On Adhesivity of EGGs

Roberto Biondo

Dipartimento di Informatica
Università di Pisa

Friday 29th November, 2024

Contenuti

Formalismi Grafici

Formalismi grafici:

- Analisi delle dipendenze
- Bioinformatica
- Sistemi di riscrittura
- Ottimizzazione
- Ingegneria del software

Categorie

Modello *categoriale*.

Una categoria $\mathscr{C}$ è composta da:

- *oggetti*
- *morfismi* (o *frecce*)

$$\begin{tikzcd} id_A \arrow[loop left] & A \arrow[r, "f"] & B \arrow[loop right] & id_B \end{tikzcd}$$

La categoria i cui oggetti sono gli insiemi e i cui morfismi sono le funzioni tra essi è denominata **Set**.

$f : A \to B$ può essere:

- un *monomorfismo* : $f \circ g = f \circ h \Rightarrow g = h$;
- un *epimorfismo* : $g \circ f = h \circ f \Rightarrow g = h$;
- un *isomorfismo* : esiste k tale che $f \circ k = id_B$ e $k \circ f = id_A$.

Funtori, Trasformazioni Naturali

Un *funtore* $F : \mathscr{C} \to \mathscr{D}$:

$$\begin{tikzcd} F(A) \arrow[loop left, "F(id_A) = id_{F(A)}"] \arrow[r, "F(f)"] \arrow[rr, bend left, "F(g \circ f) = F(g) \circ F(f)"] & F(B) \arrow[r, "F(g)"] & F(C) \arrow[loop right, "F(id_C) = id_{F(C)}"] \end{tikzcd}$$

Una *trasformazione naturale* $\eta : F \dot{\to} G$:

$$\begin{tikzcd} F(A) \arrow[r, "\eta_A"] \arrow[d, "F(f)"'] & G(A) \arrow[d, "G(f)"] \\ F(B) \arrow[r, "\eta_B"'] & G(B) \end{tikzcd}$$

Definizione dei Grafi

Categoria di funtori $[\mathscr{C}, \mathscr{D}]$:

- Oggetti: funtori (con dominio $\mathscr{C}$ e codominio $\mathscr{D}$)
- Morfismi: trasformazioni naturali

Le categorie di funtori il cui codominio è **Set** sono dette *categorie di prefasci*.

Un grafo G è costituito da:

- Un insieme di nodi V
- Un insieme di archi E
- $s : E \to V$ funzione sorgente
- $t : E \to V$ funzione destinazione

Un morfismo fra grafi è una coppia $f_V : V_1 \to V_2$, $f_E : E_1 \to E_2$ che rispetta sorgente e destinazione.

In altre parole, un grafo è un oggetto della categoria

$$[E \overset{s}{\underset{t}{\rightrightarrows}} V, \mathbf{Set}]$$

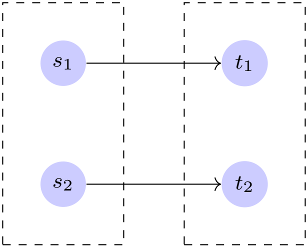
Grafi ed Equivalenze

Un *grafo con equivalenza* $(G, \sim)$ è dato da:

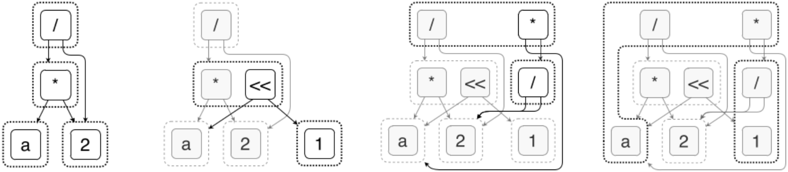
- Un grafo G
- Una relazione di equivalenza $\sim$ definita sui vertici

Ossia, un elemento di $[E \rightrightarrows V \to Q, \mathbf{Set}]$, tale che la freccia $q : V \to Q$ sia surgettiva (epi).
Un E-Grafo
$(G, \sim)$ è un grafo con equivalenza tale che

$$\frac{s_G(e) \sim s_G(e')}{t_G(e) \sim t_G(e')}$$



EGGs



La categoria degli E-Grafi è detta **EGG**, dal nome della libreria che li implementa in Rust.

La definizione attuale è esplicitamente operativa, e risulta difficile per ottenere dei risultati teorici.

Tuttavia gli utilizzi sono molteplici, dalla *equality saturation* all'ottimizzazione in fase di compilazione.

Pullback, Pushout, Equalizzatori

$$\begin{tikzcd} P' \arrow[dr, dashed, "v"] \arrow[drr, bend left, "p'_1"] \arrow[ddr, bend right, "p'_2"'] & & \\ & P \arrow[r, "p_1"] \arrow[d, "p_2"'] & A \arrow[d, "f"] \\ & B \arrow[r, "g"'] & C \end{tikzcd}$$

(a) Pullback

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "g"'] & B \arrow[d, "q_1"] \arrow[ddr, bend left, "q'_1"] & \\ C \arrow[r, "q_2"'] \arrow[drr, bend right, "q'_2"'] & Q \arrow[dr, dashed, "u"] & \\ & & Q' \end{tikzcd}$$

(b) Pushout

In **Set**:

- Pullback di f e g: $P = \{(x, y) \in A \times B \mid f(x) = g(y)\}$
- Pushout di f e g: $Q = (A + B)/\sim$

Approccio DPO

$$L \xleftarrow{l} K \xrightarrow{r} L$$

(a) Regola

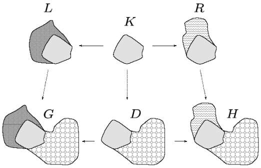
$$\begin{tikzcd}
L \arrow[d, "m_L"'] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d, "m_K"] & R \arrow[d, "m_R"] \\
G & D \arrow[l, "l*"] \arrow[r, "r*"'] & H
\end{tikzcd}$$

(b) Passo di derivazione

l e r sono morfismi, il primo dei quali mono, di $\mathscr{C}$, mentre i quadrati sono pushout.

Tecniche per parlare di:

- Confluenza
- Terminazione
- Parallelismo e concorrenza

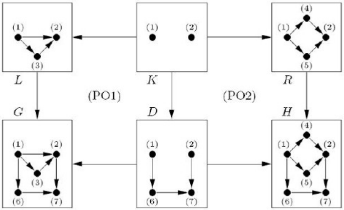


Adesività

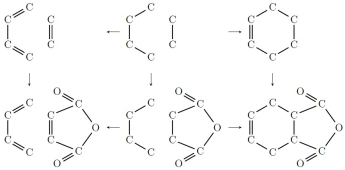
Una categoria $\mathscr{C}$ è adesiva se:

- ha tutti i pushout di monomorfismi
- ha tutti i pullback
- i pushout di monomorfismi sono *Van Kampen*
  - ossia tali pushout sono stabili sotto pullback e i pullback sono stabili sotto pushout combinati con pullback

L'adesività permette di estendere l'approccio DPO a categorie di grafi più complesse.



(a) DPO su grafi



(b) DPO per reazione di Dies-Alder

Conclusioni

Nel lavoro di tesi abbiamo

- Formalizzato gli E-Grafi mediante categorie di prefasci
- Provato l'adesività rispetto a una sottoclasse di monomorfismi

Come lavori futuri vogliamo

- Sostituire la categoria **Set** con una generica categoria adesiva
- Generalizzare il risultato ad altri tipi di grafi, e.g., ipergrafi e term graph