Bollettino
Unione Matematica Italiana

Sezione A – La Matematica nella Società e nella Cultura

Francesca Scozzari

Teoria dei domini nell'interpretazione astratta: equazioni, completezza e logica

Bollettino dell'Unione Matematica Italiana, Serie 8, Vol. **3-A**—*La Matematica nella Società e nella Cultura* (2000), n.1S, p. 213–216.

Unione Matematica Italiana

<http://www.bdim.eu/item?id=BUMI_2000_8_3A_1S_213_0>

Bollettino dell'Unione Matematica Italiana, Unione Matematica Italiana, 2000.

Bollettino U. M. I.
La Matematica nella Società e nella Cultura
Serie VIII, Vol. III-A, Supplemento ad Aprile 2000, 213-216

Teoria dei domini nell'interpretazione astratta: equazioni, completezza e logica.

Francesca Scozzari

1. – Interpretazione astratta.

La tesi si occupa della costruzione sistematica di domini astratti nell'ambito della teoria dell'*interpretazione astratta*. L'interpretazione astratta – formulata da Patrick e Radhia Cousot in [1, 2] – è una teoria generale per descrivere il comportamento di sistemi dinamici discreti a differenti livelli di astrazione ed è stata applicata in molte aree dell'informatica. Tipici esempi sono la descrizione di semantiche non-standard di linguaggi di programmazione e l'analisi statica di programmi. L'interpretazione astratta nasce dall'idea che l'analisi di un sistema (ad esempio un programma) può essere ottenuta come una approssimazione della sua semantica formale. Quest'ultima viene specificata da un insieme di oggetti C (detto dominio concreto) e da una funzione semantica $S : \mathit{Programmi} \to C$ (detta semantica concreta) la quale associa ad ogni programma P la sua semantica $S(P) \in C$. Un'interpretazione astratta si ottiene sostituendo il dominio concreto C e la semantica concreta S con, rispettivamente, un dominio astratto A ed una semantica astratta $S^\sharp$: $\mathit{Programmi} \to A$ che approssimi in modo corretto la semantica concreta, cioè, per ogni programma P, $S^\sharp(P)$ rappresenti qualche oggetto concreto $c \in C$ tale che $S(P) \leqslant c$. Una delle caratteristiche fondamentali dell'interpretazione astratta è che molte proprietà dell'analisi – ad esempio la precisione, la completezza e la composizionalità – dipendono esclusivamente dalla nozione di astrazione, cioè dal dominio A prescelto. I domini astratti assumono quindi un ruolo centrale nella progettazione di interpretazioni astratte.

1.1. – *Domini astratti.*

Un dominio astratto è un insieme di oggetti matematici che rappresenta le proprietà del sistema dinamico alle quali siamo interessati. La nozione di approssimazione è specificata tramite opportuni ordinamenti parziali, e si assume che entrambi i domini C ed A siano reticoli completi rispetto a questo ordine. Un oggetto astratto $a \in A$ approssima $c \in C$ se a rappresenta un qualche oggetto concreto $\tilde{c} \in C$ tale che $c \leqslant \tilde{c}$. Un insieme A di oggetti astratti è un dominio astratto (fissato un qualche dominio concreto C) se, per ogni oggetto concreto $c \in C$, esiste in A la migliore approssimazione di c o, equivalentemente, se è possibile definire una funzione di astrazione $\alpha : C \to A$ che associ ad ogni valore concreto c la sua migliore approssimazione $\alpha(c) \in A$ tale che, per ogni altro oggetto astratto $b \in A$ che approssimi c, valga $\alpha(c) \leqslant b$. Ogni dominio astratto è isomorfo ad un sottoinsieme del dominio concreto C e l'insieme dei domini astratti di un dominio concreto C è isomorfo all'insieme degli operatori di chiusura superiori sull'insieme C.

Nel seguito, ci limiteremo quindi a considerare domini astratti sottoinsiemi del dominio concreto. Denotiamo con $uco(C)$ l'insieme di tutti i domini astratti su C e dato un dominio astratto $A \in uco(C)$, denotiamo con α_A la funzione di astrazione associata. Inoltre, diciamo che un dominio astratto A è più concreto di B (o equivalentemente che B è più astratto di A) se $B \subseteq A$.

1.2. – *Funzioni astratte: correttezza e completezza.*

Dato un dominio concreto C ed una funzione concreta $f: C \to C$ monotona, un'interpretazione astratta è una coppia $\langle A, f^\sharp \rangle$ dove A è un dominio astratto e $f^\sharp: A \to A$ è una funzione astratta che approssimi correttamente la semantica concreta, cioè $f \leqslant f^\sharp \circ \alpha_A$. Per ogni dominio astratto A, esiste sempre la migliore approssimazione corretta della funzione concreta f, che è esattamente $\alpha_A \circ f$, cioè $\lambda x.\ \alpha_A(f(x))$. Ne consegue che molte proprietà interessanti di una interpretazione astratta (ad esempio concernenti la precisione) dipendono unicamente dalla scelta del dominio astratto. Considerare interpretazioni astratte del tipo $\langle A, \alpha_A \circ f \rangle$ (dove la funzione astratta è la migliore approssimazione della funzione concreta) non è però sufficiente per ottenere interpretazioni astratte complete (dove il risultato di una computazione astratta sia la migliore approssimazione in A del risultato della corrispondente computazione concreta). Formalmente, un'interpretazione astratta $\langle A, f^\sharp \rangle$ è *completa* per la funzione concreta f se, per ogni $c \in C$, il risultato di una computazione astratta $f^\sharp(\alpha_A(c))$ è la migliore approssimazione in A del risultato della corrispondente computazione concreta $f(c)$, cioè $\alpha_A(f(c)) = f^\sharp(\alpha_A(c))$. Una caratteristica fondamentale della completezza nell'interpretazione astratta è che dipende unicamente dalla funzione di astrazione. Se $\langle A, f^\sharp \rangle$ è completa, $\langle A, \alpha_A \circ f \rangle$ è ancora completa, e inoltre $f^\sharp$ coincide con $\alpha_A \circ f$.

Quindi, dato un dominio astratto A, $\langle A, f^\sharp \rangle$ è completa se e solo se $\langle A, \alpha_A \circ f \rangle$ è completa e $f^\sharp = \alpha_A \circ f$. Questa semplice osservazione implica che la completezza è una proprietà dei domini astratti, cioè una caratteristica del dominio indipendente dalla funzione semantica astratta. Nel seguito, diremo che un dominio astratto A è completo per una funzione concreta f se l'interpretazione astratta $\langle A, \alpha_A \circ f \rangle$ è completa per f, cioè $\alpha_A \circ f = \alpha_A \circ f \circ \alpha_A$.

La tesi si occupa delle tecniche di trasformazione dei domini astratti (cfr. [3]) al fine di ottenere domini completi per una certa funzione concreta fissata, o, più in generale, per un insieme possibilmente infinito di funzioni concrete. Intuitivamente, dato un dominio astratto A che non sia completo per una funzione concreta f, ci chiediamo se possiamo arricchire il dominio A al fine di ottenere un dominio completo per f. Si richiede inoltre che il nuovo dominio completo sia il più astratto con questa proprietà, cioè che sia ottenuto aggiungendo la minore quantità di oggetti al dominio di partenza. Dato un dominio astratto A, il più astratto dominio che contenga A e sia completo per f è detto *minima estensione completa* di A per f. In modo duale, ci domandiamo se possiamo semplificare il dominio A per ottenere un dominio completo, cioè siamo interessati a trovare un dominio completo che sia contenuto in A e sia il più concreto con questa proprietà. In questo caso parliamo di *massima riduzione completa* (greatest complete kernel) di A per f.

2. – Problemi di completezza.

L'idea della tesi nasce dall'osservazione che possiamo descrivere il grado di precisione di un dominio astratto con una opportuna generalizzazione della definizione standard di completezza ad un insieme di funzioni concrete (possibilmente tipate). La completezza diviene quindi il linguaggio con il quale esprimiamo la precisione relativa dei domini astratti nel calcolare una certa funzione f. La definizione generale di completezza utilizza due domini astratti differenti, al fine di separare la fase di calcolo e l'osservazione del risultato. Questo ci permette di considerare domini astratti le cui computazioni siano più precise possibili, quando i risultati sono osservati in un qualche altro dominio astratto. Il *dominio di calcolo* viene utilizzato per effettuare realmente il calcolo, ed è una astrazione del dominio della funzione concreta. Nel *dominio di osservazione*, che è una astrazione del codominio della funzione concreta, osserviamo il risultato della computazione astratta. Ad esempio, dati due domini concreti C e D e una funzione $f: C \to D$, possiamo fissare un dominio di osservazione $A \in uco(D)$, che intuitivamente rappresenta la proprietà da osservare, e cerchiamo il più astratto dominio $B \in uco(C)$ tale che ogni computazione astratta nel dominio B sia precisa quanto le computazioni concrete, quando il risultato è osservato in A. Formalmente, il dominio B deve essere una soluzione del seguente problema di completezza:

$$\alpha_A \circ f = \alpha_A \circ f \circ \alpha_B. \tag{1}$$

Dualmente, possiamo fissare un dominio di calcolo $B \in uco(C)$ e cercare il più concreto dominio di osservazione $A \in uco(C)$ che soddisfi il problema (1). Intuitivamente, la soluzione di questo problema descrive cosa possiamo osservare in una computazione astratta senza introdurre errori di approssimazione. Si noti come i problemi di completezza rappresentino un linguaggio compatto ed elegante per descrivere il grado di precisione di domini astratti.

3. – Dai problemi di completezza alle equazioni tra domini astratti.

Al fine di risolvere i problemi di completezza, associamo ad ogni problema un sistema di equazioni (ricorsive) tra domini astratti in modo tale che le soluzioni delle equazioni soddisfino il problema di completezza. Intuitivamente, una equazione tra domini astratti è una equazione del tipo $X = F(X)$, dove F è una funzione monotona che trasforma domini astratti (cioè operatori di chiusura). Le soluzioni delle equazioni sono i punti fissi dell'operatore F e possono essere calcolate in modo iterativo. Il passaggio dai problemi di completezza alle equazioni tra domini è quindi il punto principale di questa costruzione.

Per ogni problema di completezza, forniamo un metodo costruttivo per calcolare il dominio di osservazione partendo dal dominio di calcolo, e viceversa. Tale costruzione prevede come ipotesi la continuità della funzione concreta. Si noti che tale ipotesi non è restrittiva, in quanto la quasi totalità delle semantiche dei linguaggi di programmazione sono funzioni continue. Data una funzione semantica continua $f: C \to D$, dimostriamo che la relazione tra una qualsiasi coppia di domini di osservazione e di calcolo che soddisfino il problema di completezza (1) è precisamente un'aggiunzione, e studiamo in dettaglio le proprietà di tale relazione.

Dimostriamo che vale $\alpha_A \circ f = \alpha_A \circ f \circ \alpha_B$ se e solo se A è più astratto di un certo dominio $L_f(B)$, dipendente da B. Dualmente, questo accade se e solo se B è più concreto di un certo dominio $R_f(A)$, dipendente da A.

Le funzioni R_f: $uco(D) \to uco(C)$ e L_f: $uco(C) \to uco(D)$ formano quindi un'aggiunzione. Sfruttando questi risultati, siamo in grado di caratterizzare:

- la minima estensione completa di B rispetto ad un dominio di osservazione A;
- la massima riduzione completa di A rispetto ad un dominio di calcolo B.

Un punto fondamentale di questo risultato è che forniamo esplicitamente una caratterizzazione costruttiva di entrambi gli operatori R_f e L_f, e quindi della minima estensione completa e massima riduzione completa.

L'ultima parte della tesi affronta il problema di rappresentare gli oggetti nei domini astratti completi. A tal fine, forniamo un modo naturale di rappresentare gli oggetti astratti tramite opportuni frammenti della logica proposizionale intuizionista e lineare. In particolare, nel caso di funzioni concrete additive, si ereditano le caratteristiche basilari dei quantali (cfr. [4]), le quali permettono di semplificare molti dei risultati per la risoluzione delle equazioni tra domini astratti.

BIBLIOGRAFIA

[1] COUSOT P. e COUSOT R., *Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints*, Conference Record of the 4th ACM Symposium on Principles of Programming Languages (*POPL '77*) (1977), 238-252.

[2] COUSOT P. e COUSOT R., *Systematic design of program analysis frameworks*, Conference Record of the 6th ACM Symposium on Principles of Programming Languages (*POPL '79*) (1979), 269-282.

[3] FILÉ G., GIACOBAZZI R. e RANZATO F., *A unifying view of abstract domain design*, ACM Computing Surveys, **28(2)** (1996), 333-336.

[4] ROSENTHAL K. I., *Quantales and their Applications*, Longman Scientific & Technical (1990).

Laboratoire d'Informatique - École Polytechnique, F-91128 Palaiseau Cedex, Francia
e-mail: scozzari@lix.polytechnique.fr

Dottorato in Logica Matematica e Informatica Teorica (sede amministrativa: Siena) - Ciclo X
Direttori di ricerca: Prof. Roberto Giacobazzi, Università di Verona
Prof. Giorgio Levi, Università di Pisa