\,] \\
\rho(y_1 \odot (y_2 \odot x)) & = & [\,\text{precisione di } \odot\,]
\end{array}$$

$$\begin{array}{lll}
\rho(y_1 \odot \rho(y_2 \odot x)) & \sqsubseteq & [\text{per } (1)] \\
\rho(y_1 \odot \rho(z_2 \odot x)) & \sqsubseteq & [\text{per } (1)] \\
\rho(z_1 \odot \rho(z_2 \odot x)) & = & [\text{precisione e associatività di } \odot] \\
\rho((z_1 \odot z_2) \odot x) & &
\end{array}$$

che, per il teorema 3.6.6, conduce alla tesi. ∎

Capitolo 4

Framework semantico

È arrivato il momento di introdurre il quadro semantico che utilizzeremo per le nostre argomentazioni, presentato originariamente in [Meo96] e [CLM95].

4.1 Introduzione

L'obiettivo è quello di definire un struttura che ci permetta di ragionare sulle proprietà di composizionalità delle derivazioni SLD e delle loro astrazioni (osservabili) nel caso di programmi logici definiti. Ciò che vogliamo essere in grado di trattare è:

- la relazione tra le semantiche operazionali (top-down) e quelle denotazionali (bottom-up),
- l'esistenza di una denotazione per un insieme finito di clausole,
- le proprietà di una denotazione (composizionalità rispetto a vari operatori sintattici, correttezza e minimalità).

Simili problemi sono stati studiati in dettaglio per varie astrazioni delle derivazioni SLD. Quello che si desidera, invece, è un unico schema col quale trattare in maniera uniforme una grande varietà di osservabili.

Per far ciò si sviluppa l'approccio, ideato in [CL94], basato sulla caratterizzazione algebrica degli osservabili, prendendo due semantiche, una denotazionale e un sistema di transizione che definisce le derivazioni SLD. Inoltre si esprimono le proprietà semantiche come proprietà di composizionalità, cosa che porta ad una maggiore flessibilità nella definizione di osservabile. L'utilizzo, poi, di tecniche di interpretazione astratta per modellare gli osservabili ci consente di fornire condizioni molto semplici che garantiscono la validità di vari teoremi di portata generale.

Nella sezione 4.2 definiamo i nostri domini, la semantica operazionale e il sistema di transizione sulle derivazioni SLD in termini di quattro operatori, direttamente collegati alla struttura sintattica del linguaggio. Successivamente, nella sezione 4.3, definiamo gli osservabili come inserzioni di Galois od operatori di chiusura e caratterizziamo due classi di osservabili in termini di semplici proprietà di precisione degli operatori semantici di base.

La ricostruzione di una semantica già esistente o la costruzione di una nuova semantica richiede nel framework solo pochi semplici passi.

1. La formalizzazione della proprietà che vogliamo modellare come una inserzione di Galois tra alberi SLD e un adeguato dominio astratto o come un operatore di chiusura sugli alberi SLD.

2. La verifica di semplici proprietà algebriche che collegano gli operatori semantici alla nostra inserzione di Galois od operatore di chiusura, per assegnare l'osservabile alla giusta classe.

3. A seconda della classe di appartenza, si ricavano automaticamente la semantica denotazionale, il sistema di transizione, la denotazione top-down e/o bottom-up, insieme a vari interessanti teoremi (equivalenza delle due semantiche, composizionalità, correttezza e minimalità delle denotazioni, ecc ...).

4.2 Il framework di base

Per semplicità d'ora in poi, quando parleremo di derivazione SLD, intenderemo una derivazione secondo la regola di selezione sinistra del PROLOG. Ad ogni modo, i risultati che vedremo si possono estendere facilmente a tutte le regole di selezione locali.

4.2.1 Rappresentazione di alberi SLD

La semantica della programmazione logica che utilizziamo, per essere la più precisa possibile e consentirci di definire una grande varietà di osservabili, è modellata sugli alberi SLD. Scegliamo di rappresentare tali alberi come insiemi particolari diderivazioni, modulo una relazione di equivalenza che ci consente di astrarre dagli aspetti puramente sintattici rappresentati dai nomi delle variabili nelle clausole renamed apart.

Definiamo un relazione di preordinamento $\preceq$ sulle derivazioni: $d \preceq d'$ sse $\mathrm{first}(d) = \mathrm{first}(d')$ ed esiste una rinominazione σ t.c. $\mathrm{clauses}(d)\sigma$ è un prefisso di $\mathrm{clauses}(d')$. Come tutti i preordinamenti, $\preceq$ induce una relazione di equivalenza su Derivs che indicheremo con $\overset{\mathrm{der}}{\approx}$, definita come segue: $d \overset{\mathrm{der}}{\approx} d' \Leftrightarrow (d \preceq d') \wedge (d' \preceq d)$.

A questo punto su $\mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}}$ si possono definire le seguenti operazioni, indotte da quelle omonime sulle derivazioni SLD:

1. $\mathrm{first} : \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}} \to \mathrm{Goals}$ t.c. $\mathrm{first}([d]) = \mathrm{first}(d)$,
2. $\mathrm{last} : \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}} \to \mathrm{Goals}_{/\equiv}$ t.c. $\mathrm{last}([d]) = [\mathrm{last}(d)]_{\equiv}$,
3. $\mathrm{answer} : \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}} \to \mathrm{Subst}_{/\equiv}$ t.c. $\mathrm{answer}([d]) = [\mathrm{answer}(d)]_{\equiv}$,
4. $\mathrm{result} : \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}} \to \mathrm{Resultants}_{/\equiv}$ t.c. $\mathrm{result}([d]) = [\mathrm{result}(d)]_{\equiv}$.

Il fatto che siano ben definite discende direttamente dal lemma A.2.1. Analogamente la relazione $\preceq$ induce su $\mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}}$ un ordinamento che continueremo ad indicare con $\preceq$.

D'ora in poi, qualora non nascano ambiguità, indicheremo la classe di equivalenza di una derivazione d con d stesso.

4.2.2 Collezioni ed interpretazioni

Definizione 4.2.1 (Insiemi ben formati) *$S \subseteq Derivs_{/\overset{\mathrm{der}}{\approx}}$ è detto* ben formato *sse $d \in S$ implica $\forall d' \in Derivs_{/\overset{\mathrm{der}}{\approx}}$, $d' \preceq d \Rightarrow d' \in S$.*

Con WFS indichiamo tutti i possibili sottoinsiemi ben formati di $\mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}}$, con $\mathrm{WFS}_{\boldsymbol{G}}$ il sottoinsieme di WFS tale che $d \in S \in \mathrm{WFS}_{\boldsymbol{G}} \Rightarrow \mathrm{first}(d) = \boldsymbol{G}$. Infine sarà $\mathrm{PuWFS} = \bigcup_{\boldsymbol{G} \in \mathrm{Goals}} \mathrm{WFS}_{\boldsymbol{G}}$.

Definizione 4.2.2 (Collezioni) *Una* collezione D *è una funzione* $D : Goals \to PuWFS$ *tale che* $D(\boldsymbol{G}) \in WFS_{\boldsymbol{G}}$. *Indichiamo con* $\mathbb{D}$ *il dominio di tutte le collezioni. Esso è un reticolo completo con l'ordinamento* $\sqsubseteq$, *dove* $D \sqsubseteq D'$ *sse* $\forall \boldsymbol{G}.\ D(\boldsymbol{G}) \subseteq D'(\boldsymbol{G})$.

Indichiamo con ϕ l'elemento minimo di $\mathbb{D}$, ovvero la collezione $\lambda \boldsymbol{G}.\emptyset$. Data una collezione D, con la notazione $D\,[\,S/\,\boldsymbol{G}]$ intediamo la collezione D' tale che $D'(\boldsymbol{G}') = D(\boldsymbol{G}')$ se $\boldsymbol{G}' \neq \boldsymbol{G}$, S altrimenti. Infine, Id è la collezione identità $\lambda \boldsymbol{G}.\{\boldsymbol{G}\}$. Poniamo inoltre, per ogni $L \subseteq \mathrm{Goals}$:

$$D|_L\,(\boldsymbol{G}) = \begin{cases} D(\boldsymbol{G}) & \text{se } \boldsymbol{G} \in L \\ \emptyset & \text{se } \boldsymbol{G} \notin L \end{cases} \tag{4.2.1}$$

Ogni collezione in cui tutte le clausole usate nelle derivazioni fanno parte del programma P (collezione in P) è la rappresentazione di una famiglia di alberi-SLD parziali per P, uno per ogni goal $\boldsymbol{G}$. La collezione massima in P rappresenta la famiglia di tutti gli alberi SLD per il programma P. La relazione d'ordine su $\mathbb{D}$ formalizza l'evoluzione del processo di computazione.

Collezioni e insiemi ben formati

L'applicazione $\lambda D. \bigcup_{\boldsymbol{G} \in \mathrm{Goals}} D(\boldsymbol{G})$ è un isomorfismo tra $\mathbb{D}$ e WFS. Possiamo allora indifferentemente vedere le collezioni come funzioni o come semplici insiemi ben formati, a seconda di come ci risulta più comodo. Ad esempio, sfrutteremo questa nuova visione delle collezioni per estendere ad esse le relazioni di equivalenza sulle derivazioni, passando dapprima attraverso l'estensione a WFS.

La corrispondenza che si viene a creare tra operazioni su $\mathbb{D}$ e operazioni su WFS è la seguente:

$$\begin{array}{lcl}
D(\boldsymbol{G}) & = & D \cap \top_{\mathrm{WFS}_G} \\
D \sqsubseteq D' & \Leftrightarrow & D \subseteq D' \\
D|_L & = & D \cap \bigcup_{\boldsymbol{G} \in L} \top_{\mathrm{WFS}_{\boldsymbol{G}}} = \bigcup_{\boldsymbol{G} \in L} D(\boldsymbol{G}) \\
\lambda \boldsymbol{G}.E(\boldsymbol{G}) & = & \bigcup_{\boldsymbol{G} \in \mathrm{Goals}} E(\boldsymbol{G})
\end{array}$$

dove $D, D' \in \mathbb{D}$, $\boldsymbol{G} \in \mathrm{Goals}$, $L \subseteq \mathrm{Goals}$ ed $E(\boldsymbol{G})$ è una espressione contenente un certo numero di occorrenze di $\boldsymbol{G}$.

Collezioni ed interpretazioni

Le collezioni mantengono distinti gli alberi SLD per goals che sono tra loro varianti. Talvolta, però, si vorrebbero astrarre le derivazioni e gli alberi SLD anche rispetto al goal inziale. Si definisce allora la relazione di varianza $\equiv$ sulle derivazioni. Per ogni $d, d' \in \mathrm{Derivs}$:

$$d \equiv d' \text{ sse } \langle \mathrm{first}(d), \mathrm{clauses}(d) \rangle \equiv \langle \mathrm{first}(d'), \mathrm{clauses}(d') \rangle \tag{4.2.2}$$

Si verifica immediatamente che $d \overset{\mathrm{der}}{\approx} d' \Rightarrow d \equiv d'$. Ciò vuol dire che è possibile definire $\equiv$ anche su $\mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}}$ ponendo $[d]_{\overset{\mathrm{der}}{\approx}} \equiv [d']_{\overset{\mathrm{der}}{\approx}}$ sse $d \equiv d'$ e, quindi, pure sulle collezioni, viste come insiemi ben formati. Si noti in particolare che $[d] \equiv [d']$ e $\mathrm{first}(d) = \mathrm{first}(d')$ sse $[d] = [d']$.

Nello sviluppo del framework ci piacerebbe disporre di una denotazione per i programmi che modelli gli alberi SLD. Vedremo come, in generale, la denotazione di un programma sia una collezione di tipo particolare, definita solo per i *goal atomici puri*, ovvero per quelli del tipo $p(\boldsymbol{x})$ con $\boldsymbol{x}$ tupla di variabili distinte.

Definizione 4.2.3 (Interpretazione) *Una* intepretazione I *è una collezione, modulo varianza, per cui* $I(\boldsymbol{G}) = \emptyset$ *se* $\boldsymbol{G}$ *non è un goal atomico puro. Indichiamo con* $\mathbb{I}$ *l'insieme di tutte le interpretazioni, che è un reticolo completo con l'ordinamento indotto dalle collezioni. Si definisce la restrizione su* $\mathbb{I}$ *di una collezione* D *come segue:*

$$D|_{\mathbb{I}} = \left[\bigsqcup \{ D(\boldsymbol{G}) \mid \boldsymbol{G} \text{ è un goal atomico puro} \} \right]_{\equiv}$$

Si noti che ogni clausola $c = p(\boldsymbol{t}) \leftarrow \boldsymbol{B}$ può essere vista come una derivazione ad un passo, e quindi anche come l'interpretazione

$$\operatorname{tree}(c) = \Big[\phi\big[\,\{p(\boldsymbol{x}), p(\boldsymbol{x}) \xrightarrow[c]{\{\boldsymbol{x}/\boldsymbol{t}\}} \boldsymbol{B}\}\Big/\, p(\boldsymbol{x})\big]\Big]_{\equiv} \tag{4.2.3}$$

dove $\boldsymbol{x}$ è una tupla di variabili distinte renamed apart da c. Dato un programma P, poniamo:

$$\operatorname{tree}(P) = \Big[\sum_{c\in P} \operatorname{tree}(c)\Big]_{\equiv} \tag{4.2.4}$$

4.2.3 Semantica denotazionale

La semantica denotazionale è costituita da un certo numero di funzioni semantiche corrispondenti alle varie categorie sintattiche.

$$\begin{array}{lll}
\mathcal{Q} : QUERY & \longrightarrow & \mathbb{D}, \\
\mathcal{G} : GOAL & \longrightarrow & (\mathbb{I} \to \mathbb{D}), \\
\mathcal{A} : ATOM & \longrightarrow & (\mathbb{I} \to \mathbb{D}), \\
\mathcal{P} : PROG & \longrightarrow & (\mathbb{I} \to \mathbb{I}), \\
\mathcal{C} : CLAUSE & \longrightarrow & (\mathbb{I} \to \mathbb{I}).
\end{array}$$

Ogni elemento $I \in \mathbb{I}$ sarà per noi non solo una interpretazione ma anche una collezione ottenuta scegliendo un elemento arbitrario della classe di equivalenza. Tutte le operazioni su interpretazioni saranno in effetti definite su collezioni facendo attenzione che risultino indipendenti dal rappresentante scelto.

Le funzioni semantiche si baseranno su una serie di operatori semantici $\cdot$, $\times$, $\bowtie$, $\sum$, indotti dalle operazioni sintattiche e dalla natura composizionale delle definizioni in stile denotazionale. Il loro significato informale è il seguente. L'operatore $\cdot$ "risolve" un goal atomico A in una interpretazione I. L'operatore $\times$ calcola la congiunzione di due denotazioni. L'operatore $\bowtie$ calcola le denotazioni ottenute per rimpiazzamento e infine $\sum$ esegue l'unione non deterministica di un insieme di denotazioni. Quando tale insieme è finito si usa la notazione infissa $+$.

$$
\begin{aligned}
\mathcal{Q}[\![\boldsymbol{G}\ \texttt{in}\ P]\!] &= \mathcal{G}[\![\boldsymbol{G}]\!]_{\mathrm{lfp}\,\mathcal{P}[\![P]\!]} && (4.2.5)\\
\mathcal{G}[\![\square]\!]_I &= Id|_{\square} && (4.2.6)\\
\mathcal{G}[\![A,\boldsymbol{G}]\!]_I &= \mathcal{A}[\![A]\!]_I \times \mathcal{G}[\![\boldsymbol{G}]\!]_I && (4.2.7)\\
\mathcal{A}[\![A]\!]_I &= A\cdot I && (4.2.8)\\
\mathcal{P}[\![\emptyset]\!]_I &= Id|_{\mathbb{I}} && (4.2.9)\\
\mathcal{P}[\![\{c\}\cup P]\!]_I &= \big[\mathcal{C}[\![c]\!]_I + \mathcal{P}[\![P]\!]_I\big]_{\equiv} && (4.2.10)\\
\mathcal{C}[\![p(\boldsymbol{t})\leftarrow \boldsymbol{B}]\!]_I &= \big[\mathrm{tree}(p(\boldsymbol{t})\leftarrow \boldsymbol{B})\bowtie \mathcal{G}[\![\boldsymbol{B}]\!]_I\big]_{\equiv} && (4.2.11)
\end{aligned}
$$

La bontà di queste definizioni dipende dal fatto che la relazione $\equiv$ è una congruenza per tutti gli operatori semantici di base, $\cdot$, $\times$, $\bowtie$ e $\sum$.

Operatori sulle derivazioni

Prima di definire formalmente gli operatori $\cdot$, $\times$, $\bowtie$ e $\sum$, occorre introdurre tre operatori ausiliari sulle derivazioni, che formalizzano le operazioni di concatenamento, instanziazione ed AND-composizione. Gli operatori sulle collezioni saranno poi poco più che versioni collecting di queste.

1. Siano $d_1, d_2 \in$ Derivs tali che $\mathrm{last}(d_1) \equiv \mathrm{first}(d_2)$. Si determini $d_2' \in$ Derivs variante di d_2 tale che $\mathrm{first}(d_2') = \mathrm{last}(d_1)$ e $\mathrm{var}(d_1) \cap \mathrm{var}(d_2') = \mathrm{var}(\mathrm{first}(d_2'))$. Allora $d_1 :: d_2$ denota la derivazione ottenuta dalla concatenazione di d_1 e d_2'.

2. Sia $d \in$ Derivs e γ una sostituzione. Si determini $d' = \boldsymbol{G}_0' \xrightarrow[c_1]{\theta_1'} \cdots \in$ Derivs, t.c. $d \overset{\mathrm{der}}{\approx} d'$ e $\mathrm{var}(\boldsymbol{G}_0'\gamma) \cap \mathrm{var}(\mathrm{clauses}(d')) = \emptyset$. Allora $\partial_\gamma(d) = \boldsymbol{G}_0 \xrightarrow[c_1]{\theta_1} \ldots$ tali che:

   - $\boldsymbol{G}_0 = \boldsymbol{G}_0'\gamma$ e
   - per ogni $0 \le i \le \mathrm{length}(d)$, se $\boldsymbol{G}_{i-1}$ è definito ed esiste $\gamma_i \in \mathrm{mgu}(\boldsymbol{G}_{i-1}, \boldsymbol{G}_{i-1}'\theta_i')$ t.c. $\mathrm{var}(\mathrm{range}(\gamma_i)) \cap \mathrm{var}(\boldsymbol{G}_{i-1}') \subseteq \mathrm{var}(\boldsymbol{G}_{i-1})$, allora
     - $\theta_i = (\theta_i'\gamma_i)|_{(\boldsymbol{G}_{i-1}, c_i)}$ e

- $\boldsymbol{G}_i = \boldsymbol{G}'_i\gamma_i$.

3. Siano $d_1 = \boldsymbol{G}'_0 \xrightarrow[c_1]{\theta_1} \ldots$ e d_2 derivazioni finite. Se esiste k t.c. $\boldsymbol{G}'_k = \Box$ allora $d_1 \wedge d_2 = d' :: \partial_{\theta_1\ldots\theta_n}(d_2)$, altrimenti $d_1 \wedge d_2 = d'$, dove $d' = \boldsymbol{G}_0 \xrightarrow[c_1]{\theta_1} \ldots$ è tale che
   - $\boldsymbol{G}_0 = (\boldsymbol{G}'_0, \mathrm{first}(d_2))$ e,
   - per ogni $1 \leq i \leq \mathrm{length}(d_1)$, $\boldsymbol{G}_i = \boldsymbol{G}'_i, \bar{\boldsymbol{G}}\theta_i$, se esiste $\bar{\boldsymbol{G}}$ t.c. $\boldsymbol{G}_{i-1} = \boldsymbol{G}'_{i-1}, \bar{\boldsymbol{G}}$.

Si noti che questi operatori non sono a rigore ben definiti perché non restituiscono un risultato univoco. Si dimostrano però le seguenti:

Proposizione 4.2.4 *Siano $d_1, d_2 \in$ Derivs, $\gamma \in$ Subst. Vale quanto segue:*

1. *se* $\mathrm{last}(d_1) \equiv \mathrm{first}(d_2)$ *allora* $d_1 :: d_2 \in$ *Derivs,*
2. $\partial_\gamma(d_1) \in$ *Derivs,*
3. $d_1 \wedge d_2 \in$ *Derivs.*

Proposizione 4.2.5 *Siano $d_1, d_2, d_3, d_4 \in$ Derivs tali che $d_1 \overset{der}{\approx} d_2$ e $d_3 \overset{der}{\approx} d_4$, $\gamma \in$ Subst. Vale quanto segue:*

1. *se* $\mathrm{last}(d_1) \equiv \mathrm{first}(d_3)$ *allora* $\mathrm{last}(d_2) \equiv \mathrm{first}(d_4)$ *e* $d_1 :: d_3 \overset{der}{\approx} d_2 :: d_4$,
2. $\partial_\gamma(d_1) \overset{der}{\approx} \partial_\gamma(d_3)$,
3. $d_1 \wedge d_3 \overset{der}{\approx} d_2 \wedge d_4$.

Ciò consente di definire gli stessi operatori anche su $\mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}}$, dominio sul quale, tra l'altro, diventano ben definiti.

Operatori sulle collezioni

E finalmente veniamo alla definizione dei nostri operatori sulle collezioni.

Definizione 4.2.6 *Siano $D, D_1, D_2 \in \mathbb{D}$, $A \in Goals$. Si pone:*

$$\sum_{i \in I} D_i = \bigcup_{i \in I} D_i \tag{4.2.12}$$

$$A \cdot D = \{\partial_\gamma(d) \mid d \in D, \exists \gamma \in Subst \ t.c.\ A = \text{first}(d)\gamma\} \tag{4.2.13}$$

$$D_1 \times D_2 = \{d_1 \wedge d_2 \mid d_1 \in D_1, d_2 \in D_2\} \tag{4.2.14}$$

$$D_1 \bowtie D_2 = D_1 \cup \{d_1 :: d_2 \mid d_1 \in D_1, d_2 \in D_2, \text{last}(d_1) \equiv \text{first}(d_2)\} \tag{4.2.15}$$

Le operazioni sono state definite sfruttando l'isomorfismo tra $\mathbb{D}$ e WFS che porta ad una notevole semplificazione del formalismo rispetto a quanto si trova su [Meo96].

Per evitare un eccessivo uso delle parentesi consideriamo $\times$ prioritario su $\bowtie$ e questo su $\cdot$.

4.2.4 Semantica operazionale

La semantica operazionale delle query al programma P è descritta in termini di un sistema di transizione $\mathcal{T} = (\mathbb{D}, \stackrel{P}{\longmapsto})$. La regola di $\mathcal{T}$ è:

$$\frac{D \in \mathbb{D}, \ \ D \neq D \bowtie \mu(\text{tree}(P))}{D \stackrel{P}{\longmapsto} D \bowtie \mu(\text{tree}(P))} \tag{4.2.16}$$

dove

$$\mu(D) = \sum \{(A \cdot D|_{\mathbb{I}}) \times Id\}_{A \in Atoms} \tag{4.2.17}$$

Gli stati iniziali di $\mathcal{T}$ sono tutte le collezioni con solo derivazioni SLD di lunghezza zero, mentre gli stati finali sono le collezioni di tutte le refutazioni SLD e i fallimenti finiti.

Come suggerisce l'intuizione il sistema di transizione $\mathcal{T}$ definisce la nozione usuale di derivazione SLD. La caratteristica importante è che è definito a partire dagli stessi operatori utilizzati per la semantica denotazionale.

poiché siamo interessati a sapere tutte le derivazioni SLD di un dato goal $\boldsymbol{G}$ in un programma P, definiamo la semantica di una query $\boldsymbol{G}$ `in` P come:

$$\mathcal{B}[\![\boldsymbol{G} \ \texttt{in} \ P]\!] = \sum \{D \mid Id|_{\boldsymbol{G}} \stackrel{P}{\longmapsto}{}^{*} D\} \tag{4.2.18}$$

Vale la seguente relazione tra la nostra semantica operazionale e le derivazioni SLD di un programma P:

Proposizione 4.2.7 *Sia P un programma e $\boldsymbol{G}$ un goal. Allora:*

$$\mathcal{B}[\![\boldsymbol{G} \;\; in \;\; P]\!] = \{[d]_{\underset{\approx}{der}} \mid d = \boldsymbol{G} \xrightarrow[P]{\theta}{}^{*} \boldsymbol{B}\}$$

che è proprio quello che desideravamo.

4.2.5 La denotazione dei programmi

Quello che ora ci serve è una buona nozione di denotazione per un programma logico. La prima cosa che viene in mente di usare è la collezione $\sum\{\mathcal{B}[\![\boldsymbol{G} \;\; \mathtt{in} \;\; P]\!]\}_{\boldsymbol{G}\in\text{Goals}}$. Tuttavia è possibile fare di meglio basandosi soltanto sul comportamento del programma P in risposta ai goal atomici puri modulo varianza.

La *denotazione top-down* di un programma P è l'interpretazione:

$$\mathcal{O}[\![P]\!] = \left[\sum\left\{\mathcal{B}[\![p(\boldsymbol{x}) \;\; \mathtt{in} \;\; P]\!]\right\}_{p\in\Pi}\right]_{\equiv} \tag{4.2.19}$$

con $\boldsymbol{x}$ tupla di variabili distinte.

Dal punto di vista bottom-up, si nota che la funzionae semantica $\mathcal{P}[\![P]\!]$ altro non è se non la versione dell'operatore T_P delle conseguenze immediate nel caso di derivazioni. Si può dimostrare che $\mathcal{P}[\![P]\!]$ è continuo, per cui possiamo definire la *denotazione bottom-up* di un programma P come:

$$\mathcal{F}[\![P]\!] = \operatorname{lfp} \mathcal{P}[\![P]\!] = \mathcal{P}[\![P]\!] \uparrow \omega \tag{4.2.20}$$

Definiamo a questo punto la nozione di equivalenza $\approx$ tra programmi. Dati $P_1, P_2 \in$ Progs, si pone:

$$P_1 \approx P_2 \Leftrightarrow \forall \boldsymbol{G} \in \text{Goals}, \mathcal{B}[\![\boldsymbol{G} \;\; \mathtt{in} \;\; P_1]\!] = \mathcal{B}[\![\boldsymbol{G} \;\; \mathtt{in} \;\; P_2]\!] \tag{4.2.21}$$

Rircordiamo poi le definizioni per collegare l'equivalenza tra programmi alla sua denotazione. Sia $\mathcal{S}[\![S]\!]$ una denotazione di un programma e $\approx$ una relazione si equivalenza. Allora:

1. $\mathcal{S}$ è *corretta* rispetto a $\approx$ se $\mathcal{S}[\![P_1]\!] = \mathcal{S}[\![P_2]\!] \Longrightarrow P_1 \approx P_2$,

2. $\mathcal{S}$ è *minimale* rispetto a $\approx$ se $P_1 \approx P_2 \Longrightarrow \mathcal{S}[\![P_1]\!] = \mathcal{S}[\![P_2]\!]$.

4.2.6 Proprietà semantiche

Cominciamo col vedere alcune semplici proprietà di cui godono i nostri operatori semantici.

Lemma 4.2.8 *$\times$ è associativo, ovvero per ogni $D_1, D_2, D_3 \in \mathbb{D}$, $D_1 \times (D_2 \times D_3) = (D_1 \times D_2) \times D_3$. Inoltre, $(D_1 \bowtie D_2) \bowtie D_3 = D_1 \bowtie ((D_2 \bowtie D_3) + D_3)$ per cui se $D_3 \sqsubseteq D_2$, allora $D_1 \bowtie (D_2 \bowtie D_3) = (D_1 \bowtie D_2) \bowtie D_3$*

Si noti inoltre che per ogni $D' \sqsubseteq Id$, $D \bowtie D' = D$.

Lemma 4.2.9 *$\cdot$, $\times$, $\bowtie$ sono operazioni additive su $\mathbb{D}$.*

Mostriamo ora alcune interessanti proprietà di cui gode la semantica operazionale . Iniziamo con la composizionalità di $\mathcal{B}$ rispetto alle chiamate di procedure (goal atomici) e congiunzione di goal (AND-composizionalità).

Nel seguito, data una collezione D, indichiamo con $\mu_n(D)$ la collezione

$$\mu_n(D) = (\ldots(\mu(D) \bowtie \mu(D))\ldots) \bowtie \mu(D) \tag{4.2.22}$$

dove l'operatore di rimpiazzamento $\bowtie$ è applicato $n - 1$ volte. Si noti che $\mu_1(D) = \mu(D)$ e, per definzione, si pone $\mu_0(D) = \emptyset$.

È importante notare che per ogni collezione D, $\mu(D)$ è chiuso per ridenominazione e instanziazione. Data una collezione D, valgono infatti le seguenti proprietà:

- Se $d = \partial_\sigma(d')$, σ è una rinominazione e $d' \in \mu(D)$, allora $d \in \mu(D)$.
- Se $d = \partial_\gamma(d')$, $\gamma \in$ Subst e $d \in \mu(D)$ allora $\partial_\gamma(d) \in \mu(D)$. Inoltre se $\partial_\gamma(d) \in \mu(D)$ esiste una derivazione $d' \in \mu(D)$ tale che d' è un prefisso di d e $\partial_\gamma(d) = \partial_\gamma(d')$.

Lemma 4.2.10 *Sia P un programma e $\boldsymbol{G} \in$ Goals. Allora:*

1. $\mathcal{B}[\![\boldsymbol{G} \ \boldsymbol{in} \ P]\!] = \sum\{Id|_{\boldsymbol{G}} \bowtie \mu_n(\text{tree}(P))\}_{n\geq 0}$
2. $\mathcal{O}[\![P]\!] = \left[Id|_{\mathbb{I}} \bowtie \sum\{\mu_n(\text{tree}(P))\}_{n\geq 0}\right]_{\equiv}$

Lemma 4.2.11 *Sia A un atomo, D,D_1 e D_2 collezioni e $\boldsymbol{G}$ un goal. Allora, per ogni $n \geq 0$:*

1. $A \cdot (D_1 \bowtie \mu_n(D)) = (A \cdot D_1) \bowtie \mu_n(D)$.

2. $(D_1 \bowtie \mu_n(D)) \times Id|_{\boldsymbol{G}} \sqsubseteq (D_1 \times Id|_{\boldsymbol{G}}) \bowtie \mu_n(D)$.

3. *Se* $D_1 \bowtie \mu_n(D) = D_1$, *allora* $(D_1 \times D_2) \bowtie \mu_n(D) = D_1 \times (D_2 \bowtie \mu_n(D))$

Teorema 4.2.12 *Sia A un atomo, $\boldsymbol{G}_1$, $\boldsymbol{G}_2$ goals e P un programma. Allora:*

1. $\mathcal{B}[\![A \;\; \boldsymbol{in} \;\; P]\!] = A \cdot \mathcal{O}[\![P]\!]$,

2. $\mathcal{B}[\![(\boldsymbol{G}_1, \boldsymbol{G}_2) \;\; \boldsymbol{in} \;\; P]\!] = \mathcal{B}[\![\boldsymbol{G}_1 \;\; \boldsymbol{in} \;\; P]\!] \times \mathcal{B}[\![\boldsymbol{G}_2 \;\; \boldsymbol{in} \;\; P]\!]$.

Corollario 4.2.13 (correttezza e minimalità) *Siano P_1 e P_2 due programmi. Allora:*

$$P_1 \approx P_2 \Longleftrightarrow \mathcal{O}[\![P_1]\!] = \mathcal{O}[\![P_2]\!] \tag{4.2.23}$$

Usando l'operatore $\bowtie$ possiamo definire un operatore $\uplus$ che calcola la composizione OR di due denotazioni. Si pone allora $I_1 \uplus I_2 = (I_1 + I_2)^*$ dove $(I)^*$ è la soluzione minima dell'equazione:

$$(I)^* = \left[Id|_{\mathbb{I}} + ((I)^* \bowtie \mu(I))\right]_{\equiv} \tag{4.2.24}$$

Questo nuovo operatore ci consente di dimostrare la OR-composizionalità di $\mathcal{O}$, grazie al seguente

Teorema 4.2.14 (OR-composizionalità) *Siano P_1 e P_2 due programmi. Allora:*

$$\mathcal{O}[\![P_1 \cup P_2]\!] = \mathcal{O}[\![P_1]\!] \uplus \mathcal{O}[\![P_2]\!]$$

Infine, si può dimostrare che le denotazioni top-down e bottom-up sono tra loro equivalenti. Abbiamo prima bisogno di qualche terminologia e proprietà sulle collezioni. Nel seguito, per semplificare la notazione, dato $D \in \mathbb{D}$, denoteremo con $\nu(D)$ la collezione:

$$\nu(D) = \sum \{\mathcal{G}[\![\boldsymbol{G}]\!]_{D|_{\mathbb{I}}}\}_{\boldsymbol{G} \in \text{Goals}} \tag{4.2.25}$$

Si noti che $\nu(D)$ può essere visto come una sorta di *unfolding parallelo* della collezione più generale contenuta in D. Analogamente $\mu(D)$ è un operatore di *unfolding sequenziale.*

Grazie alla definizione di $\nu(D)$ è possibile rimpiazzare la definizione 4.2.11 di $\mathcal{C}$ con

$$\mathcal{C}[\![c]\!]_I = \left[\text{tree}(c) \bowtie \nu(I)\right]_{\equiv} \tag{4.2.26}$$

ed è facile verificare che:

$$\mathcal{P}[\![P]\!]_I = \left[Id|_{\mathbb{I}} + (\text{tree}(P) \bowtie \nu(I))\right]_{\equiv} \tag{4.2.27}$$

Si noti che $\nu(D)$ è chiusa rispetto a renaming e istanziazione esattamente come $\mu(D)$. Data una collezione D, poniamo

$$\nu_n(D) = (\ldots(\nu(D) \bowtie \nu(D))\ldots) \bowtie \nu(D) \tag{4.2.28}$$

e analogamente

$$\nu^n(D) = \nu(D \bowtie \nu(D \bowtie \nu(\ldots \bowtie \nu(D)))) \tag{4.2.29}$$

dove il rimpiazzamento è eseguito $n-1$ volte. Si noti che $\nu_1(D) = \nu^1(D) = \nu(D)$ e, per definizione, $\nu_0(D) = \nu^0(D) = \emptyset$.

Cominciamo a esaminare le relazioni che legano unfolding sequenziali e parallelo e come l'uno è in grado di approssimare correttamente l'altro.

Lemma 4.2.15 *Sia $D \in \mathbb{D}$. Allora, per ogni $n \geq 0$, $\mu_n(D) \sqsubseteq \nu_n(D + Id|_{\mathbb{I}})$.*

Lemma 4.2.16 *Sia $D \in \mathbb{D}$. Allora $\nu(D) \sqsubseteq Id + \sum\{\mu_n(D)\}_{n \geq 0}$*

Lemma 4.2.17 *Sia $D \in \mathbb{D}$. Allora, $\sum\{\nu^h(D)\}_{h \geq 0} = \sum\{\nu^k(D)\}_{k \geq 0}$.*

Corollario 4.2.18 *Sia P un programma. Allora:*

$$\mathcal{F}[\![P]\!] = \left[(Id|_{\mathbb{I}} + \mathrm{tree}(P)) \bowtie \sum \left\{\nu^n(Id|_{\mathbb{I}} + \mathrm{tree}(P))\right\}_{n\geq 0}\right]_{\equiv}$$

e da qui al risultato cercato:

Teorema 4.2.19 *Sia P un programma. Allora $\mathcal{O}[\![P]\!] = \mathcal{F}[\![P]\!]$.*

Teorema 4.2.20 *Sia P un programma e $\boldsymbol{G}$ un goal. Allora*

$$\mathcal{Q}[\![\boldsymbol{G}\ \ \boldsymbol{in}\ \ P]\!] = \mathcal{B}[\![\boldsymbol{G}\ \ \boldsymbol{in}\ \ P]\!].$$

4.3 Il framework delle astrazioni

Le proprietà che abbiamo trovato per $\mathcal{O}$ e $\mathcal{F}$ ci permettono di affermare che abbiamo una buona denotazione modellante gli alberi SLD. L'obiettivo è di trovare gli stessi risultati per le denotazioni basate su osservabili più astratti.

4.3.1 Gli osservabili

Come già detto modelleremo gli osservabili come iniezioni di Galois tra il dominio semantico $\mathbb{D}$ e un dominio astratto $\mathbb{A}$ o come operatori di chisura su $\mathbb{D}$, seguendo, insomma, la teoria della interpretazione astratta. D'altronde vogliamo che un osservabile sia astrazione di un albero SLD e non di una intera famiglia di alberi, per cui:

1. Definiamo $\mathbb{A}$ come l'insieme delle funzioni $A : \mathrm{Goals} \to \mathcal{D}$, dove $\mathcal{D} = \bigcup_{\boldsymbol{G}\in\mathrm{Goals}} \mathcal{D}_{\boldsymbol{G}}$ e $\mathcal{D}_{\boldsymbol{G}}$ è il dominio delle astrazioni degli alberi SLD per il goal $\boldsymbol{G}$.

2. Dati $\alpha^* : \mathrm{PuWFS} \to \mathcal{D}$ e $\gamma^* : \mathcal{D} \to \mathrm{PuWFS}$ tali che

   (a) $\gamma^*(\alpha^*(\emptyset)) = \emptyset$,

   (b) $\langle \mathrm{WFS}_{\boldsymbol{G}}, \alpha^*, \gamma^*, \mathcal{D}_{\boldsymbol{G}}\rangle$ è una inserzione di Galois per ogni $\boldsymbol{G} \in \mathrm{Goals}$,

   (c) $S \equiv S' \Rightarrow \gamma^*(\alpha^*(S)) \equiv \gamma^*(\alpha^*(S'))$

chiamiamo *osservabile* la G.i. $\langle \mathbb{D}, \alpha, \gamma, A \rangle$ così definita:

$$\alpha(D) = \lambda \boldsymbol{G}.\alpha^*(D(\boldsymbol{G})) \tag{4.3.1}$$
$$\gamma(A) = \lambda \boldsymbol{G}.\gamma^*(A(\boldsymbol{G}))$$

3. L'equivalente caratterizzazione con gli uco si basa su una funzione ρ^* : PuWFS $\to$ PuWFS tale che

   (a) $\rho^*(\emptyset) = \emptyset$,

   (b) ρ^* è un uco su $\mathrm{WFS}_{\boldsymbol{G}}$ per ogni $\boldsymbol{G} \in$ Goals,

   (c) $S \equiv S' \Rightarrow \rho^*(S) \equiv \rho^*(S')$

   utilizzato per definire l'*osservabile* $\rho : \mathbb{D} \to \mathbb{D}$ come segue:

$$\rho(D) = \lambda \boldsymbol{G}.\rho^*(D(\boldsymbol{G})) \tag{4.3.2}$$

4. Indichiamo l'insieme di tutti gli osservabili con **Obs**. La corrispondenza tra osservabili espressi usando inserzioni di Galois od operatori di chisura è evidente. Dal primo si passa al secondo semplicemente ponendo $\rho^* = \gamma^* \circ \alpha^*$, viceversa si definisce $\mathcal{D}_{\boldsymbol{G}} = \rho^*(\mathrm{WFS}_{\boldsymbol{G}})$, $\alpha^* = \rho^*$, $\gamma^* = \lambda x.x$.

La condizione (c) della definizione di osservabile consente di definire una nozione di *varianza astratta* $\equiv_\alpha$ su $\mathbb{A}$, tale che per ogni $X_1, X_2 \in \mathbb{A}$

$$X_1 \equiv_\alpha X_2 \Longleftrightarrow \gamma(X_1) \equiv \gamma(X_2) \tag{4.3.3}$$

L'equivalente $\equiv_\rho$ per gli osservabili definiti tramite operatori di chiusura coincide con la varianza del caso concreto. Ciò ci porta, tra l'altro, alla definizione dell'analogo delle interpretazioni nel dominio astratto. Come nel caso concreto, una interpretazione altro non è che un elemento di $\mathbb{A}$ (o di $\rho(\mathbb{D})$) ristretto ai soli goal atomici puri e quozientato da $\equiv_\alpha$ ($\equiv_\rho$). Dato $X \in \rho(\mathbb{D})$, $X|_{\mathbb{I}}$ è, quindi, oltre che un elemento di $\mathbb{I}$, anche una *interpretazione in senso astratto*. L'insieme di queste intepretazioni si indica con $\rho(\mathbb{I})$.

La nostra definizione di osservabile è leggermente diversa da quella di [Meo96] non solo nella forma ma anche nella varietà di astrazioni che è possibile trattare. Seguendo quello stile avremmo dovuto definire $\alpha^*(S)$ come immagine di una funzione abs : $\mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}} \to \mathcal{D}$, ovvero sarebbe stato $\alpha^*(S) = \mathrm{abs}(S)$. Questo, però, impedisce di avere astrazioni che si basino su una visione globale dell'albero SLD. Ad esempio:

$$\alpha^*(S) = \#\{d \in S \text{ t.c. } \mathrm{last}(d) = \Box\} \tag{4.3.4}$$

che conta quante refutazioni vi sono in una collezione, non sarebbe esprimibile.

Nonostante questo ampliamento della classe degli osservabili, tutti i risultati in [CLM95] rimangono ancora validi, grazie alla aggiunta della condizione $\gamma^*(\alpha^*(\emptyset)) = \emptyset$ o $\rho^*(\emptyset) = \emptyset$. Questa, infatti, ci consente di dimostrare una proprietà che, anche se mai enunciata espressamente, viene utilizzata dagli autori dell'articolo in alcune dimostrazioni.

Teorema 4.3.1 *Se ρ è un osservabile e $L \subseteq$ Goals, allora $\rho(D|_L) = \rho(D)|_L$.*

Dimostrazione. Sia dato $\boldsymbol{G} \in$ Goals. Per definizione di osservabile esiste ρ^* : PuWFS $\to$ PuWFS tale che:

- $\boldsymbol{G} \in L$.

$$\begin{array}{lcl} \rho(D|_L)(\boldsymbol{G}) & = & [\text{per def. di osservabile}] \\ \rho^*(D|_L\,(\boldsymbol{G})) & = & [\text{per def. di restrizione}] \\ \rho^*(D(\boldsymbol{G})) & = & [\text{per def. di osservabile}] \\ \rho(D)(\boldsymbol{G}) & = & [\text{per def. di restrizione}] \\ \rho(D)|_L\,(\boldsymbol{G}) & & \end{array}$$

- $\boldsymbol{G} \notin L$.

$$\begin{array}{lcl} \rho(D|_L)(\boldsymbol{G}) & = & \\ \rho(D|_L\,(\boldsymbol{G})) & = & [\text{per def. di restrizione}] \\ \rho(\emptyset) & = & [\text{per def. di osservabile}] \\ \emptyset & = & [\text{per def. di restrizione}] \\ \rho(D)|_L\,(\boldsymbol{G}) & & \end{array}$$

Dall'estensionalità della definizione di funzione segue la tesi. ■

Un teorema analogo vale per $\gamma^* \circ \alpha^*$. D'ora in poi abbandoneremo quasi completamente l'uso delle inserzioni di Galois a favore degli operatori di chiusura, dato che trasporre i risultati da un ambito all'altro è solo questione di notazione. Useremo ancora le inserzioni di Galois essenzialmente per affiancare la definizione degli osservabili tramite uco con un'altra presentaziione, spesso più intuitiva.

Si noti, infine, che ogni osservabile ρ induce una *equivalenza osservazionale* $\approx_\rho$, definita per ogni coppia di programmi P_1 e P_2 come:

$$P_1 \approx_\rho P_2 \iff \forall \boldsymbol{G} \in \text{Goals},\ \rho(\mathcal{B}[\![\boldsymbol{G} \in P_1]\!]) = \rho(\mathcal{B}[\![\boldsymbol{G} \in P_2]\!]) \tag{4.3.5}$$

4.3.2 La semantica astratta

Una volta che abbiamo ottenuto un osservabile ρ, vorremmo derivare sistematicamente la corrispondente semantica astratta. L'idea è quella di definire la versione astratta ottimale degli operatori semantici di base e controllare sotto quali condizioni ciò ci consente di ottenere una semantica astratta ottimale e, possibilmente, precisa. Quindi, $\forall X, X_1, X_2 \in \rho(\mathbb{D})$, $\{X_i\} \subseteq \rho(\mathbb{D})$, poniamo:

$$\tilde{\sum} \{X_i\}_{i\in I} = \rho\Big(\sum\{X_i\}_{i\in I}\Big) \tag{4.3.6}$$

$$A \mathbin{\tilde{\cdot}} X = \rho(A \cdot X) \tag{4.3.7}$$

$$X_1 \mathbin{\tilde{\times}} X_2 = \rho(X_1 \times X_1) \tag{4.3.8}$$

$$X_1 \mathbin{\tilde{\bowtie}} X_2 = \rho(X_1 \bowtie X_2) \tag{4.3.9}$$

A questo punto si definisce la corrispondente semantica astratta semplicemente sostituendo agli operatori semantici concreti quelli astratti in tutte le definizioni che li utilizzano. Si ottiene in sostanza:

Unfolding

$$\tilde{\mu}(X) \;=\; \tilde{\sum} \{(A \mathbin{\tilde{\cdot}} X|_{\mathbb{I}}) \mathbin{\tilde{\times}} \rho(Id)\}_{A\in\text{Atoms}} \tag{4.3.10}$$

$$\tilde{\nu}(X) \;=\; \tilde{\sum} \{\mathcal{G}_\rho[\![\boldsymbol{G}]\!]_{X|_{\mathbb{I}}}\}_{\boldsymbol{G}\in\text{Goals}} \tag{4.3.11}$$

Semantica denotazionale

$$
\begin{aligned}
\mathcal{Q}_\rho[\![\boldsymbol{G} \texttt{ in } P]\!] &= \mathcal{G}_\rho[\![\boldsymbol{G}]\!]_{\operatorname{lfp}\mathcal{P}_\rho[\![P]\!]} && (4.3.12)\\
\mathcal{G}_\rho[\![\Box]\!]_X &= \rho(Id|_{\Box}) && (4.3.13)\\
\mathcal{G}_\rho[\![A, \boldsymbol{G}]\!]_X &= \mathcal{A}_\rho[\![a]\!]_X \mathbin{\tilde{\times}} \mathcal{G}_\rho[\![\boldsymbol{G}]\!]_X &&\\
\mathcal{A}_\rho[\![A]\!]_X &= A \mathbin{\tilde{\cdot}} X && (4.3.14)\\
\mathcal{P}_\rho[\![\emptyset]\!] &= \rho(Id)|_{\mathbb{I}} && (4.3.15)\\
\mathcal{P}_\rho[\![\{c\} \cup P]\!]_X &= \left[\mathcal{C}_\rho[\![c]\!]_X \mathbin{\tilde{+}} \mathcal{P}_\rho[\![P]\!]_X\right]_{\equiv_\rho} &&\\
\mathcal{C}_\rho[\![p(\boldsymbol{t} \leftarrow \boldsymbol{B}]\!]_X &= \left[\rho(\operatorname{tree}(p(\boldsymbol{t}) \leftarrow \boldsymbol{B})) \bowtie \mathcal{G}_\rho[\![\boldsymbol{B}]\!]_X\right]_{\equiv_\rho} &&\\
\mathcal{F}_\rho[\![P]\!] &= \operatorname{lfp}\mathcal{P}_\rho[\![P]\!] = \mathcal{P}_\rho[\![P]\!] \uparrow \omega && (4.3.16)
\end{aligned}
$$

Semantica operazionale

$$
\frac{X \in \rho(\mathbb{D}), \qquad X \neq X \bowtie \tilde{\mu}(\rho(\operatorname{tree}(P)))}{X \xmapsto{P}_\rho X \bowtie \tilde{\mu}(\rho(\operatorname{tree}(P)))} \tag{4.3.17}
$$

$$
\begin{aligned}
\mathcal{B}_\rho[\![\boldsymbol{G} \texttt{ in } P]\!] &= \tilde{\sum} \{X \mid \rho(Id|_{\boldsymbol{G}}) \xmapsto{P}{}^*_\rho X\} && (4.3.18)\\
\mathcal{O}_\rho[\![P]\!] &= \left[\tilde{\sum} \{\mathcal{B}[\![p(\boldsymbol{x}) \texttt{ in } P]\!]\}_{p(\boldsymbol{x}) \in \text{Goals}}\right]_{\equiv_\rho} && (4.3.19)
\end{aligned}
$$

Adesso vediamo come alcune condizioni sulla precisione degli operatori semantici astratti portino ad una serie di utili risultati.

4.3.3 Osservabili perfetti

La prima classe di osservabili che consideriamo è quella dei *perfetti*, che hanno semantica operazionale e denotazionale astratta precisa. Come conseguenza si ha che valgono per i perfetti tutte le proprietà di composizionalità, minimalità e correttezza viste nel caso concreto. Un tipico esempio ne è l'osservabile dei risulanti presentato in [GLM95].

Definizione 4.3.2 (Osservabile perfetto) *Un osservabile ρ si dice* perfetto *se valgono le seguenti proprietà. Per ogni $D, D_1, D_2 \in \mathbb{D}$:*

$$
\begin{aligned}
\rho(A \cdot D) &= \rho(A \cdot \rho(D)) && (4.3.20)\\
\rho(D_1 \times D_2) &= \rho(\rho(D_1) \times \rho(D_2)) && (4.3.21)\\
\rho(D_1 \bowtie D_2) &= \rho(\rho(D_1) \bowtie \rho(D_2)) && (4.3.22)
\end{aligned}
$$

Si noti che per ogni operatore di chiusura vale una proprietà analoga a queste per il lub del reticolo concreto, ovvero, per ogni $\{D_i\} \subseteq \mathbb{D}$:

$$\rho\Big(\sum_{i \in I} D_i\Big) = \rho\Big(\sum_{i \in I} \rho(D_i)\Big) \tag{4.3.23}$$

Per questi osservabili si possono ridimostrare tutti i teoremi i già visti nel caso della semantica concreta. Chiaramente l'operatore di composizione OR tra due denotazioni diventa $\tilde{\uplus}$, definito come $X_1 \tilde{\uplus} X_2 = (X_1 \tilde{+} X_2)^*_\rho$ per ogni $X_1, X_2 \in \rho(\mathbb{I})$, dove $(X)^*_\rho$ è la più piccola soluzione dell'equazione

$$(X)^*_\rho = \big[\rho(Id|_{\mathbb{I}}) \tilde{+} ((X)^*_\rho \bowtie \ \tilde{\mu}(X))\big]_{\equiv_\rho} \tag{4.3.24}$$

Inoltre si hanno anche i seguenti risultati di precisione:

Teorema 4.3.3 *Sia ρ un osservabile perfetto e P un programma. Allora per ogni $D \in \mathbb{D}$, esiste $D' \in \mathbb{D}$ tale che $X' = \rho(D')$ e $D \xmapsto{P}{}^{*} D' \iff \rho(D) \xmapsto{P}{}^{*}_{\rho} X'$.*

Corollario 4.3.4 *Sia ρ un osservabile perfetto, $\boldsymbol{G}$ un goal e P un programma. Allora:*

1. $\rho(\mathcal{B}[\![\boldsymbol{G} \ \ \boldsymbol{in} \ \ P]\!]) = \mathcal{B}_\rho[\![\boldsymbol{G} \ \ \boldsymbol{in} \ \ P]\!]$
2. $\rho(\mathcal{O}[\![P]\!]) = \mathcal{O}_\rho[\![P]\!]$

Teorema 4.3.5 *Sia ρ un osservabile perfetto, c una clausola, A un atomo, $\boldsymbol{G}$ un goal e P un programma. Allora:*

1. *ρ è preciso rispetto agli operatori $\mathcal{A}[\![A]\!]$, $\mathcal{G}[\![\boldsymbol{G}]\!]$, $\mathcal{C}[\![c]\!]$ e $\mathcal{P}[\![P]\!]$.*
2. *$\mathcal{P}_\rho[\![P]\!]$ è continuo,*
3. *ρ è preciso rispetto alla semantica denotazionale delle query $\mathcal{Q}[\![\boldsymbol{G} \ \ \boldsymbol{in} \ \ P]\!]$ e alla denotazione bottom-up $\mathcal{F}[\![P]\!]$.*

Ci sono molti casi, però, di osservabili per cui l'operatore $\bowtie$ non è preciso. In questa caso l'idea è di definire la semantica a partire da astrazioni di livello più alto rispetto agli operatori semantici di base.

4.3.4 Osservabili denotazionali

Gli osservabili *denotazionali* si ottengono dai perfetti rilassando la proprietà (4.3.22) e richiedendo la precisione solo sul secondo argomento.

Definizione 4.3.6 (Osservabile denotazionale) *Un osservabile ρ si dice* denotazionale *se valgono le seguenti proprietà:*

$$\begin{aligned} \rho(A \cdot D) &= \rho(A \cdot \rho(D)) \\ \rho(D_1 \times D_2) &= \rho(\rho(D_1) \times \rho(D_2)) \\ \rho(D_1 \bowtie D_2) &= \rho(D_1 \bowtie \rho(D_2)) \end{aligned} \tag{4.3.25}$$

Esempi di osservabili denotazionali sono le *risposte calcolate* [FLMP89], i *call patterns* [GM92], le *istanze (ground) di risposte calcolate*, ecc ...

Si può dimostrare che sotto queste condizioni, semplicemente rimpiazzando $\mathcal{C}_\rho$ con $\tilde{\mathcal{C}}$, versione ottimale di $\mathcal{C}$, ovvero

$$\tilde{\mathcal{C}}[\![c]\!] = \rho \circ \mathcal{C}[\![c]\!] \tag{4.3.26}$$

vale ancora il teorema 4.3.5 e quindi la precisione della semantica denotazionale astratta. Per quanto riguarda la semantica operazionale, si hanno invece i seguenti risultati:

Teorema 4.3.7 *Sia ρ un osservabile denotazionale, A un atomo, $\boldsymbol{G}_1$ e $\boldsymbol{G}_2$ goals e P un programma. Allora:*

1. $A \,\tilde{\cdot}\, \mathcal{O}_\rho[\![P]\!] \sqsubseteq \mathcal{B}_\rho[\![A \;\; \boldsymbol{in} \;\; P]\!]$

2. $\mathcal{B}_\rho[\![\boldsymbol{G}_1 \;\; \boldsymbol{in} \;\; P]\!] \,\tilde{\times}\, \mathcal{B}_\rho[\![\boldsymbol{G}_2 \;\; \boldsymbol{in} \;\; P]\!] \sqsubseteq \mathcal{B}_\rho[\![(\boldsymbol{G}_1, \boldsymbol{G}_2) \;\; \boldsymbol{in} \;\; P]\!]$

Teorema 4.3.8 *Sia ρ un osservabile denotazionale. Allora:*

1. $\mathcal{F}_\rho[\![P]\!] \sqsubseteq \mathcal{O}_\rho[\![P]\!]$

2. $\mathcal{Q}_\rho[\![\boldsymbol{G} \;\; \boldsymbol{in} \;\; P]\!] \sqsubseteq \mathcal{B}_\rho[\![\boldsymbol{G} \;\; \boldsymbol{in} \;\; P]\!]$

Capitolo 5

Proprietà ed estensioni del framework semantico

In questo capitolo ci occuperemo di estendere il framework semantico che abbiamo appena presentato. Inizieremo dando un'occhiata più da vicino alla varianza tra derivazioni e collezioni e al modo in cui è legata all'operatore ∂. Continueremo esaminando le proprietà di cui gode una classe particolari di osservabili, quelli per cui l'operatore di chiusura associato è additivo. Vedremo, in particolare, una caratterizzazione di tali osservabili in funzione di un opportuno predicato binario su $\mathbb{D}$. Successivamente introdurremo la classe degli *osservabili operazionali*, che sono per molti aspetti "complementari" ai denotazionali, sia per le proprietà di cui godono che per il modo in cui sono definiti.

5.1 Collezioni e varianza

Abbiamo definito precedentemente la relazione di varianza tra derivazioni e, di conseguenza, anche tra collezioni. Prima di andare avanti è bene riprendere brevemente il discorso per far vedere come in realtà di varianti di una determinata collezione o derivazione c'è n'è solo una, una volta fissato un goal iniziale.

Lemma 5.1.1 *Sia $d \in Derivs$ e σ una rinominazione. Allora $\partial_\sigma(d) \equiv d$.*

Dimostrazione. Sia $d' \overset{\text{der}}{\approx} d$ tale che $\mathrm{var}(\mathrm{first}(d')\sigma) \cap \mathrm{var}(\mathrm{clauses}(d')) = \emptyset$. Allora $\partial_\sigma(d)$ è una derivazione che ha le stesse clausole di d' e come goal $\mathrm{first}(d)\sigma \equiv \mathrm{first}(d)$. D'altronde $\mathrm{clauses}(d') \equiv \mathrm{clauses}(d)$ da cui la tesi. ∎

Lemma 5.1.2 *Siano $d, d' \in Derivs$ tali che $\mathrm{first}(d) = \boldsymbol{G}$ e $\mathrm{first}(d') = \boldsymbol{G}\sigma$ per un opportuno renaming σ. Allora $d \equiv d'$ sse $d' \overset{\text{der}}{\approx} \partial_\sigma(d)$.*

Dimostrazione. Dire $d \equiv d'$ equivale ad affermare che $\mathrm{first}(d) \equiv \mathrm{first}(d')$ e $\mathrm{clauses}(d) \equiv \mathrm{clauses}(d')$. Se fisso $\mathrm{first}(d')$, tutte le possibili d' sono uguali a meno di varianza delle clausole, ovvero coincidono se prese modulo $\overset{\text{der}}{\approx}$. poiché per il lemma 5.1.1 $\partial_\sigma(d) \equiv d$ e ha primo goal uguale a $\boldsymbol{G}\sigma$, segue banalmente la tesi. ∎

Teorema 5.1.3 *L'operatore ∂_σ, con σ renaming, è un isomorfismo tra i sottoinsiemi di $Derivs_{/\overset{\text{der}}{\approx}}$ formati da tutte le derivazioni con primo goal uguale a $\boldsymbol{G}$ e a $\boldsymbol{G}\sigma$, qualunque sia $\boldsymbol{G}$.*

Dimostrazione. Dal lemma 5.1.2 si ha che ∂_σ è una applicazione bigettiva tra gli insiemi da noi considerati, con inversa pari a $\partial_{\sigma^{-1}}$. D'altronde è anche monotona rispetto alla relazione di prefisso tra derivazioni, come si ricava direttamente dalla definizione, da cui la tesi. ∎

Teorema 5.1.4 *Siano $S, S' \in PuWFS$. Allora $S \equiv S'$ sse $S' = \{\partial_\sigma(d) \mid d \in S\}$ e σ è un renaming tale che $S \in WFS_{\boldsymbol{G}}$ e $S' \in WFS_{\boldsymbol{G}\sigma}$.*

Dimostrazione. Prima di tutto si noti che se $S' = \{\partial_\sigma(d) \mid d \in S\}$, $S' \equiv S$ per il lemma 5.1.1. D'altronde sia $S' \equiv S$, con $S \in \mathrm{WFS}_{\boldsymbol{G}}$, $S' \in \mathrm{WFS}_{\boldsymbol{G}\sigma}$ e σ renaming. Allora per ogni $d \in S$ esiste $d' \in S'$ tale che $d \equiv d'$. Dal lemma 5.1.2 si ha che $d' = \partial_\sigma(d)$, per cui $S' \supseteq \{\partial_\sigma(d) \mid d \in S\}$. Se valesse la diseguaglianza stretta, esisterebbe $d'' \in S'$ tale che $d'' \notin \{\partial_\sigma(d) \mid d \in S\}$. Allora $\partial_{\sigma^{-1}} \notin S$ poiché ∂_σ è bigettiva, per cui non sarebbe vero che $S \equiv S'$. Dunque non rimane altra possibilità che $S' = \{\partial_\sigma(d) \mid d \in S\}$. ∎

Come per il teorema 5.1.3 si verifica che ∂_σ, visto come operatore su insiemi di derivazioni, è un isomorfismo tra $\mathrm{WFS}_{\boldsymbol{G}}$ e $\mathrm{WFS}_{\boldsymbol{G}\sigma}$.

5.2 Osservabili additivi

Diremo *additivi* gli osservabili ρ per cui la funzione ρ è additiva. Ne vedremo, adesso, la caratterizzazione in termini di una opportuna relazione su $\mathrm{Derivs}_{/\overset{der}{\approx}}$ e l'impatto sulle dimostrazioni di precisione rispetto ai vari operatori semantici. Cominciamo con quello che è il risultato fondamentale su questi osservabili:

Teorema 5.2.1 *Sia ρ un osservabile. Allora ρ è additivo sse $\exists \propto$ relazione binaria su $Derivs_{/\overset{der}{\approx}}$ t.c. per ogni $\boldsymbol{G} \in Goals$, $S \in WFS_{\boldsymbol{G}}$*

$$\rho^*(S) = \{d' \in Derivs_{/\overset{der}{\approx}} \mid \exists d \in S \ t.c.\ d' \propto d\}$$

Dimostrazione. Sappiamo già che ρ è additivo rispetto a somme di elementi di $\mathrm{WFS}_{\boldsymbol{G}}$ con $\boldsymbol{G}$ diversi tra loro. Basta allora limitarsi a dimostrare l'additività dei ρ^* per avere automaticamente anche quella di ρ. Esaminiamo le due implicazioni separatamente.

$\Longleftarrow)$ Sia $\{S_i\}_{i\in I} \subseteq \mathrm{WFS}_{\boldsymbol{G}}$, abbiamo

$$\begin{aligned}
&\rho^*(\textstyle\sum\{S_i\}_{i\in I}) && = \quad [\,\text{ipotesi def. di } \rho^*\,] \\
&\{d' \mid \exists d \in \textstyle\sum\{S_i\}_{i\in I} \text{ t.c. } d' \propto d\} && = \quad [\,\text{teoria degli insiemi}\,] \\
&\textstyle\sum\{d' \mid \exists d \in S_i \text{ t.c. } d' \propto d\}_{i\in I} && = \quad [\,\text{sempre per ipotesi def. di } \rho^*\,] \\
&\textstyle\sum\{\rho^*(S_i)\}_{i\in I}
\end{aligned}$$

tenendo conto che l'operatore $\sum$ altro non è che il lub sul reticolo delle collezioni.

$\Longrightarrow)$ Se ρ è additivo poniamo

$$d' \propto d \Longleftrightarrow d' \in \rho^*(\mathrm{WFS}\{d\})$$

dove, se $S \subseteq \mathbb{D}$, con $\mathrm{WFS}(S)$ indichiamo il più piccolo insieme ben formato che contiene S, ottenuto aggiungendo, per ogni $d \in S$, tutti i prefissi di d. Si ottiene allora, per ogni $S \in \mathrm{WFS}_{\boldsymbol{G}}$:

$$\{d' \mid \exists d \in S \text{ t.c. } d' \propto d\} \qquad = \quad [\,\text{def. di} \propto]$$

$$
\begin{aligned}
& \{d' \mid \exists d \in S \text{ t.c. } d' \in \rho^*(\mathrm{WFS}\{d\})\} && = && [\text{teoria degli insiemi}] \\
& \sum\{\rho^*(\mathrm{WFS}\{d\})\}_{d\in S} && = && [\text{ipotesi di additività}] \\
\rho^*(S) &
\end{aligned}
$$

come volevasi dimostrare. ■

In genere a noi interessa, data una relazione binaria $\propto$ su $\mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}}$, stabilire se la funzione ρ, definita come nel teorema 5.2.1, sia effettivamente un osservabile o meno. A questa domanda risponde il seguente:

Teorema 5.2.2 *Condizione necessaria e sufficiente perché* $\rho^*(S) = \{d \in Derivs_{/\overset{\mathrm{der}}{\approx}} \mid \exists d' \in S \text{ t.c. } d \propto d'\}$ *definisca un osservabile è che* $\propto$ *goda di queste proprietà:*

1. $d \propto d' \Rightarrow \mathrm{first}(d) = \mathrm{first}(d')$,
2. $d \propto d' \Rightarrow \forall d_1 \preceq d \ \exists d'_1 \preceq d' \text{ t.c. } d_1 \propto d'_1$,
3. $\forall d \ \exists d' \preceq d \text{ t.c. } d \propto d'$,
4. $d \propto d' \wedge d' \propto d'' \Rightarrow \exists d''' \preceq d'' \text{ t.c. } d \propto d'''$,
5. $d \propto d' \wedge d \equiv d_1 \wedge d' \equiv d'_1 \wedge \mathrm{first}(d_1) = \mathrm{first}(d'_1) \Rightarrow \exists d''_1 \preceq d'_1 \text{ t.c. } d_1 \propto d''_1$

Dimostrazione. Dimostriamo separatamente le due implicazioni.

$\Longleftarrow$) La condizione 1 ci assicura prima di tutto che se $S \in \mathrm{WFS}_{\boldsymbol{G}}$, tutti gli elementi di $\rho^*(S)$ sono derivazioni con goal iniziale pari a $\boldsymbol{G}$.

Sia allora $d \in \rho^*(S)$ ed $S \in \mathrm{WFS}_{\boldsymbol{G}}$. Dobbiamo intanto dimostrare che se d_1 è prefisso di d, $d_1 \in \rho^*(S)$, ovvero che $\rho^*(S)$ è ben formato. Ma $d \in \rho^*(S)$ vuol dire che esiste $d' \in S$ tale che $d \propto d'$. Per la condizione 2, esiste allora $d'_1 \preceq d'$ t.c. $d_1 \propto d'_1$ e poiché S è ben formato, $d'_1 \in S$. Segue banalmente $d'_1 \in \rho^*(S)$.

A questo punto va dimostrato che ρ^* è un uco su $\mathrm{WFS}_{\boldsymbol{G}}$ per ogni $\boldsymbol{G}$ e che $\rho^*(\emptyset) = \emptyset$. Quest'ultima e la monotonia di ρ^* sono dirette conseguenze della sua definizione in funzione di $\propto$, per cui rimane da verificare solo estensionalità e idempotenza.

Dalla condizione 3 si ricava che se $d \in S$, esiste un suo prefisso d' per cui $d \propto d'$. poiché S è ben formato, $d' \in S$ da cui $d \in \rho^*(S)$ come desiderato.

Per dimostrare l'idempotenza, dato che abbiamo già l'estensionalità, basta verificare che $\rho^*(\rho^*(S)) \subseteq \rho^*(S)$, come segue:

$$
\begin{array}{lcl}
d \in \rho^*(\rho^*(S)) & \Leftrightarrow & [\,\text{def. di } \rho^*\,] \\
\exists d' \in \rho^*(S) \text{ t.c. } d \propto d' & \Leftrightarrow & [\,\text{def. di } \rho^*\,] \\
\exists d' \in \mathrm{Derivs}_{/\overset{\text{der}}{\approx}}, d'' \in S \text{ t.c.} & & \\
\quad d \propto d' \wedge d' \propto d'' & \Rightarrow & [\,\text{proprietà } 4\,] \\
\exists d'' \in S, d''' \preceq d'' \text{ t.c. } d \propto d''' & \Leftrightarrow & [\,\text{ancora def. di } \rho^*\,] \\
d \in \rho^*(S) & &
\end{array}
$$

Rimane ora da verificare che $\equiv$ è una congruenza per ρ^*. Sia $S \equiv S_1$ e $d \in \rho^*(S)$, ovvero $\exists d' \in S$ t.c. $d \propto d'$. D'altronde esiste $d'_1 \equiv d'$ in S e, per la condizione 5, dato $d_1 \equiv d$ con $\mathrm{first}(d_1) = \mathrm{first}(d'_1)$, esiste $d''_1 \preceq d'_1$, quindi elemento di S, per cui $d_1 \propto d''_1$, ovvero $d_1 \in \rho^*(S_1)$.

Dunque tutte le condizioni a cui deve sottostare ρ^* perché definisca effettivamente un osservabile sono dimostrare. Dal teorema 5.2.1 si ha come conseguenza che ρ è pure additivo.

$\Longrightarrow$) Si dimostra per assurdo, supponendo che almeno una delle condizioni di cui sopra sia non verificata e mostrando come, in tal caso, la funzione ρ da noi definita non sia un osservabile.

Cominciamo supponendo che la condizione 1 sia falsa, ovvero che esistano d e d' con $d \propto d'$ e $\mathrm{first}(d) \neq \mathrm{first}(d')$. Allora $d \in \rho^*(\mathrm{WFS}\{d'\})$ per cui ρ^* non può essere ristretta ad una funzione da $\mathrm{WFS}_{\boldsymbol{G}}$ a $\mathrm{WFS}_{\boldsymbol{G}}$ per $\boldsymbol{G} = \mathrm{first}(d')$.

Nel caso sia la condizione 2 a non essere vera, esistono d, d' e $d_1 \preceq d$ tali che $d \propto d'$ ma $d_1 \not\propto d'_1$ per tutti i $d'_1 \preceq d'$. Allora è evidente che $\rho^*(\mathrm{WFS}\{d\})$ non è ben formato perché d vi appartiene ma d_1 no.

Quando è la condizione 3 ad essere falsa, esiste una derivazione d tale che $d \not\propto d'$ per ogni $d' \preceq d$. Come conseguenza diretta, $d \notin \rho^*(\mathrm{WFS}\{d\})$ e quindi ρ^* non è estensivo.

Se poi $\propto$ non gode della proprietà 4, esistono $d, d', d'' \in S \in \mathrm{WFS}_{\boldsymbol{G}}$ tali che $d \propto d'$ e $d' \propto d''$ ma $d \not\propto d'''$ per tutti i $d''' \preceq d''$. Consideriamo allora $S = \rho^*(\mathrm{WFS}\{d''\})$ che chiaramente contiene la derivazione d' ma non d. Però $d \in \rho^*(S)$ per cui ρ^* non è idempotente.

Infine, se la condizione 5 non vale, esistono d, d', d_1, d'_1 tali che $d \equiv d'$, $d_1 \equiv d'_1$, $d \propto d'$, $\mathrm{first}(d) = \mathrm{first}(d')$ ma $d_1 \not\propto d''_1$ per tutti i $d''_1 \preceq d'_1$. Sia allora $S = \mathrm{WFS}\{d'\}$ e $S_1 = \mathrm{WFS}\{d'_1\}$. Evidentemente $d \in \rho^*(S)$ ma $d_1 \notin \rho^*(S_1)$. poiché per il lemma 5.1.2 esiste un'unica variante di una derivazione una volta fissato il goal iniziale, d_1 è l'unica variante di d che ha come primo goal proprio $\boldsymbol{G} = \mathrm{first}(d_1)$. poiché $\rho^*(S_1) \in \mathrm{WFS}_{\boldsymbol{G}}$, non esiste variante di d diversa da d_1 che possa appartenere a $\rho^*(S_1)$ e quindi $\rho^*(S) \not\equiv \rho^*(S_1)$.

In sostanza, qualunque sia la condizione su $\propto$ che non è verificata, ρ^* non definisce un osservabile. ∎

Le condizioni che ci consentono di affermare quando $\propto$ è adatta alla definizione di un osservabile sono, come abbiamo visto, piuttosto complesse. È possibile però restringerci sempre a relazioni più semplici, come risulta dal seguente

Teorema 5.2.3 *Condizione necessaria e sufficiente perché la ρ sia un osservabile additivo è che esista una relazione $\propto$ su $Derivs_{/\overset{der}{\approx}}$ tale che $\rho^*(S) = \{d \in Derivs_{/\overset{der}{\approx}} \mid \exists d' \in S$ t.c. $d \propto d'\}$ e che soddisfi le seguenti proprietà:*

1. $d \propto d' \Rightarrow \mathrm{first}(d) = \mathrm{first}(d')$,
2. $d \propto d' \Rightarrow \forall d_1 \preceq d\ \exists d'_1 \preceq d'$ *t.c.* $d_1 \propto d'_1$,
3. $\propto$ *è riflessiva,*
4. $\propto$ *è transitiva,*
5. $d \propto d' \Rightarrow \partial_\sigma(d) \propto \partial_\sigma(d')$ *per ogni renaming* σ.

Dimostrazione. Dimostriamo separatamente le due implicazioni.

$\Longleftarrow$) Questa parte è banale considerando che le proprietà 1-5 sono uguali o comunque implicano le omonime del teorema 5.2.2.

$\Longrightarrow$) Come fatto nel teorema 5.2.1, se ρ è additivo prendiamo $d \propto d'$ sse $d \in \rho^*(\mathrm{WFS}\{d'\})$. Sappiamo già che $\rho^*(S)$ si può scrivere nella forma indicata dal teorema, per cui ci basta dimostrare che $\propto$ gode delle proprietà indicate. La 1 e la 2 sono necessarie in virtù del teorema 5.2.2, per le altre abbiamo:

3) $d' \in \rho^*(\mathrm{WFS}\{d'\})$ per cui $\propto$ è riflessiva.

4) se $d \propto d' \propto d''$ vuol dire che $d \in \rho^*(\mathrm{WFS}\{d'\})$ e $d' \in \rho^*(\mathrm{WFS}\{d''\})$. Ma allora $\mathrm{WFS}\{d'\} \subseteq \rho^*(\mathrm{WFS}\{d''\})$ e quindi $d \in \rho^*(\rho^*(\mathrm{WFS}\{d''\})) = \rho^*(\mathrm{WFS}\{d''\})$. Dalla definizione di $\propto$ segue $d \propto d''$.

5) Sia $d \propto d'$, $d_1 = \partial_\sigma(d)$ e $d'_1 = \partial_\sigma(d')$. Allora $d \in \rho^*(\mathrm{WFS}\{d'\})$ e poiché $\equiv$ è una congruenza per ρ, abbiamo $d_1 \in \rho^*(\mathrm{WFS}\{d'_1\})$, tenendo conto anche del teorema 5.1.4. Segue $d_1 \propto d'_1$. ∎

Gli osservabili additivi, oltre ad essere interessanti in sè in quanto permettono di rappresentare senza perdita di precisione l'informazione disgiuntiva, hanno dei vantaggi anche per la maggiore semplicità con cui ne è possibile dimostrare la precisione rispetto ai vari operatori semantici. Si ha infatti:

Teorema 5.2.4 *Sia ρ un osservabile additivo e* op : $\mathbb{D}^n \to \mathbb{D}$ *un operatore sulle collezioni, anch'esso additivo. Allora ρ è preciso rispetto a* op *sse lo è anche rispetto a* op *ristretto all'insieme $WFS(Derivs_{/\overset{der}{\approx}}) = \{WFS\{d\} \mid d \in Derivs_{/\overset{der}{\approx}}\}$.*

Dimostrazione. Basta osservare che $\mathrm{WFS}(\mathrm{Derivs}_{/\overset{der}{\approx}})$ è una basa disgiuntiva di $\mathbb{D}$. Infatti, per ogni $D \in \mathbb{D}$, abbiamo $D = \bigcup\{\mathrm{WFS}\{d\} \mid d \in \mathbb{D}\}$. Per il teorema 3.1.4 si ha la tesi. ∎

Supponiamo ora di avere una relazione $\propto$ che soddisfi tutte le condizione del teorema 5.2.3 o 5.2.2 tranne la 2. In questo caso i ρ^* possono restituire come risultato anche insiemi che non sono ben formati. Sia allora $\mathrm{gwf}(S)$, per ogni $S \subseteq \mathrm{Derivs}_{/\overset{der}{\approx}}$, il più grande sottoinsieme ben formato di S (che esiste in quanto gli insiemi ben formati sono chiusi per unione). Si noti che gwf è un operatore di chiusura verso il basso sui sottoinsiemi di $\mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}}$. Si ha il seguente:

Teorema 5.2.5 *Data una relazione $\propto$ su $Derivs_{/\overset{der}{\approx}}$ che soddisfi le condizioni 1,3,4 e 5 del teorema 5.2.2, la funzione*

$$\rho^*(S) = \mathrm{gwf}\{d \in Derivs_{/\overset{der}{\approx}} \mid \exists d' \in S \ t.c. \ d \propto d'\}$$

definisce un osservabile.

Dimostrazione. Il fatto che $\rho^*(\emptyset) = \emptyset$ e la monotonia di ρ^* sono banali.

Indichiamo con $\bar{\rho}^*(S)$ l'insieme $\{d \in \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}} \mid \exists d' \in S \text{ t.c. } d \propto d'\}$. Dalla dimostrazione del teorema 5.2.2 si ha che $\bar{\rho}^*$ è estensivo. Dunque $S \subseteq \bar{\rho}^*(S)$ e poiché S è ben formato, $\mathrm{gwf}(\bar{\rho}^*(S)) \supseteq S$, per cui ρ^* è estensionale.

Ancora, si ha che $\bar{\rho}^*$ è idempotente e, poiché $\rho^*(S) \subseteq \bar{\rho}^*(S)$, anche $\bar{\rho}^*(\rho^*(S)) \subseteq \bar{\rho}^*(S)$ e, quindi, applicando gwf ad ambo i membri, $\rho^*(\rho^*(S)) \subseteq \rho^*(S)$. La diseguaglianza opposta deriva dall'estensionalità sopra dimostrata.

Infine, si ha che $\equiv$ è una congruenza per gwf e per $\bar{\rho}^*$, per cui lo è anche per la loro composizione ρ^*. ∎

Teorema 5.2.6 *Sia ρ un osservabile definito come da teorema 5.2.5. Allora ρ è additivo rispetto a $\cdot$, $\times$ o $\bowtie$ se, per ogni $d_1, d_2, d_1', d_2' \in Derivs_{\overset{der}{\approx}}$ con $d_1 \propto d_1'$ e $d_2 \propto d_2'$, vale la corrispondente proprietà per ∂, $\wedge$ o $::$ tra le seguenti:*

1. *per ogni $\gamma \in Subst$, se $\mathrm{first}(d_1)\gamma$ e $\mathrm{first}(d_1)\,\mathrm{answer}(d_1)$ sono unificabili, allora $\partial_\gamma(d_1) \propto \partial_\gamma(d_1')$,*

2. *$d_1 \wedge d_2 \propto d_1' \wedge d_2'$,*

3. *se $d_1 :: d_2$ esiste, allora o $d_1 :: d_2 \propto d_1'$ oppure $d_1' :: d_2'$ esiste e $d_1 :: d_2 \propto d_1' :: d_2'$.*

Dimostrazione. Esaminiamo i tre casi separatamente, ricordando che per la precisione basta dimostrare solo le diseguaglianza non banali.

operatore $\cdot$) Sia $d \in A \cdot \rho(D)$, voglio dimostrare $d \in \rho(A \cdot D)$, ovvero deve esistere, per ogni prefisso di d', un $d'_1 \in A \cdot D$ che è in relazione con d' tramite $\propto$. Sia allora d' un generico prefisso di d. poiché lavoriamo con insiemi ben formati, avremo che anche $d' \in A \cdot \rho(D)$, quindi $d' = \partial_\gamma(d_1)$ con $\gamma \in \mathrm{Subst}$ e $d_1 \in \rho(D)$. Si può inoltre scegliere d_1 in modo tale che $\mathrm{length}(d_1) = \mathrm{length}(d')$ il che vuol dire, grazie al Lemma A.2.3 di [Meo96], che $\mathrm{first}(d_1)\gamma$ e $\mathrm{first}(d_1)\,\mathrm{answer}(d_1)$ sono unificiabili. Per definizione di ρ esiste $d'_1 \in D$ con $d_1 \propto d'_1$ e, dalle ipotesi del teorema, segue $d' \propto \partial_\gamma(d'_1)$ che è un elemento di $A{\cdot}D$, visto che $\mathrm{first}(d'_1) = \mathrm{first}(d_1)$. Dunque $d \in \rho(A \cdot D)$ come volevasi dimostrare.

operatore $\times$) Sia $d \in \rho(D_1) \times \rho(D_2)$. Come fatto sopra, sia d' un prefisso di d. poiché lavoriamo con insiemi ben formati abbiamo $d' = d_1 \wedge d_2$ ed esistono $d'_1 \in D_1$ e $d'_2 \in D_2$ tali che $d_1 \propto d'_1$ e $d_2 \propto d'_2$. Segue dalla condizione 2 che $d_1 \wedge d_2 \propto d'_1 \wedge d'_2$ da cui $d \in \rho(D_1 \bowtie D_2)$ come desiderato.

operatore $\bowtie$) Sia $d \in \rho(D_1) \bowtie \rho(D_2)$ e $d' \preceq d$. Allora d' è ancora elemento di $\rho(D_1) \bowtie \rho(D_2)$. In particolare, si hanno due possibilità. Può essere $d' \in \rho(D_1)$ per cui banalmente è anche $d' \in \rho(D_1 \bowtie D_2) \sqsupseteq \rho(D_1)$. Oppure può essere $d' = d_1 :: d_2$ con $d_1 \in \rho(D_1)$ e $d_2 \in \rho(D_2)$. In tal caso la dimostrazione procede come per l'operatore $\times$. ■

Osservazione 5.2.7 Talvolta, in particolare per quanto riguada l'operatore $\bowtie$, ci serve dimostrare la precisione solo rispetto ad uno dei due argomenti. In tal caso il teorema di cui sopra rimane essenzialmente valido ma le condizioni diventano, ad esempio nel caso in cui si voglia la precisione sul primo argomento:

2. $d_1 \wedge d_2 \propto d'_1 \wedge d_2$

3. se $d_1 :: d_2$ esiste, allora $d'_1 :: d_2$ esiste e $d_1 :: d_2 \propto d'_1 :: d_2$.

Analogamente nel caso di precisione sul secondo argomento. ■

5.3 Osservabili operazionali

Esaminiamo adesso una nuova classe di osservabili, gli *operazionali.* Essi sono essenzialmente i duali dei denotazionali, sia dal punto di vista della definizione che delle proprietà di cui godono. Mentre per i denotazionali abbiamo visto che la semantica di punto fisso $\mathcal{F}_\rho$ è precisa, per gli operazionali la precisione sarà una caratteristica della $\mathcal{O}_\rho$.

Definizione 5.3.1 (Osservabile operazionale) *Un osservabile ρ si dice* operazionale *sse per ogni $D, D_1, D_2 \in \mathbb{D}$ valgono le seguenti proprietà:*

$$\begin{aligned} \rho(A \cdot D) &= \rho(A \cdot \rho(D)) \\ \rho(D_1 \times D_2) &= \rho(\rho(D_1) \times \rho(D_2)) \\ \rho(D_1 \bowtie D_2) &= \rho(\rho(D_1) \bowtie D_2) \end{aligned} \tag{5.3.1}$$

5.3.1 Esempi di osservabili operazionali

Tipici osservabili operazionali sono quelli che mantengono delle informazioni che possono essere ricostruite top-down ma non bottom-up. Ad esempio, sia:

$$\rho^*(S) = \mathrm{gwf}\{d \mid \exists d' \in S \text{ t.c. } d \propto d'\}$$

secondo lo schema del teorema 5.2.5, con $\propto$ così definito:

$$\begin{aligned} d \propto d' \iff & d = \boldsymbol{G}_0 \to \ldots \to \boldsymbol{G}_m,\ d' = \boldsymbol{G}_0 \to \ldots \to \boldsymbol{G}'_k, \\ & \boldsymbol{G}_0 \leq (p(\boldsymbol{x}), \boldsymbol{G}) \text{ per opportuni } p \in \Pi \text{ e } \boldsymbol{G}, \\ & \boldsymbol{x} \text{ renamed apart da } d \text{ e } d', \\ & \mathrm{result}(d) \equiv \mathrm{result}(d'), \boldsymbol{G} \neq \Box \text{ oppure} \\ & \#\{i \mid \mathrm{first}(\boldsymbol{G}_i) \leq p(\boldsymbol{x})\} = \#\{i \mid \mathrm{first}(\boldsymbol{G}'_i) \leq p(\boldsymbol{x})\} \end{aligned} \tag{5.3.2}$$

dove $\text{first}(\boldsymbol{G}) = A$ se $\boldsymbol{G} = A, \boldsymbol{G}'$. In notazione con le inserzioni di Galois diventa:

$$\mathcal{D}_{\boldsymbol{G}} = \begin{cases} \text{Resultants}_{/\overset{\text{der}}{\approx}} \times \mathbb{N} & \text{se } \boldsymbol{G} \text{ è un atomo} \\ \text{Resultants}_{/\overset{\text{der}}{\approx}} & \text{altrimenti} \end{cases} \tag{5.3.3}$$

$$\tag{5.3.4}$$

$$\begin{aligned} \alpha^*(S) = \big\{ \langle R, n \rangle \mid \exists d = \boldsymbol{G}_0 \to \ldots \to \boldsymbol{G}_n \in S \text{ t.c.} \\ \boldsymbol{G}_0 \leq p(\boldsymbol{x}) \text{ per un opportuno } p \in \Pi, \\ \text{result}(d) = R \text{ e } \#\{ i \mid \text{first}(\boldsymbol{G}_i) \leq p(\boldsymbol{x}) \} = n \big\} \end{aligned} \tag{5.3.5}$$

se $S \in \text{WFS}_A$ con $A \in \mathit{Atom}$, altrimenti:

$$\alpha^*(S) = \big\{ R \mid \exists d \in S \text{ t.c. } \text{result}(d) = R \big\} \tag{5.3.6}$$

che non è altro se non una versione più concreta dell'osservabile ρ_R dei risultanti che conta anche il numero di volte in cui in una derivazione viene "chiamato" lo stesso predicato del goal atomico inziale.

Che si tratti di un osservabile lo si vede abbastanza facilmente, grazie al teorema 5.2.5. L'unica proprietà forse non proprio banale da verificare è la 5. Ma basta tenere conto del teorema A.2.2 che ci assicura che $\partial_\sigma(d)$ è d hanno stesso risultante (modulo varianza, ovviamente), stessa lunghezza e goals tra loro varianti.

Vogliamo dimostrare che questo è un osservabile operazionale ma non perfetto. Per far ciò passiamo ad esaminare le proprietà di precisione degli operatori semantici, limitandoci alle sole diseguaglianze non banali. Utilizziamo allo scopo il teorema 5.2.6 e quanto detto a proposito dell'osservazione 5.2.7, dimostrando le condizioni elencate nell'enunciato.

operatore $\boldsymbol{\partial}$) Se $d \propto d'$ vuol dire che le due derivazioni hanno stesso risultante (modulo varianza) e stessa cardinalità delle occorrenze del predicato iniziale nel caso di $\text{first}(d)$ atomico. Per il teorema A.2.6 si ha che $\text{result}(\partial_\gamma(d)) = \text{result}(\partial_\gamma(d'))$ e $\text{length}(\partial_\gamma(d)) = \text{length}(\partial_\gamma(d'))$. Inoltre, l'operazione ∂ non altera i predicati dei goal all'interno delle derivazioni, per cui anche la cardinalità calcolata dall'astrazione rimane uguale. Segue la tesi.

operatore $\times$) Sia $d_1 \propto d_1'$ e $d_2 \propto d_2'$. Le derivazioni $d_1 \wedge d_2$ e $d_1' \wedge d_2'$ o coincidono con d_1 e d_1', qualora fosse $d_2 = d_2' = \Box$, oppure hanno un goal iniziale non atomico. In tal caso, sempre per A.2.6, segue $\text{result}(d_1 \wedge d_2) = \text{result}(d_1' \wedge d_2')$ da cui $d_1 \wedge d_2 \propto d_1' \wedge d_2'$.

operatore $\bowtie$) Sia $d_1 \propto d_1'$ e d_2 una derivazione tale che $d_1 :: d_2$ esiste. Per il teorema A.2.6 si ha che $d_1' :: d_2$ esiste ed ha lo stesso risultante modulo varianza di $d_1 :: d_2$. Se fosse $\text{first}(d_1) \notin \mathit{Atoms}$ si potrebbe già concludere che $d_1 :: d_2 \propto d_1' :: d_2'$, altrimenti basta solo notare che è evidente come anche la cardinalità dei goal istanza del goal iniziale sia uguale in entrambi i casi.

Dunque si tratta di un osservabile operazionale. Che non sia perfetto si verifica facilmente. Basta prendere $D_1 = \text{WFS}\{p(a) \to t(a)\}$ e $D_2 = \text{WFS}\{t(a) \to \Box\}$. Chiaramente $t(a) \to p(b) \to \Box \in \rho(D_2)$ da cui $d = p(a) \to t(a) \to p(b) \to \Box \in \rho(\rho(D_1) \bowtie \rho(D_2))$ mentre $d \notin \rho(D_1 \bowtie D_2)$.

5.3.2 Proprietà degli osservabili operazionali

È evidente che un osservabile perfetto è operazionale ma che il viceversa in generale non vale. Tuttavia è vero quanto segue:

Teorema 5.3.2 *Se ρ è un osservabile denotazionale ed operazionale, allora è perfetto.*

Dimostrazione. Basta dimostrare che vale la proprietà (4.3.22), come segue:

$$\begin{array}{lll} \rho(\rho(D_1) \bowtie \rho(D_2)) & = & [\rho \text{ è operazionale}] \\ \rho(D_1 \bowtie \rho(D_2)) & = & [\rho \text{ è denotazionale}] \\ \rho(D_1 \bowtie D_2) & & \end{array}$$

che era appunto ciò che volevamo ottenere. ■

Ripercorriamo adesso tutti i teoremi che valgono per gli osservabili perfetti tentando di ridimostrarli nel caso sotto esame degli operazionali.

Lemma 5.3.3 *Sia ρ un osservabile operazionale, $X_1, X_2, X_3 \in \rho(\mathbb{D})$. Allora:*

$$X_1 \;\tilde{\times}\; (X_2 \;\tilde{\times}\; X_3) = (X_1 \;\tilde{\times}\; X_2) \;\tilde{\times}\; X_3$$

Dimostrazione. Dalla precisione dell'operatore $\times$ segue che:

$$\begin{aligned} X_1 \;\tilde{\times}\; (X_2 \;\tilde{\times}\; X_3) &= \rho(X_1 \times (X_2 \times X_3)) \\ (X_1 \;\tilde{\times}\; X_2) \;\tilde{\times}\; X_3 &= \rho((X_1 \times X_2) \times X_3) \end{aligned}$$

Dal Lemma 4.2.8 si ha l'uguaglianza dei secondi membri che completa la prova.

■

Lemma 5.3.4 *Sia ρ un osservabile operazionale. Allora $\tilde{\cdot}$ e $\tilde{\times}$ sono additivi, $\bowtie$ è additivo sul primo argomento.*

Dimostrazione. Dato che $\tilde{\sum}$ non è altro che il lub del nostro reticolo delle astrazioni, bisogna dimostrare che per ogni $\{X_i\}_{i \in I} \subseteq \rho(\mathbb{D})$ e $X \in \rho(\mathbb{D})$,

$$\tilde{\sum} \{X_i\}_{i \in I} \bowtie X = \tilde{\sum} \{X_i \bowtie X\}_{i \in I}$$

La prova procede facilmente usando la definizione di osservabile operazionale:

$$\begin{aligned}
& \tilde{\sum} \{X_i\}_{i \in I} \bowtie X && = \quad [\text{def. di operatori astratti}] \\
& \rho\Big(\rho\Big(\sum\{X_i\}_{i \in I}\Big) \bowtie X\Big) && = \quad [\text{def. di oss. operazionale, prop. (5.3.1)}] \\
& \rho\Big(\sum\{X_i\}_{i \in I} \bowtie X\Big) && = \quad [\text{additività di } \bowtie\text{, teorema 4.2.9}] \\
& \rho\Big(\sum\{X_i \bowtie X\}_{i \in I}\Big) && = \quad [\text{precisione di } \textstyle\sum] \\
& \rho\Big(\sum\{\rho(X_i \bowtie X)\}_{i \in I}\Big) && = \quad [\text{def. degli operatori astratti}] \\
& \tilde{\sum} \{X_i \bowtie X\}_{i \in I}
\end{aligned}$$

Analogamente per gli altri casi. ■

L'operatore astratto $\bowtie$ ci è comunque di scarsa utilità a causa della mancanza di precisione sul secondo argomento. Per sopperire a questa situazione sarà necessario definire la nostra semantica in modo che tutte le concatenazioni vengano effettuate tra derivazioni astratte e derivazioni concrete. A tale scopo definiamo l'operatore $\bar{\bowtie}$: $\rho(\mathbb{D}) \times \mathbb{D} \to \rho(\mathbb{D})$ come segue. Per ogni $X \in \rho(\mathbb{D})$, $D \in \mathbb{D}$:

$$X \mathbin{\bar{\bowtie}} D = \rho(X \bowtie D) \tag{5.3.7}$$

L'operatore $\bar{\bowtie}$ gode di proprietà simili a quelle di $\bowtie$ e $\tilde{\bowtie}$.

Lemma 5.3.5 *Sia ρ un osservabile operazionale, $X \in \rho(\mathbb{D})$, $D_1, D_2 \in \mathbb{D}$ con $D_2 \sqsubseteq D_1$. Allora:*

$$(X \mathbin{\bar{\bowtie}} D_1) \mathbin{\bar{\bowtie}} D_2 = X \mathbin{\bar{\bowtie}} (D_1 \bowtie D_2)$$

Dimostrazione. Dalla definizione di $\bar{\bowtie}$ e per la proprietà (5.3.1) si ha:

$$(X \mathbin{\bar{\bowtie}} D_1) \mathbin{\bar{\bowtie}} D_2 = \rho((X \bowtie D_1) \bowtie D_2)$$
$$X \mathbin{\bar{\bowtie}} (D_1 \bowtie D_2) = \rho(X \bowtie (D_1 \bowtie D_2))$$

e dal Lemma 4.2.8 segue la tesi. ∎

Lemma 5.3.6 *Sia ρ un osservabile operazionale, allora $\bar{\bowtie}$ è additivo, ovvero per ogni $\{X_i\}_{i \in I} \subseteq \rho(\mathbb{D})$, $\{D_i\}_{i \in I} \subseteq \mathbb{D}$, $X \in \rho(\mathbb{D})$ e $D \in \mathbb{D}$, si ha:*

1. $\tilde{\sum} \{X_i\}_{i \in I} \mathbin{\bar{\bowtie}} D = \tilde{\sum} \{X_i \mathbin{\bar{\bowtie}} D\}_{i \in I}$
2. $X \mathbin{\bar{\bowtie}} \sum \{D_i\}_{i \in I} = \tilde{\sum} \{X \mathbin{\bar{\bowtie}} D_i\}_{i \in I}$

Dimostrazione. Simile al lemma 5.3.4. ∎

Analogamente al caso dei denotazionali è possibile, per questa classe di osservabili, ricavare una semantica precisa, che questa volta si rivela essere quella operazionale. Occorre ridefinire la relazione di transizione rispetto al caso ottimale degli osservabili perfetti ponendo:

$$\frac{X \in \rho(\mathbb{D}), \quad X \neq X \mathbin{\bar{\bowtie}} \mu(\text{tree}(P))}{X \xmapsto{P}_{\rho} X \mathbin{\bar{\bowtie}} \mu(\text{tree}(P))} \tag{5.3.8}$$

ovvero, tramite le inserzioni di Galois,

$$\frac{X \in \mathbb{A}, \quad X \neq \alpha\big(\gamma(X) \bowtie \mu(\text{tree}(P))\big)}{X \xmapsto{P}_{\alpha} \alpha\big(\gamma(X) \bowtie \mu(\text{tree}(P))\big)} \tag{5.3.9}$$

Esprimiamo adesso le funzioni semantiche $\mathcal{B}_\rho$ e $\mathcal{O}_\rho$, definite sulla base del sistema di transizione, in modo che utilizzino direttamente l'operatore $\bar{\bowtie}$.

Teorema 5.3.7 *Sia P un programma, $\boldsymbol{G} \in Goals$ e ρ un osservabile operazionale. Allora:*

1. $\mathcal{B}_\rho[\![\boldsymbol{G} \ \ \boldsymbol{in} \ \ P]\!] = \tilde{\sum} \left\{\rho(Id|_{\boldsymbol{G}}) \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))\right\}_{n\geq 0}$

2. $\mathcal{O}[\![P]\!]_\rho = \left[\rho(Id|_{\mathbb{I}}) \mathbin{\bar{\bowtie}} \sum\{\mu_n(\mathrm{tree}(P))\}_{n\geq 0}\right]_\equiv$

Dimostrazione. Procede in maniera analoga a quella del Corollario 6.5.5 in [Meo96]. Utilizziamo la notazione $X \stackrel{P}{\longmapsto}{}^i_\rho X'$ per indicare che da X si passa ad X' in *al più i* passi del sistema di transizione. Cominciamo col verificare che, per definizione di $\stackrel{P}{\longmapsto}{}_\rho$, dato $X \in \rho(\mathbb{D})$,

$$\tilde{\sum} \{X' \mid \bar{X} \stackrel{P}{\longmapsto}{}^1_\rho X'\} = \bar{X} \mathbin{\bar{\bowtie}} \mu(\mathrm{tree}(P)) \tag{1}$$

e pertanto, per ogni $\{X_i\}_{i\in I} \subseteq \rho(\mathbb{D})$

$$\begin{array}{lcl}
\tilde{\sum} \{X' \mid \tilde{\sum} \{X_i\}_{i\in I} \stackrel{P}{\longmapsto}{}^1_\rho X'\} & = & [\text{per } (1)] \\
\tilde{\sum} \{X_i\}_{i\in I} \mathbin{\bar{\bowtie}} \mu(\mathrm{tree}(P)) & = & [\text{Lemma } 5.3.6] \\
\tilde{\sum} \{X_i \mathbin{\bar{\bowtie}} \mu(\mathrm{tree}(P))\}_{i\in I} & = & [\text{per } (1)] \\
\tilde{\sum} \left\{\tilde{\sum} \{X' \mid X_i \stackrel{P}{\longmapsto}{}^1_\rho X'\}\right\}_{i\in I} & &
\end{array} \tag{2}$$

Ora proviamo che $\tilde{\sum} \{X' \mid \bar{X} \stackrel{P}{\longmapsto}{}^n_\rho X'\} = \bar{X} \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))$. Lo facciamo per induzione su n:

$\boldsymbol{n = 0}$) Banale, perché $\tilde{\sum} \{X' \mid \bar{X} \stackrel{P}{\longmapsto}{}^0_\rho X'\} = \bar{X} = \bar{X} \mathbin{\bar{\bowtie}} \mu_0(\mathrm{tree}(P))$

$\boldsymbol{n > 0}$) Si dimostra come segue:

$$\begin{array}{lr}
\tilde{\sum} \{X' \mid \bar{X} \stackrel{P}{\longmapsto}{}^n_\rho X'\} & = \\
\quad [\text{definizione di } \stackrel{P}{\longmapsto}{}^n] & \\
\tilde{\sum} \{X' \mid \exists X'' \text{ t.c. } \bar{X} \stackrel{P}{\longmapsto}{}^{n-1}_\rho X'' \text{ e } X'' \stackrel{P}{\longmapsto}{}^1_\rho X'\} & = \\
\quad [\text{per la teoria degli insiemi}] & \\
\tilde{\sum} \left\{\tilde{\sum} \{X' \mid X'' \stackrel{P}{\longmapsto}{}^1_\rho X'\} \mid \bar{X} \stackrel{P}{\longmapsto}{}^{n-1}_\rho X''\right\} & = \\
\quad [\text{per } (2)] &
\end{array}$$

$$\tilde{\sum} \{X'' \mid \bar{X} \overset{P}{\longmapsto}{}^{n-1} X''\} \bar{\bowtie} \mu(\text{tree}(P)) \quad =$$

[per ipotesi induttiva]

$$(\bar{X} \bar{\bowtie} \mu_{n-1}(\text{tree}(P))) \bar{\bowtie} \mu(\text{tree}(P)) \quad =$$

[Lemma 5.3.5]

$$\bar{X} \bar{\bowtie} \mu_n(\text{tree}(P))$$

Finalmente si arriva a:

$$\begin{array}{lcl}
\mathcal{B}_\rho[\![\boldsymbol{G} \ \texttt{in} \ P]\!] & = & [\,\text{definizione di } \mathcal{B}_\rho\,] \\
\tilde{\sum} \{X \mid \rho(Id|_{\boldsymbol{G}}) \overset{P}{\longmapsto}{}^*_\rho X\} & = & [\,\text{definizione di } \overset{P}{\longmapsto}{}^*_\rho\,] \\
\tilde{\sum} \left\{\tilde{\sum} \{X \mid \rho(Id|_{\boldsymbol{G}}) \overset{P}{\longmapsto}{}^n_\rho X\}\right\}_{n\geq 0} & = & [\,\text{per il precedente risultato}\,] \\
\tilde{\sum} \left\{\rho(Id|_{\boldsymbol{G}}) \bar{\bowtie} \mu_n(\text{tree}(P))\right\}_{n\geq 0} & &
\end{array}$$

che completa la prova del primo punto. Per quanto riguarda la caratterizzazione di $\mathcal{O}_\rho$ si procede invece così:

$$\mathcal{O}_\rho[\![P]\!] \quad =$$

[per def. di $\mathcal{O}$]

$$\left[\tilde{\sum} \{\mathcal{B}_\rho[\![p(\boldsymbol{x}) \ \texttt{in} \ P]\!]\}_{p(\boldsymbol{x})\in \text{Goals}}\right]_\equiv \quad =$$

[per il punto 1]

$$\left[\tilde{\sum} \{\tilde{\sum} \{\rho(Id|_{p(\boldsymbol{x})}) \bar{\bowtie} \mu_n(\text{tree}(P))\}_{n\geq 0}\}_{p(\boldsymbol{x})\in \text{Goals}}\right]_\equiv \quad =$$

[per la teoria degli insiemi]

$$\left[\tilde{\sum} \{\tilde{\sum} \{\rho(Id|_{p(\boldsymbol{x})}) \bar{\bowtie} \mu_n(\text{tree}(P))\}_{p(\boldsymbol{x})\in \text{Goals}}\}_{n\geq 0}\right]_\equiv \quad =$$

[per Lemma 5.3.6, punto 1]

$$\left[\tilde{\sum} \{\rho(Id|_{\mathbb{I}}) \bar{\bowtie} \mu_n(\text{tree}(P))\}_{n\geq 0}\right]_\equiv \quad =$$

[ancora Lemma 5.3.6, punto 2]

$$\left[\rho(Id|_{\mathbb{I}}) \bar{\bowtie} \sum \{\mu_n(\text{tree}(P))\}_{n\geq 0}\right]_\equiv$$

che completa la prova. ■

Lemma 5.3.8 *Sia ρ un osservabile operazionale, $X \in \rho(\mathbb{D})$, $D \in \mathbb{D}$. Allora:*

$$A \mathbin{\tilde{:}} (X \bar{\bowtie} \mu(D)) = (A \mathbin{\tilde{:}} X) \bar{\bowtie} \mu(D)$$

Dimostrazione. È facile verificare la seguente:

$$
\begin{array}{lcl}
A \,\tilde{\cdot}\, (X \mathbin{\bar{\bowtie}} \mu(D)) & = & [\,\text{definizione di } \bar{\bowtie}\,] \\
A \,\tilde{\cdot}\, \rho(X \bowtie \mu(D)) & = & [\,\text{precisione di } \cdot\,] \\
\rho\big(A \cdot (X \bowtie \mu(D))\big) & = & [\,\text{grazie al Lemma 4.2.11}\,] \\
\rho\big((A \cdot X) \bowtie \mu(D)\big) & = & [\,\text{per le def. 5.3.1 e def. di } \bar{\bowtie})\,] \\
(A \,\tilde{\cdot}\, X) \mathbin{\bar{\bowtie}} \mu(D) & &
\end{array}
$$

che ci dà il risultato cercato. ■

Lemma 5.3.9 *Sia ρ un osservabile operazionale, $X_1, X_2 \in \rho(\mathbb{D})$, $D \in \mathbb{D}$. Segue:*

1. $(X_1 \mathbin{\bar{\bowtie}} \mu(D)) \mathbin{\tilde{\times}} \rho(Id|_{\boldsymbol{G}}) \sqsubseteq (X_1 \mathbin{\tilde{\times}} \rho(Id|_{\boldsymbol{G}})) \mathbin{\bar{\bowtie}} \mu(D)$
2. *Se* $X_1 \mathbin{\bar{\bowtie}} \mu(D) = X_1$ *allora* $(X_1 \mathbin{\tilde{\times}} X_2) \mathbin{\bar{\bowtie}} \mu(D) = X_1 \mathbin{\tilde{\times}} (X_2 \mathbin{\bar{\bowtie}} \mu(D))$

Dimostrazione. Distinguiamo i due punti:

1. Si dimostra come per il Lemma 5.3.8.
2. Si osservi, intanto, che poiché ρ è un uco si ha

$$X_1 \mathbin{\bar{\bowtie}} \mu(D) = \rho(X_1 \bowtie \mu(D)) \sqsupseteq X_1 \bowtie \mu(D) \sqsupseteq X_1 \tag{1}$$

   Ma dalle ipotesi, poiché $X_1 = X_1 \mathbin{\bar{\bowtie}} \mu(D)$, segue $X_1 = X_1 \bowtie \mu(D)$. A questo punto si dimostra come per il Lemma 5.3.8. ■

Corollario 5.3.10 *Sia ρ un osservabile operazionale, A un atomo, $\boldsymbol{G}$ un goal, $D \in \mathbb{D}$ e $X_1, X_2 \in \rho(\mathbb{D})$. Per qualsiasi $n \geq 0$ segue:*

1. $A \,\tilde{\cdot}\, (X \mathbin{\bar{\bowtie}} \mu_n(D)) = (A \,\tilde{\cdot}\, X) \mathbin{\bar{\bowtie}} \mu_n(D)$
2. $(X_1 \mathbin{\bar{\bowtie}} \mu_n(D)) \mathbin{\tilde{\times}} \rho(Id|_{\boldsymbol{G}}) \sqsubseteq (X_1 \mathbin{\tilde{\times}} \rho(Id|_{\boldsymbol{G}})) \mathbin{\bar{\bowtie}} \mu_n(D)$
3. *Se* $X_1 \mathbin{\bar{\bowtie}} \mu_n(D) = X_1$ *allora* $(X_1 \mathbin{\tilde{\times}} X_2) \mathbin{\bar{\bowtie}} \mu_n(D) = X_1 \mathbin{\tilde{\times}} (X_2 \mathbin{\bar{\bowtie}} \mu_n(D))$

Dimostrazione. Segue da un semplice ragionamento induttivo sui Lemmi 5.3.8 e 5.3.9. Vediamo in dettaglio solo la dimostrazione del punto 1.

$\boldsymbol{n = 0}$) Ricordiamo che per definizione $\mu_0(D) = \emptyset$, per cui, banalmente, $A \,\tilde{\cdot}\, (X \mathbin{\bar{\bowtie}} \mu_0(D)) = A \,\tilde{\cdot}\, X = (A \,\tilde{\cdot}\, X) \mathbin{\bar{\bowtie}} \mu_0(D)$.

$\boldsymbol{n > 0}$) Come già detto, per induzione su n, otteniamo

$$
\begin{array}{lcl}
A \,\tilde{\cdot}\, (X \mathbin{\bar{\bowtie}} \mu_n(D)) & = & [\text{per def. di } \mu_n] \\
A \,\tilde{\cdot}\, (X \mathbin{\bar{\bowtie}} (\mu_{n-1}(D) \bowtie \mu(D))) & = & [\text{per Lemma 5.3.5}] \\
A \,\tilde{\cdot}\, \big((X \mathbin{\bar{\bowtie}} \mu_{n-1}(D)) \mathbin{\bar{\bowtie}} \mu(D)\big) & = & [\text{per Lemma 5.3.8}] \\
\big(A \,\tilde{\cdot}\, (X \mathbin{\bar{\bowtie}} \mu_{n-1}(D))\big) \mathbin{\bar{\bowtie}} \mu(D) & = & [\text{per ipotesi induttiva}] \\
\big((A \,\tilde{\cdot}\, X) \mathbin{\bar{\bowtie}} \mu_{n-1}(D)\big) \mathbin{\bar{\bowtie}} \mu(D) & = & [\text{per Lemma 5.3.5}] \\
(A \,\tilde{\cdot}\, X) \mathbin{\bar{\bowtie}} \mu_n(D) & &
\end{array}
$$

che completa la prova. ■

Grazie a quest'ultimo corollario siamo in grado di costruire l'astrazione di un albero SLD per un generico goal dagli alberi SLD per i goal atomici puri. Vedremo ora come tutte le proprietà della semantica top-down concreta vengono ereditate dalla nostra semantica astratta.

Teorema 5.3.11 *Sia ρ un osservabile operazionale, A un atomo, $\boldsymbol{G}, \boldsymbol{G}_1, \boldsymbol{G}_2 \in$ Goals e $P \in$ Progs. Allora:*

1. $\mathcal{B}_\rho[\![A \ \textit{\textbf{in}} \ P]\!] = A \,\tilde{\cdot}\, \mathcal{O}_\rho[\![P]\!]$

2. $\mathcal{B}_\rho[\![\boldsymbol{G}_1, \boldsymbol{G}_2 \ \textit{\textbf{in}} \ P]\!] = \mathcal{B}_\rho[\![\boldsymbol{G}_1 \ \textit{\textbf{in}} \ P]\!] \mathbin{\tilde{\times}} \mathcal{B}_\rho[\![\boldsymbol{G}_2 \ \textit{\textbf{in}} \ P]\!]$

Dimostrazione. Passiamo ad esaminare separatamente i due punti.

1. È la parte più semplice del teorema e lo si dimostra facilmente sfruttando i lemmi già enunciati.

$$
\begin{array}{lcl}
\mathcal{B}_\rho[\![A \ \texttt{in} \ P]\!] & = & [\text{per il Teorema 5.3.7}] \\
\tilde{\sum} \{\rho(Id|_A) \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))\}_{n \geq 0} & = & [\text{def. di } \tilde{\cdot} \text{ e } Id] \\
\tilde{\sum} \big\{(A \,\tilde{\cdot}\, \rho(Id|_{\mathbb{I}})) \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))\big\}_{n \geq 0} & = & [\text{punto 1 Corollario 5.3.10}] \\
\tilde{\sum} \big\{A \,\tilde{\cdot}\, \big(\rho(Id|_{\mathbb{I}}) \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))\big)\big\}_{n \geq 0} & = & [\,(\text{additività di } \tilde{\cdot} \text{ e } \bar{\bowtie}]
\end{array}
$$

$$\begin{array}{lcl} A \,\tilde{:}\, \left(\rho(Id|_{\mathbb{I}}) \mathbin{\bar{\bowtie}} \sum\{\mu_n(\mathrm{tree}(P))\}_{n\geq 0}\right) & = & [\,\text{per il Teorema 5.3.7}\,] \\ A \,\tilde{:}\, \mathcal{O}_\rho[\![P]\!] & & \end{array}$$

2. Prima di tutto si noti che per ogni $n \geq 0$,

$$\begin{array}{ll}
\mathcal{B}_\rho[\![\boldsymbol{G}_1 \ \texttt{in} \ P]\!] \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P)) & = \\
\qquad [\,\text{per il Teorema 5.3.7}\,] & \\
\tilde{\sum} \left\{\rho(Id|_{\boldsymbol{G}_1}) \mathbin{\bar{\bowtie}} \mu_k(\mathrm{tree}(P))\right\}_{k\geq 0} \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P)) & = \\
\qquad [\,\text{per il Lemma 5.3.6}\,] & \\
\tilde{\sum} \left\{\left(\rho(Id|_{\boldsymbol{G}_1}) \mathbin{\bar{\bowtie}} \mu_k(\mathrm{tree}(P))\right) \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))\right\}_{k\geq 0} & = \\
\qquad [\,\text{per il Lemma 5.3.5}\,] & \\
\tilde{\sum} \left\{\rho(Id|_{\boldsymbol{G}_1}) \mathbin{\bar{\bowtie}} \mu_k(\mathrm{tree}(P))\right\}_{k\geq 0} & = \\
\qquad [\,\text{per il Teorema 5.3.7}\,] & \\
\mathcal{B}_\rho[\![\boldsymbol{G}_1 \ \texttt{in} \ P]\!] &
\end{array}$$

Inoltre, sempre per il Teorema 5.3.7 e per la definizione di $\bar{\bowtie}$, abbiamo che per ogni $\boldsymbol{G}_1 \in \mathrm{Goals}$,

$$\rho(Id|_{\boldsymbol{G}_1}) \sqsubseteq \mathcal{B}_\rho[\![\boldsymbol{G}_1 \ \texttt{in} \ P]\!] \tag{1}$$

Proviamo adesso le due inclusioni separatamente. Cominciamo con la $\sqsubseteq$.

$$\begin{array}{ll}
\mathcal{B}_\rho[\![(\boldsymbol{G}_1, \boldsymbol{G}_2) \ \texttt{in} \ P]\!] & = \\
\qquad [\,\text{per il Teorema 5.3.7}\,] & \\
\tilde{\sum} \left\{\rho(Id|_{(\boldsymbol{G}_1,\boldsymbol{G}_2)}) \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))\right\}_{n\geq 0} & = \\
\qquad [\,\text{per definizione e precisione di } \tilde{\times}\,] & \\
\tilde{\sum} \left\{\left(\rho(Id|_{\boldsymbol{G}_1}) \mathbin{\tilde{\times}} \rho(Id|_{\boldsymbol{G}_2})\right) \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))\right\}_{n\geq 0} & \sqsubseteq \\
\qquad [\,\text{per (1) e monotonia di } \bar{\bowtie} \text{ e } \tilde{\times}\,] & \\
\tilde{\sum} \left\{\left(\mathcal{B}_\rho[\![\boldsymbol{G}_1 \ \texttt{in} \ P]\!] \mathbin{\tilde{\times}} \rho(Id|_{\boldsymbol{G}_2})\right) \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))\right\}_{n\geq 0} & = \\
\qquad [\,\text{per il punto 3 del Corollario 5.3.10 e per l'osserv. precedente}\,] & \\
\tilde{\sum} \left\{\mathcal{B}_\rho[\![\boldsymbol{G}_1 \ \texttt{in} \ P]\!] \mathbin{\tilde{\times}} \left(\rho(Id|_{\boldsymbol{G}_2}) \mathbin{\bar{\bowtie}} \mu_n(\mathrm{tree}(P))\right)\right\}_{n\geq 0} & = \\
\qquad [\,\text{per additività di } \tilde{\times}\,] &
\end{array}$$

$$\mathcal{B}_\rho[\![\boldsymbol{G}_1 \texttt{ in } P]\!] \,\tilde{\times} \tilde{\sum} \left\{\rho(Id|_{\boldsymbol{G}_2}) \,\bar{\bowtie}\, \mu_n(\text{tree}(P))\right\}_{n\geq 0} =$$

[per il Teorema 5.3.7]

$$\mathcal{B}_\rho[\![\boldsymbol{G}_1 \texttt{ in } P]\!] \,\tilde{\times}\, \mathcal{B}_\rho[\![\boldsymbol{G}_2 \texttt{ in } P]\!]$$

che completa la prova della prima inclusione. Per quanto riguarda la diseguaglianza opposta, abbiamo:

$$\mathcal{B}_\rho[\![\boldsymbol{G}_1 \texttt{ in } P]\!] \,\tilde{\times}\, \mathcal{B}_\rho[\![\boldsymbol{G}_2 \texttt{ in } P]\!] =$$

[ripetendo i passi precedenti]

$$\tilde{\sum} \left\{\left(\mathcal{B}_\rho[\![\boldsymbol{G}_1 \texttt{ in } P]\!] \,\tilde{\times}\, \rho(Id|_{\boldsymbol{G}2})\right) \,\bar{\bowtie}\, \mu_n(\text{tree}(P))\right\}_{n\geq 0} =$$

[per il Teorema 5.3.7]

$$\tilde{\sum} \left\{\left(\tilde{\sum} \{\rho(Id|_{\boldsymbol{G}_1}) \,\bar{\bowtie}\, \mu_k(\text{tree}(P))\}_{k\geq 0} \,\tilde{\times}\, \rho(Id|_{\boldsymbol{G}2})\right) \,\bar{\bowtie}\, \mu_n(\text{tree}(P))\right\}_{n\geq 0} =$$

[per additività di $\tilde{\times}$ e $\bar{\bowtie}$]

$$\tilde{\sum} \left\{\tilde{\sum} \left\{\left((\rho(Id|_{\boldsymbol{G}_1}) \,\bar{\bowtie}\, \mu_k(\text{tree}(P))) \,\tilde{\times}\, \rho(Id|_{\boldsymbol{G}2})\right) \,\bar{\bowtie}\, \mu_n(\text{tree}(P))\right\}_{k\geq 0}\right\}_{n\geq 0} \sqsubseteq$$

[per il punto 2 del Corollario 5.3.10]

$$\tilde{\sum} \left\{\tilde{\sum} \left\{\left((\rho(Id|_{\boldsymbol{G}_1}) \,\tilde{\times}\, \rho(Id|_{\boldsymbol{G}_2})) \,\bar{\bowtie}\, \mu_k(\text{tree}(P))\right) \,\bar{\bowtie}\, \mu_n(\text{tree}(P))\right\}_{k\geq 0}\right\}_{n\geq 0} =$$

[per il Lemma 5.3.5, teoria degli insiemi]

$$\tilde{\sum} \left\{(\rho(Id|_{\boldsymbol{G}_1}) \,\tilde{\times}\, \rho(Id|_{\boldsymbol{G}_2})) \,\bar{\bowtie}\, \mu_n(\text{tree}(P))\right\}_{n\geq 0} =$$

[per definizione e precisione di $\bar{\bowtie}$]

$$\tilde{\sum} \{\rho(Id|_{(\boldsymbol{G}_1,\boldsymbol{G}_2)}) \,\bar{\bowtie}\, \mu_n(\text{tree}(P))\}_{n\geq 0} =$$

[per il teorema 5.3.7]

$$\mathcal{B}_\rho[\![(\boldsymbol{G}_1, \boldsymbol{G}_2) \texttt{ in } P]\!]$$

che completa la prova. ∎

Corollario 5.3.12 *Sia ρ un osservabile operazionale e P_1, P_2 due programmi. Allora:*

$$P_1 \approx_\rho P_2 \Leftrightarrow \mathcal{O}_\rho[\![P_1]\!] = \mathcal{O}_\rho[\![P_2]\!]$$

Dimostrazione. Esaminando separatamente le due implicazioni abbiamo che la $\Rightarrow$ è banale per definizione stessa di $\approx_\rho$ e di $\mathcal{O}_\rho$ mentre la $\Leftarrow$ nasce tenendo conto che il teorema 5.3.11 ci consente di determinare univocamente il valore di $\mathcal{B}_\rho[\![\boldsymbol{G}\ \mathtt{in}\ P]\!]$ a partire da $\mathcal{O}_\rho[\![P]\!]$, come si dimostra facilmente per induzione sulla struttura dei goal. ∎

Teorema 5.3.13 *Sia ρ un osservabile operazionale, P un programma. Allora per ogni $D \in \mathbb{D}$ esiste $D' \in \mathbb{D}$ tale che $X' = \rho(D')$ e $D \stackrel{P}{\longmapsto}^* D'$ sse $\rho(D) \stackrel{P}{\longmapsto}_\rho^* X'$.*

Dimostrazione. Proviamo più in particolare che esiste $k \geq 0$ e $D' \in \mathbb{D}$ tale che $X' = \rho(D')$ e $D \stackrel{P}{\longmapsto}^k D'$ se e solo se esiste $n \geq 0$ tale che $\rho(D) \stackrel{P}{\longmapsto}_\rho^n X'$, dove $D \stackrel{P}{\longmapsto}^k D'$ e $X \stackrel{P}{\longmapsto}_\rho^n X'$ indicano che D' (ed X') si ottengono da D (ed X) in *esattamente* k ed n passi. Lo si fa per induzione su k ed n.

passo base) Immediato, dato che $D \stackrel{P}{\longmapsto}^0 D'$ sse $D = D'$ e $\rho(D) \stackrel{P}{\longmapsto}^0 X$ sse $X = \rho(D)$.

passo induttivo) Dimostriamo separatamente le due implicazioni.

$\Longrightarrow$) Sia $D \stackrel{P}{\longmapsto}^k D'$, esiste allora $D'' \in \mathbb{D}$ tale che:

$$D \stackrel{P}{\longmapsto}^{k-1} D'' \stackrel{P}{\longmapsto} D' \text{ e } D'' \sqsubset D'' \bowtie \mu(\text{tree}(P)) = D' \qquad (1)$$

Per ipotesi induttiva, esiste $n \geq 0$ tale che:

$$\rho(D) \stackrel{P}{\longmapsto}_\rho^n \rho(D'') \qquad (2)$$

Inoltre, poiché $D'' \sqsubset D'$ abbiamo due possibilità: o $\rho(D'') = \rho(D')$, nel qual caso si ha la tesi direttamente da (2), o $\rho(D'') \sqsubset \rho(D')$. in quest'ultimo caso $\rho(D'') \sqsubset \rho(D'' \bowtie \mu(\text{tree}(P))) = \rho(D')$ che, grazie a (5.3.1) e alla definizione di $\bar{\bowtie}$ diventa $\rho(D'')\ \bar{\bowtie}\ \mu(\text{tree}(P))$. Per definizione di $\stackrel{P}{\longmapsto}_\rho$ e per (2) si ha infine $\rho(D) \stackrel{P}{\longmapsto}_\rho^n \rho(D'') \stackrel{P}{\longmapsto}_\rho \rho(D')$ che completa la prova.

$\Longleftarrow$) Sia $\rho(D) \stackrel{P}{\longmapsto}_\rho^n X'$, con $n \geq 0$. Per definizione di $\stackrel{P}{\longmapsto}_\rho$, esiste $X'' \in \rho(\mathbb{D})$ t.c. $\rho(D) \stackrel{P}{\longmapsto}_\rho^{n-1} X'' \stackrel{P}{\longmapsto}_\rho X'$. Allora, per ipotesi induttiva, esiste $k \geq 0$ e $D'' \in \mathbb{D}$ tale che:

$$\rho(D'') = X'' \text{ e } D \overset{P}{\longmapsto}{}^{k} D''. \tag{3}$$

Inoltre, per definizione di $\overset{P}{\longmapsto}_{\rho}$, $X'' \sqsubset X'' \mathbin{\bar{\bowtie}} \mu(\text{tree}(P)) = X'$. Allora, per (3) e (5.3.1) si ottiene $\rho(D'') \sqsubset \rho(D'' \bowtie \mu(\text{tree}(P)))$ da cui $D'' \sqsubset D'' \bowtie \mu(\text{tree}(P))$ per monotonia degli operatori di chiusura. A questo punto per definizione di $\overset{P}{\longmapsto}$ e per la seconda parte di (3) abbiamo che $D \overset{P}{\longmapsto}{}^{k} D'' \overset{P}{\longmapsto} D'' \bowtie \mu(\text{tree}(P))$. Basta ora osservare che, per (5.3.1), $\rho(D'' \bowtie \mu(\text{tree}(P))) = X'' \mathbin{\bar{\bowtie}} \mu(\text{tree}(P)) = X'$ e si arriva alla tesi. ■

Corollario 5.3.14 *Sia ρ un osservabile operazionale, $\boldsymbol{G} \in$ Goals, $P \in$ Progs. Allora:*

1. $\rho(\mathcal{B}[\![\boldsymbol{G} \;\; in \;\; P]\!]) = \mathcal{B}_{\rho}[\![\boldsymbol{G} \;\; in \;\; P]\!]$
2. $\rho(\mathcal{O}[\![P]\!]) = \mathcal{O}_{\rho}[\![P]\!]$

Corollario 5.3.15 *Sia ρ un osservabile operazionale, P un programma e $\boldsymbol{G}$ un goal. Allora:*

1. $\mathcal{O}_{\rho}[\![P]\!] \sqsubseteq \mathcal{F}_{\rho}[\![P]\!]$
2. $\mathcal{B}_{\rho}[\![\boldsymbol{G} \;\; in \;\; P]\!] \sqsubseteq \mathcal{Q}_{\rho}[\![\boldsymbol{G} \;\; in \;\; P]\!]$

Dimostrazione. Segue banalmente dal fatto che comunque la semantica di punto fisso è corretta, in quanto definita a partire da operatori astratti corretti, e quindi risulta $\mathcal{F}[\![P]\!] = \mathcal{O}[\![P]\!] \sqsubseteq \mathcal{F}_{\rho}[\![P]\!]$. Applicando ρ ad ambo i membri e tenendo conto del corollario 5.3.14 si ottiene il punto 1. Analogamente per il punto 2. ■

Capitolo 6

Proprietà del reticolo degli osservabili

In questo capitolo studieremo alcune proprietà del reticolo degli osservabili, che introduciamo formalmente nella sezione 6.1, nella quale dimostriamo anche come gli osservabili denotazionali, operazionali e perfetti ne costituiscano un sottoreticolo. Le sezioni 6.2 e 6.3 contengono, invece, alcuni risultati negativi circa l'esistenza di elementi massimi "interessanti" nel reticolo e il fatto che gli unici osservabili dotati di proprietà di composizionalità e precisione siano quelli sopra citati.

Infine, nella sezione 6.4 riprendiamo il problema degli elementi estremali restringendoci ai soli osservabili che siano concretizzazioni dell'osservabile di Herbrand e ricavando, in questo nuovo contesto, delle limitazioni superiori per gli osservabili perfetti, operazionali e denotazionali.

6.1 Il reticolo degli osservabili

Abbiamo più volte parlato di "reticolo di osservabili", ora è giunto il momento di definire esattamente cos'è. Visto che i nostri osservabili altro non sono se non operatori di chiusura sull'insieme delle collezioni, l'idea è quella di sovrapporre il nostro con il reticolo degli *uco*. Gli operatori di lub e glb diventano allora il lub e il glb standard per gli operatori di chiusura. perché la cosa funzioni occorre però dimostrare che gli osservabili costituiscono non solo un sottoinsieme ma anche un sottoreticolo degli *uco* su $\mathbb{D}$.

Teorema 6.1.1 *Siano ρ_i per $i \in I$ degli osservabili. Allora $\bigsqcup_{i\in I} \rho_i$ e $\bigsqcap_{i\in I} \rho_i$ sono osservabili.*

Dimostrazione. Quello che occorre verificare è che $\bigsqcup_{i\in I} \rho_i$ e $\bigsqcap_{i\in I} \rho_i$ possano essere messi sotto la forma $\lambda D\ \lambda \boldsymbol{G}.\ \rho^*(D(\boldsymbol{G}))$. Cominciamo con il *glb*.

$$
\begin{array}{lcl}
\Big(\bigsqcap\limits_{i\in I} \rho_i\Big)(D) & = & [\text{def. di glb tra uco}] \\
\bigsqcap\limits_{i\in I} \rho_i(D) & = & [\text{def. di osservabile}] \\
\bigsqcap\limits_{i\in I} \lambda \boldsymbol{G}.\rho_i^*(D(\boldsymbol{G})) & = & [\text{proprietà del glb per collezioni}] \\
\lambda \boldsymbol{G}.\bigcap\limits_{i\in I} \rho_i^*(D(\boldsymbol{G})) & &
\end{array}
$$

Si pone ora $\rho^*(S) = \bigcap_{i\in I} \rho_i^*(S)$ e si ottengono facilemte le seguenti proprietà:

- $\rho^*(\emptyset) = \emptyset$,
- ρ^* è un operatore di chiusura su $\mathrm{WFS}_{\boldsymbol{G}}$ per ogni $\boldsymbol{G}$. Mentre monotonia ed estensionalità sono banali, vediamo un istante di verificare che efettivamente ρ^* è idempotente. Chiaramente $\bigcap_{j\in I} \rho_j^*(S) \subseteq \rho_i^*(S)$ per ogni $i \in I$. Allora $\rho_i^*(\bigcap_{j\in I} \rho_j^*(S)) = \rho_i^*(S)$ da cui la tesi.
- se $S \equiv S'$, allora $\rho^*(S) \equiv \rho^*(S')$. Infatti $d \in \rho^*(S)$ implica $d \in \rho_i^*(S)$ per ogni $i \in I$. Allora, se $S \equiv S'$, poiché $\rho_i^*(S) \equiv \rho_i^*(S')$, esistono $d_i \in \rho_i^*(S')$ t.c. $d_i \equiv d$ per ogni $i \in I$. Ma poiché $S' \in \mathrm{PuWFS}$, $\mathrm{first}(d_i)$ è uguale per tutte le d_i, da cui segue $d_i = d_j = d'$ per ogni $i, j \in I$ e quindi $d' \in \bigcup_{i\in I} \rho_i^*(S') = \rho^*(S)$.

Da queste discende direttamente la tesi. Passiamo adesso al *lub* tra osservabili. Per ogni collezione D si ha:

$$\Big(\bigsqcup_{i\in I} \rho_i\Big) D = D \qquad \Leftrightarrow$$

[per def. di lub tra operatori di chiusura]

$$\forall i \in I.\ \rho_i(D) = D \qquad \Leftrightarrow$$

[per def. di osservabile]

$$
\begin{aligned}
&\forall i \in I,\ \boldsymbol{G} \in \text{Goals}.\ \rho_i^*(D(\boldsymbol{G})) = D(\boldsymbol{G}) && \Leftrightarrow \\
&\qquad [\text{per la logica del prim'ordine}] \\
&\forall \boldsymbol{G} \in \text{Goals},\ i \in I.\ \rho_i^*(D(\boldsymbol{G})) = D(\boldsymbol{G}) && \Leftrightarrow \\
&\qquad [\text{poiché } \rho_i \text{ è un uco su } \mathrm{WFS}_{\boldsymbol{G}}] \\
&\forall \boldsymbol{G} \in \text{Goals}.\ \Big(\bigsqcup_{i \in I} \rho_i^*|_{\mathrm{WFS}_{\boldsymbol{G}}}\Big) D(\boldsymbol{G}) = D(\boldsymbol{G}) && \Leftrightarrow \\
&\qquad [\text{estesionalità delle funzioni}] \\
&\Big(\lambda D.\ \lambda \boldsymbol{G}.\ \Big(\bigsqcup_{i \in I} \rho_i^*|_{\mathrm{WFS}_{\boldsymbol{G}}}\Big) D(\boldsymbol{G})\Big)(D) = D
\end{aligned}
$$

Ponendo $\rho^*(S) = (\bigsqcup_{i \in I} \rho_i^*|_{\mathrm{WFS}_{\boldsymbol{G}}})(S)$ per ogni $S \in \mathrm{WFS}_{\boldsymbol{G}}$ e verificando le solite tre proprietà di cui sopra si arriva alla tesi.

- $\rho^*(\emptyset) = \emptyset$ è banale. Si noti che la buona definizione di $\rho^*(S)$ dipende strettamente dal fatto che $\bigsqcup_{i \in I} \rho_i^*|_{\mathrm{WFS}_{\boldsymbol{G}}}$, qualunque sia $\boldsymbol{G}$, manda insiemi vuoti in vuoti,
- ρ^* è un operatore di chisura su WFS_G per come lo abbiamo definito,
- Sappiamo dall'equazione (3.2.2) che $\rho^*(S) = \bigcap\{S_1 \in \mathrm{WFS}_{\boldsymbol{G}} \supseteq S \mid \forall i \in I.\ \rho_i^*(S_1) = S_1\}$ se $S \in \mathrm{WFS}_{\boldsymbol{G}}$. poiché tutti i $\mathrm{WFS}_{\boldsymbol{G}\sigma}$ sono isomorfi per ogni renaming σ, (teorema 5.1.4) e visto che $\equiv$ è una congruenza per $\bigcap$ e per ρ_i^*, si ha che è una congruenza anche per ρ^*. ■

Adesso che abbiamo definito esattamente il nostro reticolo degli osservabili possiamo iniziare a studiare le relazioni tra i suoi elementi. Una prima cosa di interesse è vedere se le operazioni di glb e lub sono chiuse se ristrette ai soli osservabili denotazionali, perfetti e operazionali.

Teorema 6.1.2 *Siano ρ_i con $i \in I$ degli* uco *che godono di una o più delle proprietà* (4.3.23), (4.3.20), (4.3.21), (4.3.22), (4.3.25), (5.3.1). *Allora anche $\bigsqcap_{i \in I} \rho_i$ e $\bigsqcup_{i \in I} \rho_i$ godono delle stesse proprietà.*

Dimostrazione. È una diretta conseguenza dei teoremi 3.3.1 e 3.3.5. ■

Corollario 6.1.3 *Gli osservabili denotazionali, operazionali e perfetti costituiscono ognuno un sottoreticolo del reticolo di tutti gli osservabili.*

6.2 Elementi estremali

La nostra attenzione si sposta ora allo studio degli elementi massimali e minimali del nostro reticolo di osservabili. Non è difficile rendersi conto che gli elementi massimi e minimi del reticolo, ovvero $\rho_\top = \lambda D.\top_{\mathbb{D}}$ e $\rho_\perp = Id$ rispettivamente, sono perfetti. Tuttavia un osservabile come $\rho_\top$ è di utilità nulla, per cui quello che interessa sapere è, ad esempio, se esiste un massimo osservabile perfetto tra quelli da esso diversi.

Un errore che si può commettere è quello di pensare che il massimo, cioè il più astratto, osservabile perfetto diverso dal banale $\rho_\top$ sia quello dei risultanti. Vediamo che così non è.

Teorema 6.2.1 *L'osservabile dei risultanti con risposte ground ρ_r non è il massimo osservabile perfetto diverso da $\rho_\top$.*

Dimostrazione. Lo si dimostra esibendo un osservabile perfetto minore di $\rho_\top$ che non è confrontabile con ρ_r^*. Fissato $t \in \Pi$, definiamo:

$$\rho(D) = D \bowtie \{d \in \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}} \mid \exists \theta, \boldsymbol{G} \text{ t.c. } \mathrm{first}(d) = (t(\boldsymbol{x})\theta, \boldsymbol{G})\} \tag{1}$$

Grazie all'additività di $\bowtie$, si verifica facilmente che ρ è effettivamente un osservabile, ottenuto a partire da:

$$\rho^*(S) = S \bowtie \{d \in \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}} \mid \exists \theta, \boldsymbol{G} \text{ t.c. } \mathrm{first}(d) = (t(\boldsymbol{x})\theta, \boldsymbol{G})\}$$

Infatti:

- al solito, $\rho^*(\emptyset) = \emptyset$ banalmente,

- $\rho^*(S)$ è evidentemente monotono ed estensivo, occorre dimostrarne l'idempotenza. Sia allora $d \in \rho^*(\rho^*(S))$, l'unico caso non banale è se $d = d' :: d''$ con $d' \in \rho^*(S) \setminus S$ e $d'' \in \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}}$. Allora $d' = d''' :: d^{iv}$ con $d''' \in S$ e $d^{iv} \in \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}}$ con $\mathrm{first}(d^{iv}) = (t(\boldsymbol{x})\theta, \boldsymbol{G})$ per qualche θ e $\boldsymbol{G}$. Allora $d = (d''' :: d^{iv}) :: d'' = d''' :: (d^{iv} :: d'') \in \rho^*(S)$,

- se $S \equiv S'$ si ha $\rho^*(S) \equiv \rho^*(S')$ per la simile proprietà dell'operatore $\bowtie$ e per l'isomorfismo che si ha tra $\mathrm{WFS}_{\boldsymbol{G}}$ e $\mathrm{WFS}_{\boldsymbol{G}\sigma}$ con σ renaming.

Resta da dimostrare che ρ è perfetto. Cominciamo a verificare la precisione di ρ rispetto a $\bowtie$. Ci si limita alla sola diseguaglianza non banale: $\rho(D_1 \bowtie D_2) \sqsupseteq \rho(D_1) \bowtie \rho(D_2)$.

Sia $d \in \rho(D_1) \bowtie \rho(D_2)$. Ciò vuol dire che o $d \in \rho(D_1)$, ma in questo caso chiaramente è anche $d \in \rho(D_1 \bowtie D_2)$, oppure $d = d' :: d''$ con $d' \in \rho(D_1), d'' \in \rho(D_2)$. Analizziamo ora vari casi a seconda che sia $d' \in D_1$ e/o $d'' \in D_2$.

- $d' \in D_1, d'' \in D_2$. Chiaramente $d' :: d'' \in D_1 \bowtie D_2 \subseteq \rho(D_1 \bowtie D_2)$.
- $d' \in D_1, d'' \notin D_2$. Allora $d'' = d''' :: d^{iv}$, con $\mathrm{last}(d''') \equiv (t(\boldsymbol{x})\theta, \boldsymbol{G})$, $d''' \in D_2$ e quindi $d' :: d''' \in D_1 \bowtie D_2$ da cui $d' :: d'' = (d' :: d''') :: d^{iv} \in \rho(D_1 \bowtie D_2)$.
- $d' \notin D_1$. Sarà $d' = d''' :: d^{iv}$ con $\mathrm{last}(d''') \equiv (t(\boldsymbol{x})\theta, \boldsymbol{G})$, $d''' \in D_1$. Ma allora $d' :: d'' = d''' :: (d^{iv} :: d'') \in \rho(D_1) \subseteq \rho(D_1 \bowtie D_2)$.

Analogo discorso si può fare per l'operatore $\times$. Sia $d \in \rho(D_1) \times \rho(D_2)$, allora $d = d' \wedge d''$ con $d' \in \rho(D_1), d'' \in \rho(D_2)$. Abbiamo i soliti casi:

- $d' \in D_1, d'' \in D_2$. Banalmente $d' \wedge d'' \in D_1 \times D_2 \subseteq \rho(D_1 \times D_2)$.
- $d' \in D_1, d'' \notin D_2$. Si ottiene $d'' = d''' :: d^{iv}$, $\mathrm{last}(d''') \equiv (t(\boldsymbol{x})\theta, \boldsymbol{G})$. Si hanno a questo punto due possibilità diverse. Se d' é una refutazione, risulta $d = (d' \wedge d''') :: d^{iv}$, altrimenti $d = d' \wedge d'''$. In entrambi i casi, comunque, $d \in \rho(D_1 \wedge D_2)$ come desiderato.
- $d' \notin D_1$. Allora $d' = d''' :: d^{iv}$, $\mathrm{last}(d''') \equiv (t(\boldsymbol{x})\theta, \boldsymbol{G}) \neq \Box$ e quindi $d' \wedge d'' = (d''' \wedge d'') :: (d^{iv} \wedge \partial_{\mathrm{answer}(d''')}(d'')) \in \rho(D_1 \times D_2)$.

Infine esaminiamo l'operatore $\cdot$. Sia $d \in A \cdot \rho(S)$, ovvero $d = \partial_\gamma(d')$ con $A = \mathrm{first}(d')\gamma, d' \in \rho(S)$. Al solito:

- se $d' \in S$, banalmente $d \in \rho(A \cdot S)$.
- se $d' \notin S$, allora $d' = d'' :: d'''$, $\mathrm{last}(d'') \equiv (t(\boldsymbol{x})\theta, \boldsymbol{G})$. Dal lemma A.2.6 di [Meo96] si ricavano le seguenti due possibilità:

1. $\partial_\gamma(d') = \partial_\gamma(d'')$ e ciò implica banalmente $d \in \rho(A \cdot S)$,
2. $d = \partial_\gamma(d'') :: \partial_{\gamma'}(d''')$ e, per il lemma A.2.3, $\mathrm{last}(\partial_\gamma(d'')) \leq (t(\boldsymbol{x})\theta, \boldsymbol{G})$ per cui $d \in \rho(A \cdot S)$.

Dunque ρ è un osservabile perfetto. Inoltre è chiaramente incomparabile con ρ_r, e ciò verifica la nostra tesi. ∎

Si noti che il significato intuitivo dell'osservabile usato nella dimostrazione precedente non è affatto evidente, tanto che è difficile trovare un dominio astratto "elegante" che gli si adatti. L'idea è quella di avere una procedura, t nel nostro caso, la cui chiamata fa perdere ogni tipo di informazione sull'evoluzione della computazione.

L'esistenza di osservabili così strani è contemporaneamente il punto di forza e il punto debole del nostro framework, che è potente abbastanza da trattare le astrazioni più disparate, ma proprio per questo perde in quantità e qualità delle proprietà di cui si dispone.

6.3 Classi se e solo se

Vogliamo verificare adesso come il fatto di essere un osservabile perfetto o denotazionale sia una condizione sufficiente ma non necessaria per avere una semantica astratta precisa e OR-composizionale.

Sia dato, ad esempio:

$$\rho(D) = D + \left\{ t(\boldsymbol{x}) \xrightarrow{\epsilon}{}^{n} t(\boldsymbol{x}) \mid t(\boldsymbol{x}) \xrightarrow{\epsilon} t(\boldsymbol{x}) \in D \right\} \tag{6.3.1}$$

che è generato da

$$\rho^*(S) = S + \left\{ t(\boldsymbol{x}) \xrightarrow{\epsilon}{}^{n} t(\boldsymbol{x}) \mid t(\boldsymbol{x}) \xrightarrow{\epsilon} t(\boldsymbol{x}) \in S \right\} \tag{6.3.2}$$

Al solito, iniziamo a dimostrare che questo è effettivamente un osservabile. L'unica proprietà non banale è quella sulla varianza delle collezioni. Sia allora $D \equiv D'$ e $d \in \rho(D)$. Abbiamo che o $d \in D$, da cui esiste $d' \equiv d$ con $d' \in D' \subseteq \rho(D')$, oppure $d = t(\boldsymbol{x}) \xrightarrow{\epsilon}{}^{n} t(\boldsymbol{x})$ per un certo $t(\boldsymbol{x}) \in \mathrm{Goals}$, con

$d_1 = t(\boldsymbol{x}) \xrightarrow{\epsilon} t(\boldsymbol{x}) \in D$. Sia allora $d_2 = t(\boldsymbol{y}) \xrightarrow{\epsilon} t(\boldsymbol{y})$ la variante di d_1 che appartiene a D'. Si ha che $t(\boldsymbol{y}) \xrightarrow{\epsilon}^n t(\boldsymbol{y}) \in \rho(D')$ è una variante di d. Segue il fatto che $\rho(D) \equiv \rho(D')$.

Si verifica facilmente che tale osservabile non è preciso rispetto ad alcuno dei nostri operatori sulle collezioni. Per di più è anche una concretizzazione dell'osservable ρ_r dei risultanti, quindi rimane valido anche restringendoci agli osservabili che conservano l'informazione di successo degli alberi SLD. Nonostante ciò vedremo che per esso valgono tutte le proprietà rilevanti degli osservabili perfetti.

Prima di tutto si noti che ρ è trasparente rispetto agli operatori $\cdot, \times, \bowtie, +$ e alla restrizione su $\mathbb{I}$.

operatore $\cdot$) La diseguaglianza $\rho(A \cdot \rho(D)) \sqsupseteq A \cdot \rho(D)$ è ovvia, quindi possiamo limitarci a dimostrare quella inversa. Sia $d \in \rho(A \cdot \rho(D))$, per definizione di ρ abbiamo che o $d \in A \cdot \rho(D)$ come desiderato, oppure $d = t(\boldsymbol{x}) \xrightarrow{\epsilon}^n t(\boldsymbol{x})$ per qualche $t(\boldsymbol{x}) \in \text{Goals}$, con $\boldsymbol{x}$ vettore di variabili distinte. Ma allora $t(\boldsymbol{x}) \xrightarrow{\epsilon} t(\boldsymbol{x})$ è un elemento di $A \cdot \rho(D)$ e quindi $A = t(\boldsymbol{x})$ e $d' = t(\boldsymbol{y}) \xrightarrow{\epsilon} t(\boldsymbol{y}) \in \rho(D)$. Ma allora $t(\boldsymbol{y}) \xrightarrow{\epsilon} t(\boldsymbol{y}) \in D$ da cui $d \in A \cdot \rho(D)$.

operatore $\times$) Anche qui ci limitiamo alla diseguaglianza non banale. Sia $d \in \rho(\rho(D_1) \times \rho(D_2))$, al solito o $d \in \rho(D_1) \times \rho(D_2)$ come desiderato, oppure $d = t(\boldsymbol{x}) \xrightarrow{\epsilon}^n t(\boldsymbol{x})$ e $d' = t(\boldsymbol{x}) \xrightarrow{\epsilon} t(\boldsymbol{x}) \in \rho(D_1) \times \rho(D_2)$. A questo punto vi sono due possibilità: $d' \in \rho(D_1)$ e $\square \in \rho(D_2)$ o viceversa. Supponiamo di trovarci nel primo caso. Banalmente si ha $d' \in D_1$ da cui $d \in \rho(D_1) \subseteq \rho(D_1) \times \rho(D_2)$. Analogamente nel secondo caso.

operatore $\bowtie$) Sia al solito $d \in \rho(\rho(D_1) \bowtie \rho(D_2))$ e, per eliminare i casi banali, $d = t(\boldsymbol{x}) \xrightarrow{\epsilon}^n t(\boldsymbol{x})$. Allora $d' = t(\boldsymbol{x}) \xrightarrow{\epsilon} t(\boldsymbol{x}) \in \rho(D_1) \bowtie \rho(D_2)$. Ci sono a questo punto due sole possibilità: o $d' \in \rho(D_1)$, o $d' \in \rho(D_2)$ e $t(\boldsymbol{x}) \in \rho(D_1)$. Nel primo caso chiaramente $d \in \rho(D_1)$ e quindi $d \in \rho(D_1) \bowtie \rho(D_2)$. Analogamente nel secondo caso.

operatore $\sum$) Sia $d \in \rho(\sum_{i \in I} \rho(D_i))$ e $d = t(\boldsymbol{x}) \xrightarrow{\epsilon}^n t(\boldsymbol{x})$. Allora $d' =$

$t(\boldsymbol{x}) \xrightarrow{\epsilon} t(\boldsymbol{x}) \in \rho(D_i)$ per qualche $i \in I$. Segue $d \in \rho(D_i)$ e quindi $d \in \sum_{i \in I} \rho(D_i)$.

restrizione su $\mathbb{I}$) La dimostrazione procede in maniera analoga a quella per gli altri operatori.

Sfruttando allora il teorema 3.4.3 e il fatto che $\rho(D) = D$ per $D \sqsubseteq Id$, si dimostra facilmente che:

- $\tilde{\mu} = \mu|_{\rho(\mathbb{D})}$,
- $\tilde{\nu} = \nu|_{\rho(\mathbb{D})}$,
- $X \overset{P}{\longmapsto}_\rho X'$ sse $X' = X \bowtie \mu(\rho(\text{tree}(P)))$,
- $\mathcal{P}_\rho[\![P]\!]_X = Id|_{\mathbb{I}} + (\rho(\text{tree}(P)) \bowtie \nu(X))$

Andiamo adesso a verificare che $\mathcal{B}_\rho$ è preciso. Per far ciò dimostriamo che per ogni $X \in \rho(\mathbb{D})$, $X \overset{P}{\longmapsto}_\rho Y$ implica $Y \subseteq \sum_{n \geq 0} Y_n$ con $X \overset{P}{\longmapsto}^n Y_n$, dove $\overset{P}{\longmapsto}^n$ o $\overset{P}{\longmapsto}^n_\rho$ indichano una derivazione del sistema di transizione di al più di n passi. Come conseguenza diretta si ha $X \overset{P}{\longmapsto}^i_\rho Y \Rightarrow Y \subseteq \sum_{n \geq 0} Y_n$ per ogni $i \in \mathbb{N}$ e infine, ponendo $X = \rho(Id|_{\boldsymbol{G}})$ e sommando su tutti gli i:

$$\mathcal{B}_\rho = \mathcal{B}$$

Sia allora $d \in Y$. Per quanto detto sopra abbiamo due possibilità:

- $d \in X$ che implica $d \in Y_0$ come desiderato
- $d = d' :: (\partial_\gamma(d'') \wedge d''')$, con $d' \in X$, $d'' \in \rho(\text{tree}(P))$, $d''' \in Id$. Qui abbiamo a sua volta i seguenti sottocasi:

  1. $d'' \in \text{tree}(P)$. È chiaro che in questo caso $d \in Y_1$.
  2. $d'' \notin \text{tree}(P)$. Vuol dire che $d'' = t(\boldsymbol{x}) \xrightarrow{\epsilon}^n t(\boldsymbol{x})$ e $t(\boldsymbol{x}) \xrightarrow{\epsilon} t(\boldsymbol{x}) \in \text{tree}(P)$. Ne segue che se $\text{first}(d'') = \boldsymbol{G}$, $d = d' :: (t(\boldsymbol{x})\gamma, \boldsymbol{G}) \xrightarrow{\epsilon}^n (t(\boldsymbol{x})\gamma, \boldsymbol{G})$ e che $d' :: (t(\boldsymbol{x})\gamma, \boldsymbol{G}) \xrightarrow{\epsilon}^i (t(\boldsymbol{x})\gamma, \boldsymbol{G}) \in Y_i$ per ogni i, da cui la tesi.

Abbiamo dunque ottenuto l'eguaglianza $\mathcal{B} = \mathcal{B}_\rho$ che è una proprietà ancora più forte della precisione della semantica operazionale astratta. Dato infatti $P \in \text{Progs}$, si ottiene $\mathcal{B}_\rho[\![P]\!] = \mathcal{B}[\![P]\!] \sqsubseteq \rho(\mathcal{B}[\![P]\!])$ mentre la diseguaglianza inversa è banale, poiché $\mathcal{B}_\rho$ è corretta.

Si ottengono facilmente anche i seguenti risultati:

- $\mathcal{O}_\rho[\![P]\!] = \left[\tilde{\sum}\{\mathcal{B}_\rho[\![G \text{ in } P]\!]\}_{p(x)\in\text{Goals}}\right]_{\equiv} = \mathcal{O}[\![P]\!]$
- $\mathcal{B}_\rho[\![A \texttt{ in } P]\!] = A\tilde{\cdot}\mathcal{B}_\rho[\![A \texttt{ in } P]\!]$
- $\mathcal{B}_\rho[\![\boldsymbol{G}_1, \boldsymbol{G}_2 \text{ in } P]\!] = \mathcal{B}_\rho[\![\boldsymbol{G}_2 \texttt{ in } P]\!]\tilde{\times}\mathcal{B}_\rho[\![\boldsymbol{G}_2 \texttt{ in } P]\!]$
- $\mathcal{O}_\rho[\![P_1 \cup P_2]\!] = \mathcal{O}_\rho[\![P_1]\!] \uplus \mathcal{O}_\rho[\![P_2]\!]$

In maniera analoga a quanto vista sopra si dimostra anche la precisione della semantica denotazionale astratta. Stavolta va verificato che per ogni programma P, se $d \in \mathcal{P}_\rho[\![P]\!] \uparrow n$, esiste $m \geq n$ tale che $d \in \mathcal{P}[\![P]\!] \uparrow m$. Da qui, sommando per tutti gli n, discende direttamente $\mathcal{F}_\rho = \mathcal{F}$.

In particolare, dimostriamo che se $X \in \rho(\mathbb{D})$, esiste n per cui $\mathcal{P}_\rho[\![P]\!]_X \sqsubseteq \mathcal{P}^n[\![P]\!]_X$. Sia d appartenente alla prima collezione della diseguaglianza. Potrebbe essere $d \in Id|_{\mathbb{I}}$ ma allora sarebbe banalmente $d \in \mathcal{P}[\![P]\!]_X$. Altrimenti può essere $d = d' :: d''$ con $d' \in \rho(\text{tree}(P))$, $d'' \in \nu(X)$. L'unico caso non banale si ha quando $d' \notin \text{tree}(P)$, ovvero $d' = t(\boldsymbol{x}) \xrightarrow{\epsilon}{}^{i} t(\boldsymbol{x})$. Allora $d''' = t(\boldsymbol{x}) \xrightarrow{\epsilon} t(\boldsymbol{x}) \in \text{tree}(P)$, per cui $d''' :: d'' \in \mathcal{P}[\![P]\!]_X$ e, in generale, $t(\boldsymbol{x}) \xrightarrow{\epsilon}{}^{j} t(\boldsymbol{x}) :: d' \in \mathcal{P}^j[\![P]\!]_X$ per ogni j da cui la proprietà cercata.

Si ha allora:

- $\mathcal{F}_\rho[\![P]\!] = \mathcal{F}[\![P]\!] = \mathcal{O}_\rho[\![P]\!]$,
- $\mathcal{Q}_\rho[\![\boldsymbol{G} \texttt{ in } P]\!] = \mathcal{Q}[\![\boldsymbol{G} \texttt{ in } P]\!]$

6.4 Osservabili e modello di Herbrand

Come abbiamo visto il nostro dominio degli osservabili è molto complesso e comprende degli elementi piuttosto particolari la cui utilità è perlomeno discutibile. Tale complessità, se da un lato ci consente una vasta libertà di movimento nella scelta dell'osservabile che più si confà alle nostre esigenze, dall'altra limita le possibilità di analisi dell'insieme $\mathbb{D}$ e delle proprietà di cui esso gode.

Un'idea è allora quella di restringerci solamente a quelle classi di osservabili che ci sembrano più interessanti. Dato che attualmente stiamo tralasciando lo studio di osservabili per l'analisi statica dei programmi concentrandoci invece su quelli che ci consentono di definire una gerarchia di semantiche a livello dinamico, una caratteristica che possiamo tranquillamente imporre a tutti i nostri osservabili è di essere concretizzazioni di quello di Herbrand, in maniera simile a quanto fatto in [Gia96]. Definiamo allora

$$\rho_h^*(S) = \begin{cases} \emptyset & \text{se } S = \emptyset \\ \{\boldsymbol{G} \longrightarrow^* \boldsymbol{B} \mid \boldsymbol{B} \neq \Box\} & \text{se } S \neq \emptyset \text{ non contiene refutazioni} \\ \top_{\mathrm{WFS}_{\boldsymbol{G}}} & \text{altrimenti} \end{cases} \tag{6.4.1}$$

per ogni $S \in \mathrm{WFS}_{\boldsymbol{G}}$ o alternativamente, come iniezione di Galois, ponendo $\mathcal{D}_{\boldsymbol{G}} = \{\bot, \boxslash, \top\}$ per ogni $\boldsymbol{G}$ e

$$\alpha^*(S) = \begin{cases} \bot & \text{se } S = \emptyset \\ \boxslash & \text{se } S \text{ non contiene refutazioni} \\ \top & \text{altrimenti} \end{cases} \tag{6.4.2}$$

con γ^* che si ottiene direttamente dalla definizione di α^*.

Che ρ_h sia un osservabile si vede abbastanza facilmente. Si noti, in particolare, che se $S \equiv S'$, le collezioni $\rho_h^*(S)$ e $\rho_h^*(S')$ sono ottenuto dallo stesso caso della definizione e quindi sono tra loro varianti.

Si noti che ρ_h^* è un osservabile additivo ottenuto definendo $\propto$ come segue:

$$d \propto d' \Leftrightarrow \mathrm{last}(d) \neq \Box \vee \mathrm{last}(d') = \Box \tag{6.4.3}$$

Chiaramente l'insieme di tutti gli osservabili $\rho \sqsubseteq \rho_h$ è un sottoreticolo completo dell'insieme **Obs**. Assume quindi interesse stabilire quali sono gli osservabili denotazionali, operazionali e perfetti più astratti limitatamente alle concretizzazioni di ρ_h.

A scopo di completezza, non ci limiteremo solo allo studio di queste categorie di osservabili note ma esamineremo anche le limitazioni che nascono dai singoli assiomi di precisione degli operatori semantici. Il primo risultato che si ottiene è il seguente:

Teorema 6.4.1 *L'osservabile ρ_h gode della proprietà* (4.3.25).

Dimostrazione. Sfruttiamo il teorema 5.2.4 per limitarci alle collezioni di PuWFS e consideriamo la sola diseguaglianza non banale. Sia $d \in S_1 \bowtie \rho_h(S_2)$. Abbiamo varie possibilità.

- $d \in S_1$, per cui si ottiene banalmente $d \in \rho_h(S_1 \bowtie S_2)$.
- $d = d_1 :: d_2$, con $d_1 \in S_1$, $d_2 \in \rho_h(S_2)$, per cui si distinguono altri sottocasi:
  1. $d_2 \in S_2$, da cui ancora una volta $d \in \rho_h(S_1 \bowtie S_2)$.
  2. $d_2 \notin S_2, \mathrm{last}(d_2) \neq \Box$, per cui $\mathrm{last}(d) \neq \Box$. poiché $\rho_h(S_1 \bowtie S_2)$ contiene tutte le derivazioni che non sono refutazioni si ha $d \in \rho_h(S_1 \bowtie S_2)$.
  3. $d_2 \notin S_2, \mathrm{last}(d_2) = \Box$, per cui esiste almeno una refutazione in S_2, che ha ovviamente lo stesso goal iniziale di d_2. Allora esiste una refutazione anche in $S_1 \bowtie S_2$ e quindi $d \in \rho_h(S_1 \bowtie S_2)$ ∎

6.4.1 Massimo osservabile denotazionale

Come discusso prima andiamo a vedere quali sono gli osservabili più astratti, in ognuna delle nostre classi di interesse, che siano concretizzazioni dell'osservabile di successo di Herbrand. Cominciamo introducendo l'osservabile delle *risposte calcolate ground* ρ_g.

Diamo le due versioni con operatori di chiusura e con inserzioni di Galois per favorire l'intuizione. Sia $\rho_g^*(\emptyset) = \emptyset$ e $\forall S \in \mathrm{WFS}_{\boldsymbol{G}} \neq \emptyset$:

$$\begin{aligned} \rho_g^*(S) = \big\{ d = \boldsymbol{G} \xrightarrow{\theta}{}^* \boldsymbol{B} \mid \boldsymbol{B} = \Box \Rightarrow \forall \boldsymbol{G'} \leq \boldsymbol{G}\theta,\ \boldsymbol{G'} \text{ ground} \\ \exists d' = \boldsymbol{G} \xrightarrow{\theta'}{}^* \Box \in S \text{ t.c. } \boldsymbol{G'} \leq \boldsymbol{G}\theta' \big\} \end{aligned} \tag{6.4.4}$$

ovvero

$$\begin{aligned} &\mathcal{D}_{\boldsymbol{G}} = \mathcal{P}\{\boldsymbol{G}\gamma \in \text{Goals} \mid \boldsymbol{G}\gamma \text{ ground}\} \cup \{\bot\} \\ &\alpha_g^*(S) = \{\boldsymbol{G}\gamma \in \mathcal{D}_{\boldsymbol{G}} \mid \exists d = \boldsymbol{G} \xrightarrow{\theta}{}^* \Box \in S \text{ t.c. } \boldsymbol{G}\gamma \leq \boldsymbol{G}\theta\} \end{aligned} \tag{6.4.5}$$

se $S \neq \emptyset$, altrimenti $\alpha_g^*(\emptyset) = \bot$.

Dobbiamo verificare che questo è effettivamente un osservabile. Esaminiamo i vari punti singolarmente:

1. $\rho_g^*(\emptyset) = \emptyset$ per definizione.

2. La monotonia è evidente. Per quanto riguarda l'estensionalità, se $d \in S$ allora S non è vuoto e quindi $\rho_g^*(S)$ contiene tutte le derivazioni che non sono refutazioni, per cui se $\text{last}(d) \neq \Box$, allora $d \in \rho_g^*(S)$. Se invece $d = \boldsymbol{G} \xrightarrow{\theta}{}^* \Box$, per ogni $\boldsymbol{G}' \leq \boldsymbol{G}\theta$, $\boldsymbol{G}'$ ground, esiste una derivazione d' in S, che è proprio la nostra d, tale che $\text{last}(d') = \Box$ e $\boldsymbol{G}' \leq \boldsymbol{G}\,\text{answer}(d')$.

   Infine, per l'idempotenza, sia $d \in \rho_g^*(\rho_g^*(S))$. Allora $\rho_g^*(S)$ non è vuoto e contiene di conseguenza almeno tutte le derivazioni SLD che non siano refutazioni. Se $\text{last}(d) \neq \Box$ allora si ha banalmente $d \in \rho_g^*(S)$, altrimenti per ogni $\boldsymbol{G}' \leq \boldsymbol{G}\,\text{answer}(d)$ ground esiste $d' \in \rho_g^*(S)$ t.c. $\text{last}(d') = \Box$, $\boldsymbol{G}' \leq \boldsymbol{G}\,\text{answer}(d')$. A sua volta esiste $d'' \in S$ t.c. $\text{last}(d'') = \Box$ e $\boldsymbol{G}' \leq \boldsymbol{G}\,\text{answer}(d'')$ da cui la tesi, ovvero $d \in \rho_g^*(S)$.

3. Sia $S \equiv S'$ con ridenominazione σ e $d \in \rho_g^*(S)$. Ciò vuol dire che nè S nè S' sono vuoti. Se $\text{last}(d) \neq \Box$, poiché $S' \neq \emptyset$, $\rho_g^*(S')$ contiene tutte le derivazioni che non sono refutazioni e quindi anche $\partial_\sigma(d) \equiv d$. Anche quando $\text{last}(d) = \Box$, però, si ha sempre $d' = \partial_\sigma(d) \in \rho_g^*(S')$. Infatti per ogni $\boldsymbol{G}' \leq \boldsymbol{G}\sigma\,\text{answer}(d')$ ground, dal lemma A.2.2, si ricava che $\boldsymbol{G}'$ è istanza di $\boldsymbol{G}\,\text{answer}(d)$, per cui esiste $d_1 \in S$ con $\text{last}(d_1) = \Box$ e $\boldsymbol{G}' \leq \boldsymbol{G}\,\text{answer}(d_1)$. Ma, allora, $d_1' = \partial_\sigma(d_1)$ appartiene ad S' perche è $S \equiv S'$ e inoltre $\text{last}(d_1') = \Box$ e, sempre per A.2.2, $\boldsymbol{G}' \leq \boldsymbol{G}\sigma\,\text{answer}(d_1')$. Per definizione di ρ_g^* si ha automaticamente $d' \in \rho_g^*(S')$.

Si noti che questo osservabile non è additivo. Sia data infatti la segnatura col solo predicato p, termini a e b e variabili x_i con $i \in \mathbb{N}$. Abbiamo:

$$\rho_g^*\{p(x_1), p(x_1) \xrightarrow{\{x_1/a\}} \Box\} \not\supseteq \{p(x_1) \xrightarrow{\{x_1/x_2\}} \Box\}$$

$$\rho_g^*\{p(x_1), p(x_1) \xrightarrow{\{x_1/b\}} \Box\} \not\supseteq \{p(x_1) \xrightarrow{\{x_1/x_2\}} \Box\}$$

ma

$$\rho_g^*\{p(x_1), p(x_1) \xrightarrow{\{x_1/a\}} \Box, p(x_1) \xrightarrow{\{x_1/b\}} \Box\} \supseteq \{p(x_1) \xrightarrow{\{x_1/x_2\}} \Box\}$$

per cui ρ_g^* non è additiva. Imponendo che le nostre collezioni siano rappresentazioni di alberi SLD finiti, quindi con un numero finito di derivazioni massimali, e limitandoci alle sole segnature con una cardinalità almeno numerabile di termini, si riesce a dimostrare la finita additività di ρ_g^*.

Lemma 6.4.2 *L'osservabile ρ_g è denotazionale*

Dimostrazione. Controlliamo separatamente la precisione rispetto ai vari operatori semantici.

operatore $\cdot$) Sia $d \in A \cdot \rho_g(D)$. Allora $d = \partial_\gamma(d')$ per un certo $d' \in \rho_g(D)$ e $A = \text{first}(d')\gamma$. Evidentemente D deve contenere almeno una derivazione con goal iniziale $\text{first}(d')$ per cui $A \cdot D$ è sicuramente non vuota. Ne segue che $d \in \rho_g(A \cdot D)$ se $\text{last}(d) \neq \Box$.

Le cose si complicano quando $\text{last}(d) = \Box$. Sia dato un qualunque $A' \leq A\,\text{answer}(d)$ ground, per il lemma A.2.3 è $A' \leq \boldsymbol{G}\,\text{answer}(d')$ e quindi, per definizione di ρ_g, esiste $d'' \in D$ t.c. $\text{last}(d'') = \Box$, $A' \leq \boldsymbol{G}\,\text{answer}(d'')$ e $\text{var}(\text{answer}(d'')) \cap \text{var}(\gamma) \subseteq \text{var}(\boldsymbol{G})$. Allora $d''' = \partial_\gamma(d'') \in A \cdot D$ ha, per il lemma A.2.3, $A\,\text{answer}(d''') = \boldsymbol{G}\,\text{answer}(d'')\sigma$ con $\text{last}(d''') = \Box$ e $\sigma \in \text{mgu}(A, \boldsymbol{G}\,\text{answer}(d''))$. poiché $A = \boldsymbol{G}\gamma$, per il lemma A.1.3 segue $A' \leq \boldsymbol{G}\,\text{answer}(d'')\sigma$ da cui $d \in A \cdot D$ come volevasi dimostrare.

operatore $\times$) Sia $d = (\boldsymbol{G}_1, \boldsymbol{G}_2) \xrightarrow{\theta}{}^* \boldsymbol{B} \in \rho_g(D_1) \times \rho_g(D_2)$, per cui $d = d_1 \wedge d_2$, $d_1 = \boldsymbol{G}_1 \xrightarrow{\theta_1}{}^* \boldsymbol{B_1} \in \rho_g(D_1)$ e $d_2 = \boldsymbol{G}_2 \xrightarrow{\theta_2}{}^* \boldsymbol{B_2} \in \rho_g(D_2)$. D_1 e D_2 devono contengono allora almeno una derivazione ognuno, per cui lo stesso avviene con $D_1 \times D_2$. Quindi se $\text{last}(d) \neq \Box$, si ottiene direttamente $d \in \rho(D_1 \times D_2)$.

Sia invece $\mathrm{last}(d) = \Box$ e $(\boldsymbol{G}'_1, \boldsymbol{G}'_2) \leq (\boldsymbol{G}_1, \boldsymbol{G}_2)\theta$ ground. Chiaramente $\boldsymbol{G}'_1 \leq \boldsymbol{G}_1\theta_1$ per cui esiste $d'_1 \in D_1$ tale che $\boldsymbol{G}'_1 \leq \boldsymbol{G}_1\,\mathrm{answer}(d'_1)$. D'altronde, grazie al lemma A.2.5, si ha $\boldsymbol{G}'_2 \leq \boldsymbol{G}_2\theta \leq \boldsymbol{G}_2\theta_2$ ovvero esiste $d'_2 \in D_2$ tale che $\boldsymbol{G}'_2 \leq \boldsymbol{G}_2\,\mathrm{answer}(d'_2)$. È possibile scegliere d'_1 e d'_2 in modo che non abbiano variabili in comune eccetto quelle di $\mathrm{var}(\boldsymbol{G}_1) \cap \mathrm{var}(G_2)$.

Consideriamo ora la derivazione $d'_1 \wedge d'_2$, siano θ'_1 e θ'_2 le risposte parziali di d'_1 e d'_2 rispettivamente e θ' quella di $d'_1 \wedge d'_2$. Dal lemma A.2.8 di [Meo96], ricaviamo che $(\boldsymbol{G}_1, \boldsymbol{G}_2)\theta' = (\boldsymbol{G}_1, \boldsymbol{G}_2)\phi$ con $\phi \in \mathrm{mgu}((\boldsymbol{G}_1, \boldsymbol{G}_2), (\boldsymbol{G}_1\theta'_1, \boldsymbol{G}_2\theta'_2))$. Per il lemma A.1.4, poiché $\boldsymbol{G}'_1 \leq \boldsymbol{G}_1\theta'_1$ e $\boldsymbol{G}'_2 \leq \boldsymbol{G}_2\theta'_2$, segue $(\boldsymbol{G}'_1, \boldsymbol{G}'_2) \leq (\boldsymbol{G}_1\theta'_1, \boldsymbol{G}_2\theta'_2)$ e successivamente $(\boldsymbol{G}'_1, \boldsymbol{G}'_2) \leq (\boldsymbol{G}_1, \boldsymbol{G}_2)\theta$ dal lemma A.1.3. Segue allora $d_1 \wedge d_2 \in \rho_g(D_1 \times D_2)$.

operatore $\bowtie$) Sia $d = \boldsymbol{G} \xrightarrow{\theta}{}^{*} \boldsymbol{B} \in D_1 \bowtie \rho_g(D_2)$, per cui o $d \in D_1$, ma allora è banale che $d \in \rho_g(D_1 \bowtie D_2)$, o $d = d_1 :: d_2$ con $d_1 = \boldsymbol{G} \xrightarrow{\theta_1}{}^{*} \boldsymbol{G}_2\sigma \in D_1$ e $d_2 = \boldsymbol{G}_2 \xrightarrow{\theta_2}{}^{*} \boldsymbol{B} \in \rho_g(D_2)$, per un opportuno renaming σ tale che $\mathrm{dom}(\sigma) \subseteq \boldsymbol{G}_2$. È evidente che $\rho_g(D_2)$, e quindi anche D_2, ha almeno una derivazione con primo goal variante di $\boldsymbol{G}_2\sigma$. Ne segue che se $\boldsymbol{B} \neq \Box$, poiché $D_1 \bowtie D_2$ risulta avere almeno una derivazione, $d \in \rho_g(D_1 \bowtie D_2)$.

Diverso, come sempre, il caso in cui $\boldsymbol{B} = \Box$. Sia $\boldsymbol{G}' \leq \boldsymbol{G}\theta$ ground e, per il lemma A.2.4, $\boldsymbol{G}\theta = \boldsymbol{G}\theta_1\sigma^{-1}\theta''_2$, con $\theta''_2 \equiv \theta_2$. Supposto $\boldsymbol{G}' = \boldsymbol{G}\theta\eta$, abbiamo che $\boldsymbol{G}_2\theta''_2\eta$ è una istanza di $\boldsymbol{G}_2\theta_2$. Sia allora $\boldsymbol{G}'_2 = \boldsymbol{G}_2\theta''_2\eta\beta$ ground, per cui esiste $d'_2 = \boldsymbol{G}_2 \xrightarrow{\theta'_2}{}^{*} \Box \in D_2$ con $\boldsymbol{G}'_2 \leq \boldsymbol{G}_2\theta'_2$ e $\mathrm{var}(\theta'_2) \cap \mathrm{var}(\theta_1\sigma^{-1}) \subseteq \mathrm{var}(\boldsymbol{G}_2)$. Vogliamo dimostrare che, dato $d_1 :: d'_2 = \boldsymbol{G}_1 \xrightarrow{\theta'}{}^{*} \Box$ con $\theta' \equiv \theta_1\sigma^{-1}\theta'_2$, si ha $\boldsymbol{G}' = \boldsymbol{G}\theta_1\sigma^{-1}\theta''_2\eta\beta \leq \boldsymbol{G}\theta_1\sigma^{-1}\theta'_2$. Ci si può limitare a dimostrare che $\theta''_2\eta\beta \leq \theta'_2$ sulle variabili di $\boldsymbol{G}\theta_1\sigma^{-1}$. Ma la cosa è evidente visto che sulle variabili di $\boldsymbol{G}_2$ risulta $\boldsymbol{G}_2\theta''_2\eta\beta = \boldsymbol{G}'_2 \leq \boldsymbol{G}_2\theta'_2$ mentre per le restanti θ'_2 non ha alcun legame.

Sembra abbastanza ragionevole supporre che sia proprio ρ_g il più astratto degli osservabili denotazionali (tra quelli che conservano l'informazione sul successo di un albero SLD). E, infatti, vale il seguente

Teorema 6.4.3 *Il lub degli osservabili $\rho \sqsubseteq \rho_h$ per cui vale una delle proprietà* (4.3.20) *o* (4.3.21) *rispetto al primo argomento è l'osservabile ρ_g.*

Dimostrazione. Dimostriamo in realtà l'implicazione inversa, ovvero che un $\rho \not\sqsubseteq \rho_g$ non può verificare nè (4.3.20) nè (4.3.21).

Se $\rho \not\sqsubseteq \rho_g$, per qualche $\boldsymbol{G}$ esiste $S \in \mathrm{WFS}_{\boldsymbol{G}}$, $d \in \rho^*(S)$ refutazione e $\gamma \leq \mathrm{answer}(d)$ ground che non è istanza di nessuna $\mathrm{answer}(d')$ con d' refutazione di S.

Allora $\partial_\gamma(d) \in \boldsymbol{G}\gamma \cdot \rho(S) \sqsubseteq \rho(\boldsymbol{G}\gamma \cdot \rho(S))$. Viceversa $\boldsymbol{G}\gamma \cdot S$ non contiene alcuna refutazione e quindi lo stesso vale per $\rho(\boldsymbol{G}\gamma \cdot \rho(S))$ visto che è $\rho \sqsubseteq \rho_h$. Dunque ρ non è preciso rispetto a $\cdot$, invalidando la (4.3.20).

D'altronde siano $x_1, \ldots, x_n$ le variabili di $\boldsymbol{G}$; prendiamo una derivazione d' del goal $t(x_1, \ldots, x_n)$ con risposta calcolata γ e poniamo $S = \mathrm{WFS}\{d'\}$. Avremo che $d \wedge d' \in \rho(S) \times S_1$ è una refutazione del goal $\boldsymbol{G}, t(x_1, \ldots, x_n)$ mentre $S \times S_1$, e quindi anche $\rho(S \times S_1)$, non contengono refutazioni. Dunque ρ non è preciso rispetto a $\times$, neanche limitatamente al primo argomento. ∎

Corollario 6.4.4 *Il lub degli osservabili denotazionali concretizzazioni di ρ_h è l'osservabile delle risposte calcolate ground ρ_g.*

6.4.2 Limitazione superiore per il massimo osservabile operazionale

Cominciamo col verificare quali sono le condizioni sotto le quali un osservabile concretizzazione di ρ_h possa essere preciso rispetto a $\bowtie$. Introduciamo allora un nuovo osservabile. Per ogni $S \in \mathrm{WFS}_{\boldsymbol{G}}$ sia

$$\rho_l^*(S) = \mathrm{gwf}\big\{d = \boldsymbol{G} \longrightarrow^* \boldsymbol{B} \mid \exists d' \in S \text{ t.c. } \mathrm{last}(d') = \Box \vee \boldsymbol{B} \equiv \mathrm{last}(d')\big\} \tag{6.4.6}$$

Che si tratti effettivamente di un osservabile è cosa abbastanza evidente, in particolar modo sfruttando il teorema 5.2.5, per cui tralasciamo i dettagli della dimostrazione.

È anche banale verificare che ρ_l è preciso rispetto a $\bowtie$ grazie al teorema 5.2.6, dove $\propto$ è immediatamente deducibile dalla definizione. Infatti, siano d_1,

d_2, d'_1 e d'_2 derivazioni con $d_1 \propto d'_1$ e $d_2 \propto d_2$ tali che $d_1 :: d_2$ esista. Ci sono a questo punto due casi. Il primo si verifica quando $\mathrm{last}(d'_1) = \Box$, nel qual caso è $d_1 :: d_2 \propto d'_1$. Altrimenti $\mathrm{last}(d'_1) \equiv \mathrm{last}(d_1)$, $\mathrm{first}(d'_2) = \mathrm{first}(d_2) \equiv \mathrm{last}(d_1)$, per cui $d'_1 :: d'_2$ esiste e, evidentemente, ha come ultimo goal $\mathrm{last}(d'_2)$. Questo a sua volta può essere uguale a una variante di $\mathrm{last}(d_2) \equiv \mathrm{last}(d_1 :: d_2)$ oppure a $\Box$. In ambo i casi si ha, comunque, $d_1 :: d_2 \propto d'_1 :: d'_2$.

Teorema 6.4.5 *Il lub degli osservabili minori di ρ_h che soddisfano* (5.3.1) *è l'osservabile ρ_l.*

Dimostrazione. Come prima dimostriamo l'implicazione inversa. Se $\rho \not\sqsubseteq \rho_l$, per qualche $\boldsymbol{G}$ esiste $S \in \mathrm{WFS}_{\boldsymbol{G}}$ e $d \in \rho^*(S)$ tale che non vi è alcun $d' \in S$ con $\mathrm{last}(d) = \mathrm{last}(d')$ o con $\mathrm{last}(d') = \Box$.

Poniamo $S_1 = \mathrm{WFS}\{d_1\}$ con d_1 refutazione di $\mathrm{last}(d)$. Abbiamo che $\rho(S) \bowtie S_1$ contiene almeno la refutazione $d :: d_1$ mentre $S \bowtie S_1$, e quindi anche $\rho(S \bowtie S_1)$, non ne hanno alcuna. ∎

Chiaramente poiché ρ_l è preciso rispetto a $\bowtie$, è anche il massimo osservabile minore di ρ_h che soddisfa (4.3.22). La necessità di trattare in maniera speciale il caso di refutazioni deriva dal fatto che l'operatore $\bowtie$ è estensivo sul primo argomento. Se si eliminasse dalla definizione di $S \bowtie S_1$ il lub con S, non ci sarebbe bisogno di questa asimmetria.

A questo punto è ovvio che il massimo osservabile operazionale (o perfetto) deve essere più concreto sia di ρ_g che di ρ_l e quindi, in sostanza, sarà minore del loro prodotto ridotto $\rho_g \sqcap \rho_l$.

Capitolo 7

Operatori di raffinamento sugli osservabili

Abbiamo visto nei capitoli precedenti la struttura del reticolo **Obs** degli osservabili. Adesso vogliamo esaminare alcuni operatori su tale dominio, detti di *raffinamento*, che ci consentono, a partire da osservabili interessanti, di derivare osservabili più concreti.

7.1 Dipendenze funzionali

Abbiamo introdotto la nozione di dipendenza funzionale nel capitolo 3. Qui vogliamo vedere come essa si possa applicare ai nostri osservabili. In generale potremmo voler utilizzare qualunque operatore additivo a sinistra per calcolare le dipendenze, ma vedremo che saranno necessarie ulteriori ipotesi in modo da ottenere effettivamente un osservabile e non un semplice uco.

Dati ρ_1 e ρ_2 osservabili e $\odot$ operatore binario additivo a sinistra, si pone $\rho = \rho_1 \to^{\odot} \rho_2$, ovvero:

$$\rho(D) = \bigsqcup\{D' \in \mathbb{D} \mid \rho_2(D' \odot X) \sqsubseteq \rho_2(D \odot X)\ \forall X \in \rho_1(\mathbb{D})\} \tag{7.1.1}$$

In generale sarà $D = \sum_{\boldsymbol{G} \in \mathrm{Goals}} S_{\boldsymbol{G}}$ con $S_{\boldsymbol{G}} \in \mathrm{WFS}_{\boldsymbol{G}}$. Supponiamo poi che $S \in \mathrm{WFS}_{\boldsymbol{G}}$ implichi $S \odot X \in \mathrm{WFS}_{\boldsymbol{G}}$ per ogni $X \in \rho_1(\mathbb{D})$. Allora:

$$\rho(D) \qquad =$$

[definizione di ρ]

$$\bigsqcup\{D' \in \mathbb{D} \mid \rho_2(D' \odot X) \sqsubseteq \rho_2(\textstyle\sum_{\boldsymbol{G}\in \text{Goals}} S_{\boldsymbol{G}} \odot X)\ \forall X \in \rho_1(\mathbb{D})\} \quad =$$

[additività a sinistra di $\odot$ e ipotesi di sopra]

$$\bigsqcup\{D' \in \mathbb{D} \mid \rho_2(D' \odot X) \sqsubseteq \textstyle\sum_{\boldsymbol{G}\in \text{Goals}} \rho_2^*(S_{\boldsymbol{G}} \odot X)\ \forall X \in \rho_1(\mathbb{D})\} \quad =$$

[struttura delle collezioni]

$$\bigsqcup\{\textstyle\sum_{\boldsymbol{G}\in \text{Goals}} S'_{\boldsymbol{G}} \mid \forall \boldsymbol{G} \in \text{Goals}.\ S'_{\boldsymbol{G}} \in \text{WFS}_{\boldsymbol{G}},$$
$$\rho_2(\textstyle\sum_{\boldsymbol{G}\in \text{Goals}} S'_{\boldsymbol{G}} \odot X) \sqsubseteq \textstyle\sum_{\boldsymbol{G}\in \text{Goals}} \rho_2^*(S_{\boldsymbol{G}} \odot X)\ \forall X \in \rho_1(\mathbb{D})\} \quad =$$

[additività di $\odot$ e ipotesi di cui sopra]

$$\bigsqcup\{\textstyle\sum_{\boldsymbol{G}\in \text{Goals}} S'_{\boldsymbol{G}} \mid \forall \boldsymbol{G} \in \text{Goals}.\ S'_{\boldsymbol{G}} \in \text{WFS}_{\boldsymbol{G}},$$
$$\textstyle\sum_{\boldsymbol{G}\in \text{Goals}} \rho_2^*(S_{\boldsymbol{G}} \odot X) \sqsubseteq \textstyle\sum_{\boldsymbol{G}\in \text{Goals}} \rho_2^*(S_{\boldsymbol{G}} \odot X)\ \forall X \in \rho_1(\mathbb{D})\} \quad =$$

[teoria degli insiemi]

$$\textstyle\sum_{\boldsymbol{G}\in \text{Goals}} \bigsqcup\{S'_{\boldsymbol{G}} \in \text{WFS}_{\boldsymbol{G}} \mid$$
$$\rho_2^*(S'_{\boldsymbol{G}} \odot X) \sqsubseteq \rho_2^*(S_{\boldsymbol{G}} \odot X)\ \forall X \in \rho_1(\mathbb{D})\}$$

A questo punto, si può porre $\rho^*(S) = \bigsqcup\{S' \in \text{WFS}_{\boldsymbol{G}} \mid \rho_2^*(S' \odot X) \sqsubseteq \rho_2^*(S \odot X)\ \forall X \in \rho_1(\mathbb{D})\}$ e verificare se ρ^* rispetti tutte le condizioni per la definizione di un osservabile.

Teorema 7.1.1 *Siano ρ_1 e ρ_2 due osservabili, $\odot$ un operatore binario su $\mathbb{D}$ additivo a sinistra tale che*

1. *se $S \in WFS_{\boldsymbol{G}}$ allora $S \odot X \in WFS_{\boldsymbol{G}}$ per ogni $X \in \rho_1(\mathbb{D})$,*

2. *$D_1 \odot D_2 = \emptyset$ sse $D_1 = \emptyset$.*

3. *$S \equiv S'$ implica $S \odot X \equiv S' \odot X$ per ogni $X \in \rho_1(\mathbb{D})$.*

Allora $\rho_1 \to^{\odot} \rho_2$ è un osservabile.

Dimostrazione. Dalla condizione 2 si ottiene facilmente:

$$\rho^*(\emptyset) \quad =$$

[definizione di ρ^*]

$$\bigsqcup\{S' \in \text{WFS}_{\boldsymbol{G}} \mid \rho_2^*(S' \odot X) \sqsubseteq \rho_2^*(\emptyset \odot X)\ \forall X \in \rho_1(\mathbb{D})\} \quad =$$

[per la condizione 2]

$$\bigsqcup\{S' \in \mathrm{WFS}_{\boldsymbol{G}} \mid S' = \emptyset\} \qquad =$$
$$\emptyset$$

che verifica una delle proprietà necessarie affinchè ρ sia un osservabile. Bisogna ora dimostrare che ρ^* è un operatore di chiusura. La cosa tuttavia è evidente perché ρ è un uco e ρ^* altro non è che la sua restrizione, evidentemente chiusa, su $\mathrm{WFS}_{\boldsymbol{G}}$.

Infine, sia $S_1 \equiv S_2$, con $S_1 \in \mathrm{WFS}_{\boldsymbol{G}_1}$ e $S_2 \in \mathrm{WFS}_{\boldsymbol{G}_2}$. Abbiamo:

$$\rho^*(S_1) \qquad =$$

[definizione di ρ^*]

$$\bigsqcup\{S' \in \mathrm{WFS}_{\boldsymbol{G}_1} \mid \rho_2^*(S' \odot X) \sqsubseteq \rho_2^*(S_1 \odot X)\ \forall X \in \rho_1(\mathbb{D})\} \qquad =$$

[per il teorema 5.1.3]

$$\bigsqcup\{\boldsymbol{G}_1 \cdot S'' \mid S'' \in \mathrm{WFS}_{\boldsymbol{G}_2}, \rho_2^*((\boldsymbol{G}_1 \cdot S'') \odot X) \sqsubseteq \rho_2^*((\boldsymbol{G}_1 \cdot S_2) \odot X)\ \forall X \in \rho_1(\mathbb{D})\} \qquad =$$

[per le ipotesi 1 e 3 del teorema]

$$\boldsymbol{G}_1 \cdot \bigsqcup\{S'' \in \mathrm{WFS}_{\boldsymbol{G}_2} \mid \boldsymbol{G}_1 \cdot \rho_2^*(S'' \odot X) \sqsubseteq \boldsymbol{G}_1 \cdot \rho_2^*(S_2 \odot X)\ \forall X \in \rho(\mathbb{D})\} \qquad =$$

[proprietà della varianza tra collezioni]

$$\boldsymbol{G}_1 \cdot \bigsqcup\{S'' \in \mathrm{WFS}_{\boldsymbol{G}_2} \mid \rho_2^*(S'' \odot X) \sqsubseteq \rho_2^*(S_2 \odot X)\ \forall X \in \rho(\mathbb{D})\} \qquad \equiv$$

[per il teorema 5.1.3]

$$\rho^*(S_2)$$

e questo completa la nostra dimostrazione che ρ è un osservabile. ■

7.1.1 Autodipendenze funzionali rispetto a $\bowtie$

Tra i vari operatori sulle collezioni che soddisfano le condizioni del teorema 7.1.1, uno particolarmente importante è sicuramente il $\bowtie$ per il fatto di essere utilizzato dalle definizioni semantiche e per le proprietà che, proprio per questo motivo, ci si aspetta dalle sue dipendenze funzionali.

Teorema 7.1.2 *L'operatore $\bowtie$ soddisfa le condizioni del teorema 7.1.1, indipendentemente dalla scelta di ρ_1.*

Dimostrazione. Basta dimostrare le proprietà per $\rho_1 = \lambda x.x$. Tuttavia la prova è banale e discende direttamente dalla definizione di $\bowtie$. ∎

Teorema 7.1.3 *Se ρ un osservabile allora $Dep^{\bowtie} \rho$ è anch'esso un osservabile che gode delle seguenti proprietà:*

1. *$Dep^{\bowtie} \rho$ è un raffinamento di ρ,*
2. *se ρ è un operazionale, allora $Dep^{\bowtie} \rho = \rho$,*
3. *se ρ è un denotazionale, allora $Dep^{\bowtie} \rho$ è il massimo osservabile tra quelli minori di ρ per cui $\bowtie$ è preciso rispetto al primo argomento. Inoltre esso è preciso anche sul secondo argomento.*

Dimostrazione. Esaminiamo separatamente i singoli punti:

1. Diretta conseguenza del teorema 3.6.3, poiché $D \bowtie \emptyset = D$ e $\emptyset \in \rho(\mathbb{D})$ qualunque sia ρ.
2. Diretta conseguenza del teorema 3.6.2.
3. Per il teorema 3.6.4, $\mathrm{Dep}^{\bowtie} \rho$ è maggiore di tutti gli osservabili minori di ρ e per i quali vale la (5.3.1). D'altronde, è possibile dimostrare che $\mathrm{Dep}^{\bowtie} \rho$ è preciso per $\bowtie$ sulla falsariga del corollario 3.6.7 sfruttando il fatto che $(D_1 \bowtie D_2) \bowtie D_3 = D_1 \bowtie ((D_2 \bowtie D_3) + D_3)$ [lemma 4.2.8] e che $+$ è preciso rispetto a qualunque osservabile. Segue banalmente la tesi. ∎

Ad esempio, consideriamo l'osservabile $\rho = \mathrm{Dep}^{\bowtie} \rho_h$. Per quanto abbiamo detto, ρ è il massimo osservabile minore di ρ_h per cui $\bowtie$ è preciso. Allora, dal teorema 6.4.5 otteniamo $\mathrm{Dep}^{\bowtie} = \rho_l$.

7.2 Completamento disgiuntivo

Anche il completamento disgiuntivo può essere applicato al nostro framework semantico, come fatto per le dipendenze funzionali. Al solito, la prima cosa da controllare è se effettivamente gli osservabili sono chiusi rispetto all'operatore $\mho$.

Teorema 7.2.1 *Sia ρ' un'osservabile e $\rho = \mho_{\mathbb{D}}\rho'$. Allora ρ' è un osservabile.*

Dimostrazione. Sia $\rho^* = \rho|_{\mathrm{PuWFS}}$, verifichiamo che esso soddisfa tutte le condizioni perché ρ sia un osservabile.

1. Il fatto che $\rho^*(\emptyset) = \rho(\emptyset) = \emptyset$ deriva banalmente dalla proprietà $\rho'(\emptyset) = \emptyset$.
2. Cominciamo col dimostrare che $\rho^*(S) \in \mathrm{WFS}_{\boldsymbol{G}}$ se $S \in \mathrm{WFS}_{\boldsymbol{G}}$. La cosa è evidente perché $\rho \sqsubseteq \rho'$ per cui per ogni S abbiamo $S \subseteq \rho(S) \subseteq \rho'(S)$. Ma $\rho'(S) \in \mathrm{WFS}_{\boldsymbol{G}}$ da cui la tesi. A questo punto, che ρ^* sia un operatore di chiusura discende direttamente dal fatto che lo è ρ per definizione.
3. se $S \equiv S'$ e segue $\rho(S) \equiv \rho(S')$ perche $\mathrm{WFS}_{\boldsymbol{G}}$ e $\mathrm{WFS}_{\boldsymbol{G}\sigma}$ sono tra loro isomorfi e così anche gli uco su di essi. ∎

Si noti che fare il completamento disgiuntivo di ρ' è lo stesso di farlo limitatamente ai $\rho'|_{\mathrm{WFS}_{\boldsymbol{G}}}$ per ogni $\boldsymbol{G}$ per ottenere ρ^*. Infatti vale il seguente

Teorema 7.2.2 *Sia ρ' un osservabile e $\rho = \mho_{\mathbb{D}}\rho'$. Allora:*

$$\rho|_{WFS_{\boldsymbol{G}}} = \mho_{WFS_{\boldsymbol{G}}}\rho'|_{WFS_{\boldsymbol{G}}}$$

Dimostrazione. Poniamo

$$\eta_{\boldsymbol{G}} = \mho_{\mathrm{WFS}_{\boldsymbol{G}}}(\rho'|_{\mathrm{WFS}_{\boldsymbol{G}}}) \tag{1}$$

$$\eta(D) = \sum_{\boldsymbol{G}\in\mathrm{Goals}} \eta_{\boldsymbol{G}}(D(\boldsymbol{G})) \tag{2}$$

Sappiamo poi che $\rho(D) = \sum_{\boldsymbol{G}\in\mathrm{Goals}} \rho(D(\boldsymbol{G}))$ per cui se dimostriamo che $\eta = \rho$ si ha anche automaticamente l'uguaglianza tra $\rho|_{\mathrm{WFS}_{\boldsymbol{G}}}$ ed $\eta_{\boldsymbol{G}}$.

Si noti allora che η è additivo, dato che ogni collezione D la si può scomporre come somma di elementi di $\mathrm{WFS}_{\boldsymbol{G}}$ distinti e, per ognuno di questi elementi, $\eta_{\boldsymbol{G}}$ risulta additiva per definizione. Dunque è $\rho \sqsubseteq \eta$. D'altronde $\rho|_{\mathrm{WFS}_{\boldsymbol{G}}}$ è additivo, e come tale deve essere maggiore di $\eta_{\boldsymbol{G}}$ che è il completamento disgiuntivo di $\rho'|_{\mathrm{WFS}_{\boldsymbol{G}}}$. Dunque anche $\rho \sqsupseteq \eta$ e da qui la tesi. ∎

Ci viene allora da chiederci come si comportano i nostri operatori sulle collezioni rispetto al completamento disgiuntivo. Dal teorema 3.5.4 segue il seguente

Corollario 7.2.3 *Sia ρ un osservabile che goda di una o più delle proprietà* (4.3.23), (4.3.20), (4.3.21), (4.3.22), (4.3.25). (5.3.1). *Allora anche $\mho_{\mathbb{D}}\rho$ gode delle stesse proprietà.*

Corollario 7.2.4 *Le classi degli osservabili denotazionali, operazionali e perfetti sono chiuse rispetto al completamento disgiuntivo.*

Non ci resta ora che vederne un esempio di applicazione. Abbiamo già parlato dell'osservabile delle risposte calcolte ground, esaminiamo adesso quello delle istanze di risposte calcolate ρ_c (c-semantica) e verifichiamo che ρ_c è il completamento disgiuntivo di ρ_g.

Definiamo intanto il nostro nuovo osservabile. Per ogni $S \in \mathrm{WFS}_{\boldsymbol{G}}$ poniamo

$$\rho_c^*(S) = \{d \in \mathrm{Derivs}_{/\overset{\mathrm{der}}{\approx}} \mid \exists d' \in S \text{ t.c. } \mathrm{first}(d) = \mathrm{first}(d') \wedge d \propto d'\} \tag{7.2.1}$$

con

$$d \propto d' \iff \mathrm{last}(d) \neq \Box \vee \mathrm{answer}(d) \leq \mathrm{answer}(d') \tag{7.2.2}$$

Dunque ρ_c è additivo, come ci si aspettava dal completamento disgiuntivo di ρ_g. Si noti che $\propto$ può essere riscritta come segue:

$$d \propto d' \iff \mathrm{last}(d) \neq \Box \vee \forall \gamma \leq \mathrm{answer}(d).\ \gamma \leq \mathrm{answer}(d') \tag{7.2.3}$$

oppure, per universi di Herbrand non banali, come

$$d \propto d' \iff \mathrm{last}(d) \neq \Box \vee \forall \gamma \leq \mathrm{answer}(d) \text{ ground}.\ \gamma \leq \mathrm{answer}(d') \tag{7.2.4}$$

sfruttando la proprietà che l'anti-istanza meno generale delle istanze ground di un termine t è proprio t stesso. Talvolta si può trovare ρ_c espresso nella forma che segue che si dimostra essere equivalente a quella da noi data.

$$\rho_c^*(S) = \{d \mid \forall \boldsymbol{G}' \leq \boldsymbol{G}\,\mathrm{answer}(d)\ \exists d' \in S \text{ t.c. } \boldsymbol{G}' \leq \boldsymbol{G}\,\mathrm{answer}(d')\} \tag{7.2.5}$$

Bisognerebbe dimostrare che ρ_c è effettivamente un osservabile ma è banale verificare che $\propto$ gode di tutte le proprietà del teorema 5.2.3. A questo punto si dimostra il seguente

Teorema 7.2.5 *Se l'universo di Herbrand è non banale, allora* $\rho_c = \mho_{\mathbb{D}}\rho_g$.

Dimostrazione. Sappiamo già che ρ_c è un osservabile additivo più preciso di ρ_g, dobbiamo solo dimostrare che è anche il più astratto tra questi. Allo scopo basta verificare che ogni elemento della sua immagine può essere ottenuta come somma di elementi delle immagini di ρ_g. Ciò vuol dire che ogni osservabile additivo minore di ρ_g deve anche essere minore di ρ_c da cui la tesi.

Sia allora $D \in \mathbb{D}$, voglio verificare che $\rho_c(D) = \sum_{d \in D} \rho_g(\mathrm{WFS}\{d\})$. Infatti se $d' \in \rho_g(\mathrm{WFS}\{d\})$ allora $d' \in \rho_c(\mathrm{WFS}\{d\}) \sqsubseteq \rho_c(D)$ dato che o $\mathrm{last}(d') \neq \Box$, nel quel caso è banale, o $\mathrm{last}(d') = \Box$ ma a questo punto in $\mathrm{WFS}\{d\}$ sta una sola refutazione per cui, se $\mathrm{first}(d) = \boldsymbol{G}$,

$$\forall \boldsymbol{G}\gamma \leq \boldsymbol{G}\,\mathrm{answer}(d') \text{ ground. } \boldsymbol{G}\gamma \leq \boldsymbol{G}\,\mathrm{answer}(d) \tag{1}$$

e quindi $\boldsymbol{G}\,\mathrm{answer}(d') \leq \boldsymbol{G}\,\mathrm{answer}(d)$. Dall'altro lato, se $d' \in \rho_c(D)$ vuol dire che esiste $d \in D$ per cui $d' \in \rho_g(\mathrm{WFS}\{d\})$ come si desiderava fosse. ∎

Come conseguenza di questo risultato e del teorema 7.2.4 si ha che l'osservabile della c semantica è denotazionale, senza bisogno di ulteriori prove.

Capitolo 8

Conclusioni

Vediamo di tirare le somme, riassumendo brevemente i risultati ottenuti nel corso della tesi. Abbiamo iniziato col dimostrare che, dato un reticolo completo C, il lub e il glb del reticolo degli uco (ovvero delle astrazioni) su C mantengono le proprietà di precisione rispetto ad eventuali operatori concreti, a patto che questi ultimi siano additivi. Lo stesso problema del mantenimento della precisione è stato affrontato, successivamente, in rapporto ad operatori di raffinamento sugli uco quali le dipendenze funzionali e il completamento disgiuntivo. Mentre per quest'ultimo si è ottenuto un risultato generale per tutti gli operatori additivi, nel caso delle dipendenze la precisione dipende da proprietà "ad hoc" degli operatori concreti.

Tutti questi, comunque, sono risultati generali della teoria dell'interpretazione astratta per cui possono essere utilizzati in ambiti completamente diversi dal nostro framework semantico a cui, invece, è dedicato tutto il resto della tesi. Dopo averlo introdotto, seguendo essenzialmente la presentazione data in [CLM95], pur con qualche piccola differenza più che altro a livello di notazioni, abbiamo cominciato ad esaminarne delle estensioni. In particolare abbiamo visto come caratterizzare gli osservabili additivi in funzione di opportune relazioni sull'insieme delle derivazioni e abbiamo introdotto gli osservabili operazionali, caratterizzati da avere una semantica operazionale, definita quindi in funzione di un apposito sistema di transizione, precisa e composizionale (rispetto alla chiamata di goal atomici e all'AND tra goal).

Successivamente abbiamo cominciato ad analizzare pù da vicino il reticolo degli osservabili. Abbiamo visto che operazionali, perfetti e denotazionali ne costituiscono tanti sottoreticoli. Abbiamo invece scoperto che tali classi non sono le uniche a godere delle proprietà di precisione e composizionalità a cui siamo interessati. Più problematico è stato determinare delle maggiorazioni utili per le varie classi di osservabili di cui sopra. La cosa è stata possibile solo dopo esserci ristretti agli osservabili che sono concretizzazioni di quello di successo di Herbrand e solo nel caso dei denotazionali abbiamo trovato una maggiorazione precisa, l'osservabile delle risposte calcolate ground.

Infine, abbiamo applicato al nostro framework quanto abbiamo visto all'inizio sulle dipendenze funzionali e completamento disgiuntivo. Per quest'ultimo si è ottenuto che il completamento di un osservabile perfetto (oppure denotazionale o operazionale), è ancora un osservabile della stessa classe. Abbiamo mostrato come l'osservabile della semantica di Clark sia il più astratto osservabile additivo che concretizzi quello delle risposte calcolate ground.

8.1 Possibili sviluppi

Tutto ciò lascia comunque irrisolti una buona quantità di problemi e propone spunti per ulteriori ricerche. Nell'ambito della interpretazione astratta pura si può continuare a studiare il problema della mantenimento della precisione rispetto ad altri operatori sugli uco noti in letteratura quali, ad esempio, la complementazione [CFG+95], il completamento di Moore [GR95], ecc ....

Anche relativamente al nostro framework molti sono i punti che si potrebbero approfondire. Ad esempio, potrebbe essere interessante classificare gli osservabili in maniera più fine di quanto fatto finora, considerando singolarmente le proprietà di precisione rispetto agli operatori $\cdot$, $\times$ e $\bowtie$, per vedere l'impatto che esse hanno, una ad una, sulle proprietà di composizionalità della semantica risultante. Ciò che ci si aspetta è che, in qualche modo, siano legate alla composizionalità rispetto alla chiamata di goal atomici, alla AND e alla OR composizionalità rispettivamente.

Attualmente, inoltre, le nostre astrazioni mantengono separate le osservazioni per alberi-SLD con differente goal iniziale. Si potrebbe estendere la definizione di osservabile eliminando questa restrizione, in modo da astrarre anche rispetto ad esso. Ciò, tra l'altro, ci consente anche di introdurre il concetto di *programma astratto* in maniera più elegante di quanto sarebbe possibile con lo schema attuale. L'interpretazione astratta di un programma (concreto) sarebbe allora ottenibile astraendo il programma e calcolandone la sua semantica sul dominio astratto.

È poi possibile passare ad analizzare le proprietà di quegli osservabili che, non essendo precisi rispetto agli operatori semantici, non sono adatti alla definizione di una semantica "standard", ma vengono usati per analisi statiche di programmi. Su di essi è possibile studiare gli effetti degli operatori sugli uco e delle operazioni di lub e glb del reticolo degli osservabili.

Infine, cosa forse più immediata, si può cercare di trovare l'osservabile perfetto più astratto tra le concretizzazioni dell'osservabile di successo ρ_h. Per ottenere qualcosa di elegante, che sia in qualche modo legato ai risultanti con risposte ground, come sarebbe auspicabile, potrebbe essere necessario modificare leggermente, nel framework, la definizione dell'operatore $\bowtie$, magari istanziando automaticamente, quando necessario, le derivazioni del secondo argomento visto che, ora come ora, l'osservabile dei risultanti con risposte ground non è perfetto proprio per questo motivo.

Appendice A

Dimostrazioni

Raccogliamo qui alcune dimostrazioni tecniche sulla teoria dell'unificazione e sulle derivazioni SLD che usiamo nel corso della tesi.

A.1 Teoria dell'unificazione

Lemma A.1.1 *Sia t un termine e θ_1, θ_2 sostituzioni. Se $t\theta_1 \equiv t\theta_2$ segue $\theta_1|_t \equiv \theta_2|_t$.*

Dimostrazione. Se esistesse $x \in \mathrm{var}(t)$ tale che $\theta_1(x) \not\equiv \theta_2(x)$, i termini $t\theta_1$ e $t\theta_2$ non potrebbero essere l'un variante dell'altro nel punto in cui tale variabile appare. Ne segue per assurdo la tesi. ∎

Lemma A.1.2 *Siano γ e θ due sostituzioni idempotenti, t un termine tale che $t\theta \leq t\gamma$ e $\mathrm{var}(\gamma) \cap \mathrm{var}(\theta) \subseteq \mathrm{var}(t)$. Allora esiste $\eta \in \mathrm{mgu}(t\theta, t\gamma)$ tale che $t\theta\eta = t\theta$.*

Dimostrazione. Poichè $t\theta \leq t\gamma$ esiste η tale che $t\gamma\eta = t\theta$ con $\mathrm{dom}(\eta) \subseteq \mathrm{var}(t\gamma)$. Se dimostrassimo che η è idempotente si avrebbe automaticamente $t\theta = t\gamma\eta = t\gamma\eta\eta = t\theta\eta$ e η sarebbe anche un unificatore per i due termini. Inoltre, dato un altro unificatore α di $t\theta$ e $t\gamma$, abbiamo che $t\theta\alpha = t\gamma\eta\alpha = t\gamma\alpha$ e quindi $\eta\alpha|_{t\gamma} = \alpha|_{t\gamma}$. Analogamente per $\alpha|_{t\theta}$. Se poi x è una variabile che non fa parte di $\mathrm{var}(t\gamma) \cup \mathrm{var}(t\theta)$, allora η non contiene legami per essa per cui $x\alpha = x\eta\alpha$. Segue $\alpha = \eta\alpha$ e quindi η è un mgu.

Sia data allora una variabile $x \in \mathrm{var}(t) \cup \mathrm{var}(\theta) \cup \mathrm{var}(\gamma)$, dimostriamo che x può appartenere a $\mathrm{dom}(\eta)$ o a $\mathrm{range}(\eta)$ ma mai ad entrambi.

- $x \in \mathrm{var}(t\gamma) \setminus \mathrm{var}(t)$. Allora $x \in \mathrm{range}(\gamma)$ e, per le ipotesi del teorema, $x \notin \mathrm{var}(\theta)$. Ne segue che x non può far parte di $\mathrm{range}(\eta)$, altrimenti esisterebbe una variabile $v \in \mathrm{var}(t)$ per cui $x \in \mathrm{var}(\eta(\gamma(v)) = \mathrm{var}(\theta(v))$ che è un assurdo.

- $x \in \mathrm{var}(t\theta) \setminus \mathrm{var}(t)$. Allora $x \in \mathrm{range}(\theta)$ e per ipotesi $x \notin \mathrm{var}(\gamma)$. Ne segue $x \notin t\gamma$, ovvero $x \notin \mathrm{dom}(\eta)$.

- $x \in \mathrm{var}(t)$. Distinguiamo vari sottocasi:
  - $x \in \mathrm{dom}(\theta) \setminus \mathrm{dom}(\gamma)$. Deve essere necessariamente $\eta(x) = \theta(x)$. Poichè θ è idempotente, $x \notin \mathrm{range}(\theta)$, quindi $x \notin \mathrm{var}(t\theta)$. Se fosse $x \in \mathrm{range}(\eta)$ si avrebbero due possibilità: o $x \in \mathrm{range}(\gamma\eta) = \mathrm{range}(\theta)$ o $\gamma(x) = y$ per qualche variabile y e $\eta(y) = x$. La prima è chiaramente assurda per le nostre ipotesi e lo stesso vale per la seconda dato che comporta $\theta(x) = x$ mentre $x \in \mathrm{dom}(\theta)$.
  - $x \in \mathrm{dom}(\gamma)$. Poichè γ è idempotente, segue $x \notin \mathrm{range}(\gamma)$ ovvero $x \notin \mathrm{var}(t\gamma)$ per cui $x \notin \mathrm{dom}(\eta)$
  - $x \notin \mathrm{dom}(\gamma) \cup \mathrm{dom}(\theta)$. Deve essere $\eta(x) = x$ per non creare legami per x che non troverebbero riscontro in θ, quindi x non appartiene nè a $\mathrm{dom}(\eta)$ nè a $\mathrm{range}(\eta)$. ■

Lemma A.1.3 *Sia t un termine, θ, θ' e γ sostituzioni idempotenti tali che $\mathrm{var}(\theta) \cap \mathrm{var}(\theta') \subseteq \mathrm{var}(t)$, $t\theta\gamma$ è istanza di $t\theta'$. Allora, se $\sigma \in \mathrm{mgu}(t\theta, t\theta')$, si ha $t\theta\gamma \leq t\theta'\sigma$.*

Dimostrazione. Basta dimostrare che esiste un unificatore β di $t\theta$ e $t\theta'$ per cui $t\theta\beta = t\theta\gamma$. Sia η la sostituzione per cui $t\theta\gamma = t\theta'\eta$. Le variabili $v \in \mathrm{var}(t\theta) \cap \mathrm{var}(t\theta')$ sono variabili di t. Infatti o $v\theta = v$ (oppure $v\theta' = v$), o $v \in \mathrm{range}(\theta) \cup \mathrm{range}(\theta')$ e, per le ipotesi del teorema, $v \in \mathrm{var}(t)$. Ma allora, poichè $v \in \mathrm{var}(t)$ e $v \in \mathrm{var}(t\theta)$, è ovvio che $v \notin \mathrm{dom}(\theta)$, dato che θ è idempotente.

Analogamente si ottiene $v \notin \mathrm{dom}(\theta')$. Ne segue che $v\gamma = v\theta\gamma = v\theta'\eta = v\eta$. Poniamo allora:

$$\beta(x) = \begin{cases} \gamma(x) = \eta(x) & \text{se } x \in \mathrm{var}(t\theta) \cap \mathrm{var}(t\theta') \\ \gamma(x) & \text{se } x \in \mathrm{var}(t\theta) \setminus \mathrm{var}(t\theta') \\ \eta(x) & \text{se } x \in \mathrm{var}(t\theta') \setminus \mathrm{var}(t\theta) \end{cases}$$

Si vede facilmente che β è un unificatore per $t\theta$ e $t\theta'$ e che $t\theta\beta = t\theta\gamma$. ■

Lemma A.1.4 *Siano t_1, t_2, t'_1 e t'_2 termini con $t'_1 \leq t_1\theta_1$, $t'_2 \leq t_2\theta_2$, $(t'_1, t'_2) \leq (t_1, t_2)$, $\mathrm{var}(\theta_1) \cap \mathrm{var}(\theta_2) \subseteq \mathrm{var}(t_1) \cap \mathrm{var}(t_2)$, θ_1 e θ_2 idempotenti. Allora $(t'_1, t'_2) \leq (t_1\theta_1, t_2\theta_2)$.*

Dimostrazione. Sia $(t'_1, t'_2) = (t_1, t_2)\eta$, $t'_1 = t_1\theta_1\eta_1$, $t'_2 = t_2\theta_2\eta_2$. Si noti che le uniche variabili in comune tra $t_1\theta_1$ e $t_2\theta_2$ sono anche variabili di t_1 e t_2, che sicuramente non fanno parte nè di $\mathrm{dom}(\theta_1)$ nè di $\mathrm{dom}(\theta_2)$ per cui $\theta_1\eta_1$ e $\theta_2\eta_2$ coincidono su di esse con η_1 ed η_2 rispettivamente e dunque con η.

A questo punto ponendo

$$\beta(x) = \begin{cases} \eta(x) & \text{se } x \in \mathrm{var}(t_1\theta_1) \cap \mathrm{var}(t_2\theta_2) \\ \eta_1(x) & \text{se } x \in \mathrm{var}(t_1\theta_1) \\ \eta_2(x) & \text{se } x \in \mathrm{var}(t_2\theta_2) \end{cases}$$

si ha banalmente $(t'_1, t'_2) = (t_1\theta_1, t_2\theta_2)\beta$. ■

A.2 Derivazioni e risultanti

Lemma A.2.1 *Siano $d = \boldsymbol{G}_0 \xrightarrow[cs]{\theta}{}^* \boldsymbol{G}_n \in Derivs$ e $d' \stackrel{der}{\approx} d$. Allora $d' = \boldsymbol{G}_0 \xrightarrow[cs']{\theta\sigma}{}^* \boldsymbol{G}_n\sigma$ dove σ è un renaming e le clausole in cs' sono varianti di quelle in cs.*

Dimostrazione. Se $d \stackrel{\mathrm{der}}{\approx} d'$ allora d e d' sono due derivazioni SLD ottenute partendo dallo stesso goal iniziale con $\mathrm{clauses}(d) \equiv \mathrm{clauses}(d')$. Il risultato si ottiene direttamente grazie al lemma delle varianti [Apt90]. ■

Teorema A.2.2 *Sia $d = \boldsymbol{G}_0 \xrightarrow[cs]{\theta}{}^* \boldsymbol{G}_n \in Derivs$ e σ un renaming con $\mathrm{var}(\boldsymbol{G}_0\sigma) \cap \mathrm{var}(cs) = \emptyset$. Allora $\partial_\sigma(d) \stackrel{der}{\approx} \boldsymbol{G}'_0 \xrightarrow[cs]{\sigma^{-1}\theta\sigma}{}^* \boldsymbol{G}'_n$, con $\boldsymbol{G}'_i = \boldsymbol{G}_i\sigma$ per ogni i.*

Dimostrazione. Per definizione $\partial_\sigma(d)$ è una derivazione con goal iniziale $\boldsymbol{G}_0\sigma$ e clauses$(\partial_\sigma(d))$ è prefisso di clauses(d). Ponendo $\partial_\sigma(d) = \boldsymbol{G}'_0 \xrightarrow[c_1]{\theta'_1} \dots \xrightarrow[c_m]{\theta'_n} \boldsymbol{G}'_m$, dimostriamo per induzione che:

1. per ogni $1 \leq i \leq n$, se $\boldsymbol{G}'_{i-1}$ esiste, si può estendere $\partial_\sigma(d)$ almeno fino al goal $\boldsymbol{G}'_i$,

2. per ogni $0 \leq i \leq n$, $\boldsymbol{G}'_i = \boldsymbol{G}_i\sigma$,

3. per ogni $1 \leq i \leq n$, $\theta'_i = \sigma^{-1}\theta_i\sigma$

Il passo base ($i = 0$) è banale dato che $\boldsymbol{G}'_0 = \boldsymbol{G}_0\sigma$ per definizione. Supponiamo ora che $\boldsymbol{G}'_{i-1} = \boldsymbol{G}_{i-1}\sigma$ e sia $c_i = H_i \leftarrow \boldsymbol{B}_i$, $\boldsymbol{G}'_{i-1} = (A'_{i-1}, \boldsymbol{G}')$, $\boldsymbol{G}_{i-1} = (A_{i-1}, \boldsymbol{G})$. Allora $\theta'_i \in \text{mgu}(A'_{i-1}, H_i) = \text{mgu}(A_{i-1}\sigma, H_i)$ mentre $\theta_i \in \text{mgu}(A_{i-1}, H_i)$. Si noti che poichè le clausole sono renamed apart, si ha $H_i = H_i\sigma$. Allora, applicando il corollario A.1.8 di [Meo96] si ricava $\theta'_i = \sigma^{-1}\theta_i\sigma$. Dunque, è possibile estendere la $\partial_\gamma(d)$ aggiungendo il goal $\boldsymbol{G}'_i = (\boldsymbol{B}_i, \boldsymbol{G}')\theta'_i = (\boldsymbol{B}_i, \boldsymbol{G}')\sigma^{-1}\theta_i\sigma = (\boldsymbol{B}_i, \boldsymbol{G})\theta_i\sigma = \boldsymbol{G}_i\sigma$. La tesi segue direttamente. ∎

Lemma A.2.3 *Sia* $d = \boldsymbol{G}_0 \xrightarrow[cs]{\theta}{}^* \boldsymbol{G}_n$ *e* $\gamma \in Subst$ *tale che* $\boldsymbol{G}_0\gamma$ *e* $\boldsymbol{G}_0\theta$ *siano unificabili con mgu* σ *e* $\text{var}(\boldsymbol{G}_0\gamma) \cap \text{var}(cs) = \emptyset$. *Allora:*

1. $\partial_\gamma(d) \overset{der}{\approx} \boldsymbol{G}\gamma \xrightarrow[cs]{\theta'}{}^* \boldsymbol{G}_n\sigma$, *con* $\boldsymbol{G}_0\gamma\theta' = \boldsymbol{G}_0\theta\sigma$.

2. *se* $\boldsymbol{G}\theta \leq \boldsymbol{G}\gamma$, *allora* $\partial_\gamma(d) \overset{der}{\approx} \boldsymbol{G}\gamma \xrightarrow[cs]{\theta'}{}^* \boldsymbol{G}_n$ *con* $\boldsymbol{G}_0\gamma\theta' = \boldsymbol{G}_0\theta$.

Dimostrazione. Dallo *strong lifting lemma* (Lemma 5.1.2 di [Meo96]) segue immeditamente il punto 1. Inoltre, se $\boldsymbol{G}_0\theta \leq \boldsymbol{G}_0\gamma$, per il lemma A.1.2, esiste $\sigma \in \text{mgu}(\boldsymbol{G}_0\gamma, \boldsymbol{G}_0\theta)$ tale che $\boldsymbol{G}_0\theta\sigma = \boldsymbol{G}_0\theta$. D'altronde, riguardando la dimostrazione, si nota che le variabili in $\text{dom}(\sigma)$ sono un sottoinsieme di $\text{var}(\boldsymbol{G}_0\gamma)$. Sia allora $x \in \text{dom}(\sigma)$, distinguiamo i casi:

- $x \in \text{var}(\boldsymbol{G}_0\gamma) \setminus \text{var}(\boldsymbol{G}_0)$. Per le ipotesi di renaming apart del teorema, $x \notin \text{var}(\boldsymbol{G}_n)$,

- $x \in \mathrm{var}(\boldsymbol{G}_0) \cap \mathrm{dom}(\theta)$. Poichè θ è idempotente, segue $x \notin \mathrm{range}(\theta)$ da cui $x \notin \mathrm{var}(\boldsymbol{G}_n)$,

Allora $\mathrm{dom}(\sigma)$ e $\mathrm{var}(\boldsymbol{G}_n)$ sono disgiunti, per cui $\boldsymbol{G}_n\sigma = \boldsymbol{G}_n$. Il punto 2 segue poi banalmente. ■

Lemma A.2.4 *Siano* $d_1 = \boldsymbol{G}_1 \xrightarrow{\theta_1}{}^* \boldsymbol{B}_1$ *e* $d_2 = \boldsymbol{B}_1\sigma \xrightarrow{\theta_2}{}^* \boldsymbol{B}_2$ *per un opportuno renaming* σ. *Allora* $d_1 :: d_2 \overset{der}{\approx} \boldsymbol{G}_1 \xrightarrow{\theta'}{}^* \boldsymbol{B}_2\eta$, *dove* $\theta' = \theta_1\sigma\theta_2\eta$ *ed* η *è un renaming tale che* $\mathrm{var}(\sigma\theta_2\eta) \cap \mathrm{var}(\theta_1) \subseteq \mathrm{var}(\boldsymbol{B}_1)$.

Dimostrazione. Sia d_2' una versione rinominata di d_2 con $\mathrm{first}(d_2') = \boldsymbol{B}_1$ e $\mathrm{var}(d_1) \cap \mathrm{var}(d_2') = \mathrm{var}(\mathrm{first}(d_2'))$. Poiche $d_1 :: d_2$ non è altro che la concatenazione di d_1 e d_2', dal lemma A.2.2 si ha la tesi. ■

Lemma A.2.5 *Siano* $d_1 = \boldsymbol{G}_1 \xrightarrow{\theta_1}{}^* \boldsymbol{B}_1$ *e* $d_2 = \boldsymbol{G}_2 \xrightarrow{\theta_2}{}^* \boldsymbol{B}_2$ *con* $\mathrm{var}(d_1) \cap \mathrm{var}(d_2) \subseteq \mathrm{var}(\boldsymbol{G}_1) \cap \mathrm{var}(\boldsymbol{G}_2)$. *Allora:*

1. *se* $\boldsymbol{B}_1 \neq \Box$, $d_1 :: d_2 = \boldsymbol{G} \xrightarrow{\theta_1}{}^* \boldsymbol{B}_1, \boldsymbol{G}_2\theta_1$,

2. *se* $\boldsymbol{B}_1 = \Box$ *e* $\boldsymbol{G}_2\theta_1$ *è unificabile con* $\boldsymbol{G}_2\theta_2$ *con mgu* σ, *allora* $d_1 \wedge d_2 \overset{der}{\approx} \boldsymbol{G}_1, \boldsymbol{G}_2 \xrightarrow{\theta_1\theta_2'}{}^* \boldsymbol{B}_2\theta_2\sigma$, *dove* $\boldsymbol{G}_2\theta_1\theta_2' = \boldsymbol{G}_2\theta_2\sigma$. *Se poi* $\boldsymbol{G}_2\theta_2 \leq \boldsymbol{G}_2\theta_1$, *allora* $\boldsymbol{G}_2\theta_2\sigma = \boldsymbol{G}_2\theta_2$.

Dimostrazione. Il punto 1 segue direttamente dalla definizione di $\wedge$. Quanto al punto 2, è conseguenza della stessa definizione e del lemma A.2.3. ■

Teorema A.2.6 *Siano* $d_1, d_2, d_1', d_2' \in$ *Derivs, tali che* d_1 *e* d_1', *così come* d_2 *e* d_2' *abbiano stesso risultante modulo varianza. Allora:*

- *per ogni* $\gamma \in$ *Subst tale che* $\mathrm{first}(d_1)\gamma$ *e* $\mathrm{first}(d_1)\,\mathrm{answer}(d_1)$ *sono unificabili, si ha* $\mathrm{result}(\partial_\gamma(d_1)) \equiv \mathrm{result}(\partial_\gamma(d_1'))$,

- $\mathrm{result}(d_1 \wedge d_2) \equiv \mathrm{result}(d_1' \wedge d_2')$,

- *se* $\mathrm{first}(d_2) \equiv \mathrm{last}(d_1)$ *allora* $\mathrm{result}(d_1 :: d_2) \equiv \mathrm{result}(d_1' :: d_2')$.

Analogamente per la funzione length.

Dimostrazione. Dai lemmi A.2.3, A.2.5 e A.2.4 si nota che il risultante di una composizione di derivazioni, sotto le ipotesi del teorema e modulo varianza, dipende unicamente dal risultante delle derivazioni di partenza, da cui la tesi.

■

Bibliografia

[Apt90] K. R. Apt. Introduction to Logic Programming. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B: Formal Models and Semantics, pages 495–574. Elsevier, Amsterdam and The MIT Press, Cambridge, 1990.

[BGLM93] A. Bossi, M. Gabbrielli, G. Levi, and M. C. Meo. An OR-compositional Semantics for Logic Programs. In J. M. Jacquet, editor, *Constructing Logic Programs*, pages 215–240. John Wiley & Sons Ltd, 1993.

[BGLM94] A. Bossi, M. Gabbrielli, G. Levi, and M. C. Meo. A Compositional Semantics for Logic Programs. *Theoretical Computer Science*, 122(1-2):3–47, 1994.

[Bir67] G. Birkhoff. *Lattiche Theory*, volume XXV of *AMS Colloquium Publications*. AMS, third edition, 1967.

[BLM91] A. Brogi, E. Lamma, and P. Mello. Compositional Model-theoretic Semantics for Logic Programs. Technical Report 8-91, Comp. Sci. Dept. University of Pisa, 1991.

[CC77] P. Cousot and R. Cousot. Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs by Construction or Approximation of Fixpoints. In *Proc. Fourth ACM Symp. Principles of Programming Languages*, pages 238–252, 1977.

[CC79] P. Cousot and R. Cousot. Systematic Design of Program Analysis Frameworks. In *Proc. Sixth ACM Symp. Principles of Programming Languages*, pages 269–282, 1979.

[CC92a] P. Cousot and R. Cousot. Abstract Interpretation and Applications to Logic Programs. *Journal of Logic Programming*, 13(2 & 3):103–179, 1992.

[CC92b] P. Cousot and R. Cousot. Abstract Interpretation Frameworks. *Journal of Logic and Computation*, 2(4):511–549, 1992.

[CFG+95] A. Cortesi, G. Filè, R. Giacobazzi, C. Palamidessi, and F. Ranzato. Complementation in abstract interpretation. In A. Mycroft, editor, *Proc. of Int. Static Analysis Symposium, (SAS'95)*, volume 983 of *Lecture Notes in Computer Science*, pages 100–117, Glashow, UK, 1995. Springer-Verlag, Berlin.

[CL94] M. Comini and G. Levi. An algebraic theory of observables. In M. Bruynooghe, editor, *Proceedings of the 1994 Int'l Symposium on Logic Programming*, pages 172–186. The MIT Press, Cambridge, Mass., 1994.

[Cla79] K. L. Clark. Predicate logic as a computational formalism. Res. Report DOC 79/59, Imperial College, Dept. of Computing, London, 1979.

[CLM95] M. Comini, G. Levi, and M. C. Meo. Compositionality of *SLD*-derivations and their abstractions. In J. Lloyd, editor, *Proceedings of the 1995 Int'l Symposium on Logic Programming*, pages 561–575. The MIT Press, Cambridge, Mass., 1995.

[DF87] P. Deransart and G. Ferrand. Programmation en logique avec negation: presentation formelle. Technical Report No. 87/3, Lab. d'Informatique, Département de Mathématiques et d'Informatique, Université d'Orléans, 1987.

[Fer87] G. Ferrand. Error Diagnosis in Logic Programming, an Adaptation of E. Y. Shapiro's Method. *Journal of Logic Programming*, 4:177–198, 1987.

[FLMP88] M. Falaschi, G. Levi, M. Martelli, and C. Palamidessi. A new Declarative Semantics for Logic Languages. In R. A. Kowalski and K. A. Bowen, editors, *Proc. Fifth Int'l Conf. on Logic Programming*, pages 993–1005. The MIT Press, Cambridge, Mass., 1988.

[FLMP89] M. Falaschi, G. Levi, M. Martelli, and C. Palamidessi. Declarative Modeling of the Operational Behavior of Logic Languages. *Theoretical Computer Science*, 69(3):289–318, 1989.

[FR95] G. Filé and F. Ranzato. The powerset operator on abstract interpretation. Technical report, Dipartimento di Matematica Pura ed Applicata, Università di Padova, 1995.

[Gab92] M. Gabbrielli. *The Semantics of Logic Programming as a Programming Language*. PhD thesis, Dipartimento di Informatica, Università di Pisa, 1992.

[GG94] M. Gabbrielli and R. Giacobazzi. Goal independency and call patterns in the analysis of logic programs. In *Proc. SAC'94*, 1994.

[Gia92] R. Giacobazzi. *Semantic Aspects of Logic Program Analysis*. PhD thesis, Dipartimento di Informatica, Università di Pisa, 1992.

[Gia96] R. Giacobazzi. "optimal" collecting semantics for analysis in a hierarchy of logic program semantics. In C. Puech and R. Reischuk, editors, *Proc. 13th International Symposium on Theoretical Aspects of Computer Science (STACS'96)*, volume 1046 of *Lecture Notes in Computer Science*, pages 503–514. Springer-Verlag, 1996.

[GLM92] M. Gabbrielli, G. Levi, and M. C. Meo. Observational Equivalences for Logic Programs. In K. Apt, editor, *Proc. Joint Int'l Conf. and Symposium on Logic Programming*, pages 131–145. The MIT Press, Cambridge, Mass., 1992.

[GLM95] M. Gabbrielli, G. Levi, and M. C. Meo. Resultants semantics for PROLOG. *Journal of Logic and Computation*, 1995. To appear.

[GM92] M. Gabbrielli and M. C. Meo. Fixpoint Semantics for Partial Computed Answer Substitutions and Call Patterns. In H. Kirchner and G. Levi, editors, *Algebraic and Logic Programming, Proceedings of the Third International Conference*, volume 632 of *Lecture Notes in Computer Science*, pages 84–99. Springer-Verlag, Berlin, 1992.

[GPR96] R. Giacobazzi, C. Palamidessi, and F. Ranzato. Weak relative pseudo-complements of closure operators. *Algebra Universalis*, 1996. Also available as Technical Report LIX/95/04, LIX, Ecole Polytechnique, 91128 Palaiseau Cedex, France.

[GR95] R. Giacobazzi and F. Ranzato. Functional dependencies and moore-set completions of abstract interpretations and semantics. In J. Lloyd, editor, *Proc. 1995 Int'l Symposium on Logic Programming (ILPS'95)*, pages 321–335, Portland, OR, 1995. The MIT Press.

[GR96] R. Giacobazzi and F. Ranzato. Compositional optimization of disjunctive abstract interpretations. In H.R. Nielson, editor, *Proc. of the 6th European Symposium on Programming (ESOP '96)*, volume 1058 of *Lecture Notes in Computer Science*, pages 141–155, Linkoping, SE, 1996. Springer-Verlag, Berlin.

[GS89] H. Gaifman and E. Shapiro. Fully abstract compositional semantics for logic programs. In *Proc. Sixteenth Annual ACM Symp. on Principles of Programming Languages*, pages 134–142. ACM, 1989.

[Llo87] J. W. Lloyd. *Foundations of Logic Programming*. Springer-Verlag, Berlin, 1987. Second edition.

[LMM88] J. L. Lassez, M. J. Maher, and K. Marriott. Unification Revisited. In J. Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 587–625. Morgan Kaufmann, Los Altos, Ca., 1988.

[Mah88] M. J. Maher. Equivalences of Logic Programs. In J. Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 627–658. Morgan Kaufmann, Los Altos, Ca., 1988.

[Meo96] M.C. Meo. *A framework for reasoning about the Semantics of Logic Program*. PhD thesis, Dipartimento di Informatica, Università di Pisa, 1996.

[MP88] P. Mancarella and D. Pedreschi. An Algebra of Logic Programs. In R. A. Kowalski and K. A. Bowen, editors, *Proc. Fifth Int'l Conf. on Logic Programming*, pages 1006–1023. The MIT Press, Cambridge, Mass., 1988.

[O'K85] R. A. O'Keefe. Towards an Algebra for Constructing Logic Programs. In *Proc. IEEE Symp. on Logic Programming*, pages 152–160, 1985.

[vEK76] M. H. van Emden and R. A. Kowalski. The semantics of predicate logic as a programming language. *Journal of the ACM*, 23(4):733–742, 1976.

[War42] Morgan Ward. The closure operators of a lattice. *Annals of Mathematics*, 43(2):191–196, 1942.

[YYD87] S. Yamasaki, M. Yoshida, and S. Doshita. A fixpoint semantics of Horn Sentences based on Substitution Sets. *Theoretical Computer Science*, 51:309–324, 1987.