BisPy

un pacchetto Python per il calcolo della massima bisimulazione di grafi diretti

Francesco Andreuzzi

Università degli Studi di Trieste, Dipartimento di Ingegneria e Architettura

8 Luglio 2021

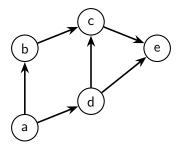
Anno accademico 2020-2021

Relatore: Prof. Alberto Casagrande

Grafi orientati, terminologia

$$V = \{a, b, c, d, e\}$$

$$E = \{\langle a, b \rangle, \langle b, c \rangle, \langle a, d \rangle, \langle c, e \rangle, \langle d, e \rangle\}$$



Definizione

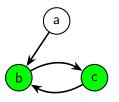
Se $\langle a,b \rangle \in E \implies b$ è un nodo figlio di a.

Un nodo privo di figli è detto pozzo.

Bisimulazione

Definizione: Bisimulazione $\mathcal{B} \subseteq V \times V$

Se $(a, b) \in \mathcal{B}$, allora da a è possibile spostarsi verso un nodo *figlio* di a che sia in relazione con un nodo *figlio* di b, e viceversa.



$$\mathcal{B} = \{(b,c)\}$$

Definizione: Nodi bisimili

I nodi a,b sono bisimili se esiste una bisimulazione $\mathcal B$ tale che $(a,b)\in \mathcal B.$

Massima bisimulazione

Definizione

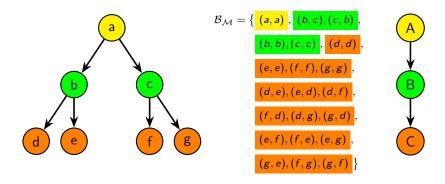
La massima bisimulazione su G=(V,E) è la bisimulazione $\mathcal{B}_{\mathcal{M}}$ tale che:

 $\mathcal{R} \subseteq \mathcal{B}_{\mathcal{M}}$ per ogni bisimulazione \mathcal{R} su G.

Si può dimostrare che:

- 1 La massima bisimulazione è unica;
- ② La massima bisimulazione è una relazione di equivalenza \implies induce un partizionamento su V;
- $\mathfrak{B}_{\mathcal{M}}(a,b) \equiv \bigcup_{\substack{\mathcal{R} \text{ è una} \\ \text{his imulazione}}} \mathcal{R}$
- ${\color{red} \bullet} \; \mathcal{B}_{\mathcal{M}} \; \text{definita come nel punto 3. è una bisimulazione.}$

Massima bisimulazione -- Esempio e applicazioni pratiche



- Ricerca di nodi equivalenti;
- Minimizzazione;

- Concurrency theory;
- XML indexing.

Algoritmi per il calcolo della massima bisimulazione

Non incrementali, complessità asintotica $|E| \log |V|$

- Algoritmo di Paige-Tarjan;
- Algoritmo di Dovier-Piazza-Policriti;

$$\mathcal{B}_{\mathcal{M}}(u,v) \implies \operatorname{rank}(u) = \operatorname{rank}(v).$$

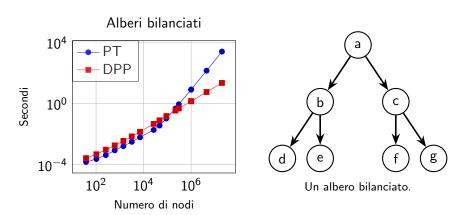
Incrementali

Algoritmo incrementale di Saha.

- Python 3;
- Open source (https://github.com/fAndreuzzi/BisPy);
- Testato su diverse tipologie di grafo, di dimensioni varie.

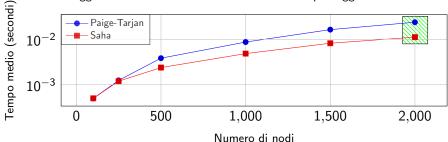
Example

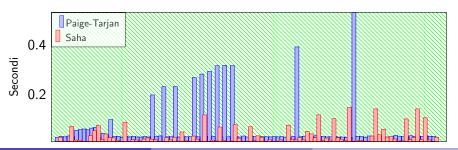
Risultati sperimentali I -- Paige-Tarjan, Dovier-Piazza-Policriti



Risultati sperimentali II -- Paige-Tarjan, Saha

Aggiornamento della massima bisimulazione dopo l'aggiunta di un arco





Conclusione

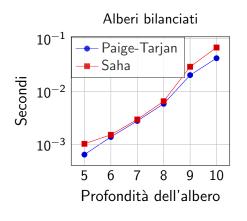
Il software sviluppato consente di osservare e confrontare il comportamento degli algoritmi su varie tipologie di grafo.

Sviluppi futuri

- Pubblicazione di BisPy su JOSS (Journal of Open Source Software);
- Ricalcolo incrementale della massima bisimulazione dopo la rimozione di un arco;
- Calcolo della massima bisimulazione con labeled edges;
- Integrazione con altre librerie (NetworkX);
- Cython per parti critiche del codice.

Grazie per l'attenzione

Risultati sperimentali III -- Paige-Tarjan, Saha



Raggruppamento per tipo di arco

