

Indice

	Pagina
1 Problemi considerati	2
1.1 Alcune definizioni fondamentali della teoria dei grafi	2
1.2 Applicazione dei grafi alla teoria degli insiemi	4
1.3 Bisimulazione	5
1.3.1 Bisimulazione massima	7
1.3.2 Interpretazione insiemistica della bisimulazione	8
Bibliografia	11

1 Problemi considerati

1.1 Alcune definizioni fondamentali della teoria dei grafi

Riportiamo la definizione di *relazione binaria* su uno o due insiemi, che sarà utile per definire formalmente il concetto di *grafo*, che sarà fondamentale all'interno di questo elaborato.

Definizione 1.1. Chiameremo *relazione binaria* su A, B qualsiasi sottoinsieme del prodotto cartesiano $A \times B$.

Chiameremo *relazione binaria* su A qualsiasi sottoinsieme del prodotto cartesiano $A \times A$.

Diremo che u, v sono *in relazione* rispetto a b se $(u, v) \in b$. In questo caso useremo la notazione ubv .

Ad una relazione binaria su un insieme è possibile associare relazioni binarie che la contengono, le cosiddette *chiusure*. Ogni chiusura è costituita dall'unione della relazione iniziale e di un insieme di coppie costruito con un criterio differente a seconda del tipo di chiusura.

Quest'ultimo insieme fornisce la proprietà che caratterizza la chiusura (riflessività, simmetria, transitività).

Definizione 1.2. Sia b una relazione binaria su N . Diamo le seguenti definizioni

- *Chiusura riflessiva*: $b_r = b \cup \{(x, x) \mid \forall x \in N\}$
- *Chiusura simmetrica*: $b_s = b \cup \{(y, x) \mid \forall x, y : (x, y) \in b\}$
- *Chiusura transitiva*: $b_t = b \cup \{(x, z) \mid \forall x, z : \exists y : (x, y) \in b \wedge (y, z) \in b\}$

Con queste premesse possiamo definire un *grafo* come segue:

Definizione 1.3. Sia $N \neq \emptyset$. Sia \rightarrow una relazione binaria su N . Chiameremo *grafo* la coppia $G = (N, \rightarrow)$.

In questo caso

- N è l'insieme dei nodi;
- \rightarrow è la relazione di raggiungibilità: $a \rightarrow b$ ($a, b \in N$) significa "nel grafo G esiste un arco dal nodo a al nodo b ".

Diremo che G è un *grafo direzionato* se $u \rightarrow v \not\Rightarrow v \rightarrow u$ (cioè se \rightarrow non è una relazione binaria simmetrica). Altrimenti diremo che G è un *grafo non direzionato*.

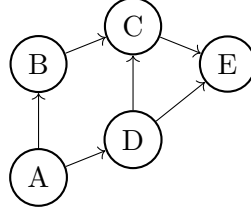


Figura 1

Osservazione 1.1. Il grafo G' rappresentato dalla coppia (N, \rightarrow') , dove \rightarrow' è la chiusura simmetrica della relazione di raggiungibilità \rightarrow di un grafo direzionato, è un grafo non direzionato.

Esempio 1.1. Il grafo di Figura 1 è descritto dalla coppia

- $N = \{A, B, C, D, E\}$
- $\rightarrow = \{(A, B), (A, D), (B, C), (D, C), (C, E), (D, E)\}$

Un grafo è quindi un insieme di elementi (i *nodì*) accoppiato con un insieme di relazioni tra questi elementi (gli *archi* o *rami*).

E' naturale associare questo concetto all'idea di percorso: ogni grafo è definito da un insieme di "luoghi" ed un insieme di "percorsi" che consentono di viaggiare da un luogo ad un altro.

La seguente definizione sorge in modo spontaneo da questo punto di vista:

Definizione 1.4. Sia $G = (N, \rightarrow)$ un grafo. Siano $u, v \in N$. Diremo che v è raggiungibile da u , o in alternativa esiste un cammino da u a v , o ancora $u \rightarrow_t v$ (la t in pedice è letta come *transitivo*), se \exists una sequenza $x_n \subset N$ finita di lunghezza $K : x_K = v, x_0 = u, x_n \rightarrow x_{n+1}$.

L'esistenza di un cammino tra nodi fornisce un criterio immediato per suddividere un grafo in più pezzi. Riportiamo questo criterio come formulato in [2]:

Definizione 1.5. Sia $G = (N, \rightarrow)$ un grafo. Chiameremo *grafo delle componenti fortemente connesse (CFC)* di G il grafo $G^{CFC} = (N^{CFC}, \rightarrow^{CFC})$ se

- $N^{CFC} = \{C : C = \{m \in N : \forall n \in C, m \rightarrow_t n\}\}$
- $\rightarrow^{CFC} = \{(A, B) \in N^{CFC} \times N^{CFC} : A \neq B, \exists m \in A, n \in B : m \rightarrow n\}$

Segue immediatamente la seguente proprietà:

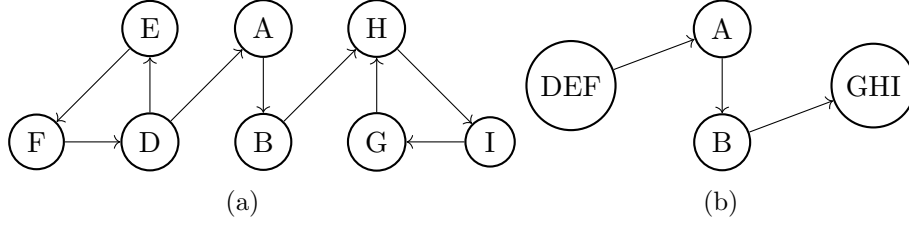


Figura 2

Proposizione 1.1. Sia G^{CFC} il grafo delle CFC di un grafo G generico. Allora G^{CFC} è aciclico.

Dimostrazione. Suppongo per assurdo che in G^{CFC} esista un ciclo, quindi è possibile partendo da un $A \in N^{CFC}$ ritornare ad A seguendo un cammino che passa per un numero finito di nodi di N^{CFC} , che raggruppiamo nell'insieme $X \subset N^{CFC}$, con $A \notin X$. Ma allora qualsiasi nodo $n \in N : (\exists B \in N^{CFC} \wedge n \in B \wedge B \in X)$ è raggiungibile da qualsiasi nodo di A . Quindi tutti i nodi inclusi nel ciclo devono appartenere alla stessa componente, ma questa deduzione è in contrasto con l'ipotesi. \square

Esempio 1.2. La figura 2.a rappresenta un grafo generico, la figura 2.b rappresenta il suo grafo delle componenti fortemente connesse associato.

1.2 Applicazione dei grafi alla teoria degli insiemi

Per motivi che risulteranno chiari in seguito risulta conveniente fornire un'interpretazione insiemistica della nozione di grafo vista sopra.

Definizione 1.6. Sia $G = (N, \rightarrow)$ un grafo direzionato. Sia $u \in N : \forall v \in N, u \rightarrow_t v$, cioè ogni nodo di G è raggiungibile da u . Allora la terna (N, \rightarrow, u) si dice *accessible pointed graph*, o *APG*.

Se si associano la relazione di raggiungibilità \rightarrow e la relazione di appartenenza \in , e se si interpreta ogni nodo come un insieme, ad ogni APG è possibile associare un unico insieme. Tuttavia ad ogni insieme si può associare più di un APG [1].

Esempio 1.3. Il grafo in figura 3.a e quello in 3.b rappresentano l'insieme $A = \{\emptyset\}$. Il grafo in figura 3.c rappresenta l'insieme $A = \{A\}$. Il grafo in figura 3.d rappresenta l'insieme $A = \{\emptyset, \{\emptyset\}\}$

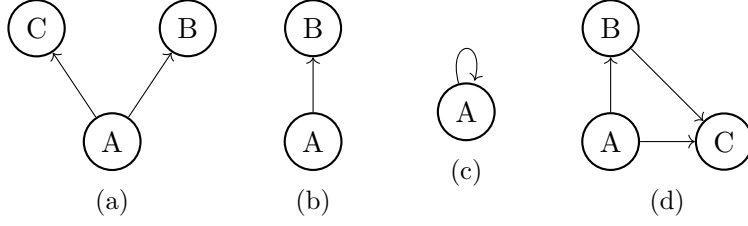


Figura 3

1.3 Bisimulazione

Definizione 1.7. Siano $G_1 = (N_1, E_1), G_2 = (N_2, E_2)$ due grafi. Diremo che G_1, G_2 sono *bisimili* (come in [1]) rispetto alla relazione binaria b su N_1, N_2 se $\forall u \in N_1, v \in N_2$ valgono congiuntamente le seguenti condizioni

- $\forall u' \in N_1 : u \rightarrow u', ubv \implies \exists v' \in N_2 : (u'bv' \wedge v \rightarrow v')$
- $\forall v' \in N_2 : v \rightarrow v', ubv \implies \exists u' \in N_1 : (u'bv' \wedge u \rightarrow u')$

Se per una determinata coppia di grafi esiste almeno una bisimulazione che li lega, possiamo dedurre che i nodi dei due grafi mostrano un comportamento simile. Questo concetto verrà chiarito in seguito, per ora forniamo solamente la seguente definizione:

Definizione 1.8. Siano $G_1 = (N_1, E_1, v_1), G_2 = (N_2, E_2, v_2)$ due APG. Diremo che sono *bisimili* se \exists una bisimulazione b su $G_1, G_2 : v_1bv_2$.

Per introdurre i risultati che verranno presentati in seguito risulta utile definire la bisimulazione su un unico grafo:

Definizione 1.9. Sia $G = (N, \rightarrow)$ un grafo, e b una relazione binaria su N . Diremo che b è una *bisimulazione* se $\forall u, v \in N$ valgono congiuntamente le seguenti condizioni

- $\forall u' \in N : u \rightarrow u', ubv \implies \exists v' \in N : (u'bv' \wedge v \rightarrow v')$
- $\forall v' \in N : v \rightarrow v', ubv \implies \exists u' \in N : (u'bv' \wedge u \rightarrow u')$

Proponiamo ora un risultato che consentirà di considerare più generali di quanto sembrino in apparenza alcuni risultati presentati nel seguito del lavoro.

Teorema 1.1. *Sia b una bisimulazione sul grafo G . La sua chiusura riflessiva, simmetrica o transitiva è ancora una bisimulazione su G .*

Dimostrazione. Consideriamo separatamente le tre relazioni b_r, b_s, b_t , rispettivamente la chiusura riflessiva, simmetrica e transitiva:

b_r Per definizione $b \subset b_r$, quindi è sufficiente dimostrare che b_r è una bisimulazione quando gli argomenti $u, v \in N$ non sono distinti.

Sia $u \in N$. Chiaramente per definizione di b_r si ha ub_ru . Se $\exists u' \in N : u \rightarrow u'$ allora (sempre per definizione di b_r) si ha $u'b_ru'$.

b_s Per definizione $b \subset b_s$, quindi è sufficiente dimostrare che b_s è una bisimulazione quando per gli argomenti $u, v \in N$ si ha ubv ma non vbu .

Sia $(u, v) \in N \times N$. Allora

$$ub_sv \implies ubv \vee vbu$$

Suppongo ad esempio che vbu .

$$\begin{aligned} &\implies \forall v' \in N : (v \rightarrow v') \exists u' \in N : (u \rightarrow u' \wedge v'bu') \\ &\implies u'b_sv' \end{aligned}$$

e

$$\begin{aligned} &\implies \forall u' \in N : (u \rightarrow u') \exists v' \in N : (v \rightarrow v' \wedge v'bu') \\ &\implies u'b_sv' \end{aligned}$$

cioè sono dimostrate le due condizioni caratteristiche della bisimulazione.

La dimostrazione è analoga se ubv .

b_t Per definizione $b \subset b_t$, quindi è sufficiente dimostrare che b_t è una bisimulazione quando per gli argomenti $u, v, z \in N$ si ha ubv, vbz ma non ubz .

Sia $(u, v, z) \in N \times N \times N$ con questa proprietà. Allora $\forall u' \in N : u \rightarrow u' \implies \exists v' \in N : v \rightarrow v' \wedge u'bv'$. Inoltre $\exists z' : z \rightarrow z' \wedge v'bz'$.

Riordinando si ha $u'bv', v'bz'$. Allora per definizione di b_t , $u'b_tz'$.

In modo speculare si ottiene la seconda condizione caratteristica della bisimulazione.

□

Da questa proposizione si deduce il seguente corollario, che risulta dall'applicazione iterativa delle tre chiusure viste in precedenza:

Corollario 1.1. *Ad ogni bisimulazione b è possibile associare una bisimulazione $\tilde{b} : b \subset \tilde{b} \wedge \tilde{b}$ è una relazione di equivalenza.*

Concludiamo la parte relativa ai risultati generali sulla bisimulazione con la seguente proposizione, che sarà utile in seguito

Proposizione 1.2. *Siano b_1, b_2 due bisimulazioni su G_1, G_2 . Allora $b = b_1 \cup b_2$ è ancora una bisimulazione.*

Dimostrazione. Siano $u, v : ubv$. Sia $u' : u \rightarrow u'$. Allora deve essere $ub_1v \vee ub_2v$. Ma quindi $\exists v' : (v \rightarrow v' \wedge u'b_1|_2v')$. \square

1.3.1 Bisimulazione massima

Definiamo ora il concetto di *bisimulazione massima*, che gioca un ruolo chiave nella risoluzione dei problemi considerati in questo elaborato.

Definizione 1.10. Diremo che una bisimulazione b_M su G_1, G_2 è *massima* se $\forall b : "b \text{ è una bisimulazione su } G_1, G_2" \text{ si ha } ubv \implies ub_Mv \forall a \in N_1, b \in N_2$.

Naturalmente la bisimulazione massima dipende dai due grafi presi in esame. Possiamo dedurre alcune caratteristiche in modo molto semplice:

Proposizione 1.3. *Valgono le seguenti proprietà:*

1. *La bisimulazione massima su due grafi G_1, G_2 è unica*
2. *La bisimulazione massima è una relazione di equivalenza*

Dimostrazione. Le proprietà seguono banalmente dal Corollario 1.1 e dall'Osservazione 1.2.

1. Suppongo per assurdo che esistano due bisimulazioni massime b_{M_1}, b_{M_2} . La loro unione è ancora una bisimulazione, che è "più massima" delle supposte bisimulazioni massime.
2. Se per assurdo la bisimulazione massima non fosse una relazione di equivalenza, potremmo considerare la sua chiusura riflessiva, simmetrica e transitiva, che sarebbe "più massima" ed anche una relazione di equivalenza.

\square

Naturalmente il concetto di *bisimulazione massima* può essere definito anche su unico grafo G . Questo caso si rivelerà di grande interesse nel seguito. Per ora dimostriamo il seguente risultato:

Teorema 1.2. *Sia G un grafo (finito). Allora $\exists b_M$ la bisimulazione massima su G .*

Dimostrazione. Può esistere solamente un numero finito di relazioni binarie su G , e questo numero fornisce un limite superiore al numero massimo di bisimulazioni su G . Allora possiamo considerare l'unione di questo numero finito di bisimulazioni, che sarà chiaramente la bisimulazione massima. \square

1.3.2 Interpretazione insiemistica della bisimulazione

Il seguente teorema è la prova che la bisimulazione può essere utilizzata per verificare l'uguaglianza tra insiemi rappresentati da due APG differenti:

Teorema 1.3. *Due APG sono bisimili \iff rappresentano lo stesso insieme.*

Dimostrazione. Da dimostrare... \square

Tenendo conto di quanto affermato nella sezione 1.2, il Teorema 1.3 dimostra che la bisimulazione può sostituire la relazione di uguaglianza tra insiemi quando questi sono rappresentati da degli APG [1].

Dopo questa considerazione, risulta naturale definire il seguente concetto:

Definizione 1.11. Sia b una bisimulazione su G che sia anche una relazione di equivalenza. Definiamo un nuovo grafo $G_b = (N_b, \rightarrow_b)$ come in [2], che chiameremo *contrazione rispetto alla bisimulazione b di G* :

- $N_b = \{A = \{m \in N : \forall n \in A, mbn\}\}$
- $[m]_b \rightarrow_b [n]_b \iff \exists c \in [n]_b : m \rightarrow c$

Risulta conveniente definire la *classe del nodo a* rispetto alla bisimulazione b , con la notazione $[a]_b$ come il nodo di N_b a cui appartiene il nodo a .

La Definizione 1.11 è di fondamentale importanza per la seguente osservazione:

Proposizione 1.4. *Sia G un grafo, e sia G_b come nella Definizione 1.11, per una bisimulazione b qualsiasi. Allora G, G_b sono bisimili.*

Dimostrazione. Sia $\equiv \subset N \times N_b$ la relazione binaria definita come segue:

$$m \equiv M \iff M = [m]_b$$

Vogliamo dimostrare che tale relazione è una bisimulazione su i grafi G, G_b . Supponiamo che $x \equiv X$, e che $x \rightarrow y$ per qualche $y \in N$. Chiamiamo $Y := [y]_b$. Allora, per la Definizione 1.11, si ha $X \rightarrow Y$. Inoltre vale banalmente $y \equiv Y$.

Per dimostrare la seconda condizione caratteristica della bisimulazione, supponiamo che $x \equiv X$, e che $X \rightarrow Y$ per qualche $Y \in N_b$. Sempre per la Definizione 1.11 deve esistere un $y \in Y : (y \equiv Y \wedge x \rightarrow y)$. \square

La Proposizione 1.4 ha una conseguenza ovvia, che risulta evidente per il Teorema 1.3:

Corollario 1.2. *Sia b una bisimulazione che sia anche una relazione di equivalenza. Allora l'APG (G, v) e l'APG $(G_b, [v]_b)$ rappresentano lo stesso insieme.*

Quindi risulta naturale sfruttare le proprietà della bisimulazione per minimizzare la rappresentazione di insiemi, considerando che è sufficiente una bisimulazione sulla rappresentazione iniziale per ottenere una rappresentazione equivalente. Definiamo una relazione d'ordine sulle rappresentazioni:

Definizione 1.12. Diremo che la rappresentazione (G_a, v_a) di un insieme è *minore* della rappresentazione equivalente (G_b, v_b) se $\#N_a < \#N_b$.

Diremo che una rappresentazione è *minima* se non esiste un'altra rappresentazione minore.

Osservazione 1.2. La *contrazione per bisimulazione* di un grafo ha sempre un numero di nodi minore o uguale di quello del grafo iniziale.

Concludiamo la sezione con il seguente risultato, che stabilisce in modo univoco la bisimulazione prescelta per minimizzare la rappresentazione di un dato insieme:

Teorema 1.4. *Sia (G, v) un APG rappresentante un insieme. Sia b_M la bisimulazione massima su (G, v) . Allora la contrazione per bisimulazione indotta da b_M su (G, v) fornisce la rappresentazione minima dell'insieme.*

Dimostrazione. Suppongo per assurdo che esista una bisimulazione b_N su (G, v) che fornisce una contrazione avente un numero di nodi strettamente inferiore alla contrazione indotta da b_M . Ma questo implica che esistono

almeno due nodi di G che sono in relazione secondo b_N e non secondo b_M . Chiaramente questa deduzione è in contrasto con il fatto che b_M è la bisimulazione massima.

Suppongo per assurdo che, dopo la contrazione indotta da b_M , sia possibile trovare una nuova bisimulazione b_O su $(G_{b_M}, [v]_{b_M})$ che induca una contrazione avente un numero di nodi strettamente inferiore a quello di $(G_{b_M}, [v]_{b_M})$. Chiaramente $b_O \subset N_{b_M} \times N_{b_M}$.

Definisco una nuova bisimulazione $b_{\widetilde{M}} \subset N \times N$ tale che

$$xb_{\widetilde{M}}y \iff (xb_My \vee [x]_{b_M}b_O[y]_{b_M})$$

Per definizione di bisimulazione massima bisogna avere $b_{\widetilde{M}} \subset b_M$, quindi non è possibile che la contrazione indotta da b_O sia una rappresentazione minore di quella indotta da b_M . \square

Bibliografia

- [1] Agostino Dovier, Carla Piazza, and Alberto Policriti. A fast bisimulation algorithm. In *International Conference on Computer Aided Verification*, pages 79–90. Springer, 2001.
- [2] Raffaella Gentilini, Carla Piazza, and Alberto Policriti. From bisimulation to simulation: Coarsest partition problems. *Journal of Automated Reasoning*, 31(1):73–103, 2003.