The Subgraph Bisimulation Problem

Agostino Dovier Carla Piazza

Abstract

We study the complexity of the Subgraph Bisimulation Problem, which stands to Graph Bisimulation as Subgraph Isomorphism stands to Graph Isomorphism and we prove its NP-Completeness. Our analysis is motivated by its applications to Semistructured Databases.

Keywords: Bisimulation, Complexity, Semistructured Data

I. Introduction

Graph and Subgraph Isomorphism are two basic algorithmic problems [6]. Although the latter is NP-complete, the lower bound for the time complexity of the former is still an open and very attractive issue. *Bisimulation*, a relation weaker than isomorphism, emerged as a fundamental property in various areas of Computer Science [1], [8]. Polynomial time procedures can be used to check whether two distinct graphs are bisimilar [9]. The *Subgraph Bisimulation problem* consists in *identifying a subgraph G_2' of a graph G_2 bisimilar to a given graph G_1*.

The graphical query language G-log [10] uses this notion for retrieving data from semistructured information [4]. Data retrieval in the languages UnQL [2] and Graphlog [3] can be implemented on the basis of this notion, as shown in [5]. Relationships between web-like databases and hypersets (where bisimulation is used for testing equivalence) are pointed out in [7].

We prove that the Subgraph Bisimulation problem is NP-complete. As a consequence, data retrieval based on this notion in its generality is not feasible. However, a class of queries that allows polynomial time data retrieval based on Subgraph Bisimulation is shown in [5].

A. Dovier is with the Dip. di Matematica e Informatica, Univ. di Udine. Via delle Scienze 206, 33100 Udine (Italy). email: dovier@dimi.uniud.it

C. Piazza is with the Dip. di Informatica, Univ. Ca' Foscari di Venezia. Via Torino 155, 30173 Mestre—Venezia (Italy). email: piazza@dsi.unive.it

II. Basic Definitions and Results

A *directed graph* (*graph*) is a pair $G = \langle N, E\rangle$, where N is the set of nodes and $E \subseteq N \times N$ is the set of edges. $G_1 = \langle N_1, E_1\rangle$ is a *subgraph* of $G_2 = \langle N_2, E_2\rangle$ if $N_1 \subseteq N_2$ and $E_1 \subseteq E_2$. $G_1 = \langle N_1, E_1\rangle$ and $G_2 = \langle N_2, E_2\rangle$ are *isomorphic* ($G_1 \equiv G_2$) if there exists a bijection $f : N_1 \to N_2$ s.t.: $\langle u_1, v_1\rangle \in E_1 \Leftrightarrow \langle f(u_1), f(v_1)\rangle \in E_2$. The *Graph Isomorphism problem* $GI(G_1, G_2)$ amounts to determining whether $G_1 \equiv G_2$; the *Subgraph Isomorphism problem* $SI(G_1, G_2)$ consists in deciding whether there exists a subgraph G_2' of G_2 s.t. $G_1 \equiv G_2'$.

A *bisimulation* [1] between $G_1 = \langle N_1, E_1\rangle$ and $G_2 = \langle N_2, E_2\rangle$ is a relation $b \subseteq N_1 \times N_2$ s.t.: $(1)\ u_1 \in N_1 \Rightarrow \exists u_2 \in N_2(u_1\, b\, u_2)$, $(2)\ u_2 \in N_2 \Rightarrow \exists u_1 \in N_1(u_1\, b\, u_2)$, $(3)\ u_1\, b\, u_2 \wedge \langle u_1, v_1\rangle \in E_1 \Rightarrow \exists v_2 \in N_2(v_1\, b\, v_2 \wedge \langle u_2, v_2\rangle \in E_2)$, $(4)\ u_1\, b\, u_2 \wedge \langle u_2, v_2\rangle \in E_2 \Rightarrow \exists v_1 \in N_1(v_1\, b\, v_2 \wedge \langle u_1, v_1\rangle \in E_1)$. If there is a bisimulation between G_1 and G_2 we say that G_1 is *bisimilar* to G_2 ($G_1 \sim G_2$).[1] The *Graph Bisimulation problem* $GB(G_1, G_2)$ amounts to determining whether $G_1 \sim G_2$; the *Subgraph Bisimulation problem* $SB(G_1, G_2)$ consists in deciding whether there exists a subgraph G_2' of G_2 s.t. $G_1 \sim G_2'$. The problem $GB(G_1, G_2)$ ($GI(G_1, G_2)$) and the problem $SB(G_1, G_2)$ ($SI(G_1, G_2)$) are different. However, each graph isomorphism is a bisimulation. In Figure 1 a bisimulation between two non isomorphic graphs is shown.

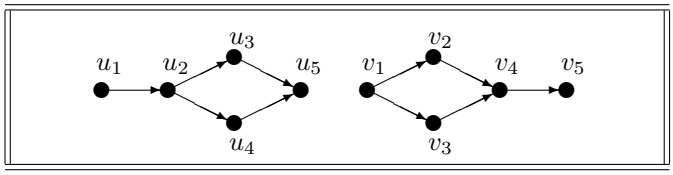

Fig. 1. $b = \{\langle u_1, v_1\rangle, \langle u_2, v_2\rangle, \langle u_2, v_3\rangle, \langle u_3, v_4\rangle, \langle u_4, v_4\rangle, \langle u_5, v_5\rangle\}$ is a bisimulation between the two graphs.

The *size* of an instance of each one of the problems is the sum of the number nodes and edges of G_1 and G_2. The polynomiality of the GB problem follows from [9], where it is shown how to find the maximum bisimulation contraction in time $O(|E|\log|N|)$.

[1] In [8] the notion is given on labeled graphs. The unlabeled notion of [1] is a particular case when all edges have the same label. Thus, NP hardness of the unlabeled version implies NP hardness of the labeled one. However, it can be shown that the labeled problem can be polynomially reduced to the unlabeled one.

III. COMPLEXITY RESULTS

$SB(G_1, G_2)$ is in NP, since $G_2' \sim G_1$ can be verified in polynomial time [9]. We prove its NP-hardness by reducing the NP-complete *Directed Hamilton Path* (HP) problem[2] ([6], p. 60) to SB. Let $n \in \mathbb{N}$; $C_n = \langle N, E\rangle$ is a *n-chain* if $N = \{x_1, \ldots, x_n\}$ and $E = \{\langle x_{i+1}, x_i\rangle : 1 \leq i < n\}$.

Lemma 1: (i) If $G = \langle N, E\rangle \sim C_n$, then $|N| \geq n$. (ii) If $G \sim C_n$ and $|N| = n$, then $G \equiv C_n$.

Proof: The property (i) can be proved by induction on $n \geq 1$.

For (ii), let $C_n = \langle\{1, \ldots, n\}, \{\langle i+1, i\rangle : 1 \leq i < n\}$ and b a bisimulation between G and C_n. We prove that b is a bijection from N to $\{1, \ldots, n\}$. (1) states that b is defined on all nodes of N. We prove that b is a function. By contradiction, let $\langle i, j\rangle$ be the minimum pair (w.r.t. lexicographic order) s.t. $\exists n \in N(y\, b\, i \wedge y\, b\, j \wedge i \neq j)$ (5). If y has no outgoing edges, by (4), i and j have no outgoing edges, i.e. $i = j = 1$: a contradiction. Otherwise, let $\langle y, z\rangle \in E$. By (3), there are i', j' nodes of C_n s.t. $\langle i, i'\rangle$ and $\langle j, j'\rangle$ are edges of C_n and $z\, b\, i'$ and $z\, b\, j'$. The form of C_n, however, implies that $i' = i - 1$ and $j' = j - 1$. Hence, $\langle i, j\rangle$ is not the minimum pair satisfying the property (5): a contradiction. Thus, b is a function, and from (2) we know that $b(N) = \{1, \ldots, n\}$, namely it is surjective. Since, by hypothesis and by the previous point, $|N| = n = |b(N)|$, b is also injective, and, thus, it is a bijection. ■

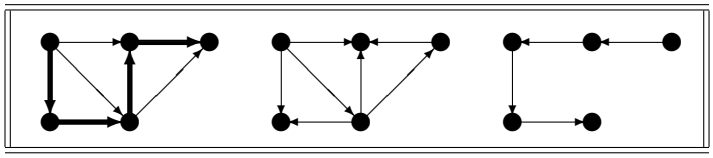

Fig. 2. From left to right, the graph G admits a Hamilton path ('thick' edges); G' does not admit Hamilton paths; and C_5 is a 5-chain (that trivially *is* a Hamilton path). Each Hamilton path on a 5-nodes graph is isomorphic (hence, bisimilar) to the 5-chain C_5.

Theorem 1: The Subgraph Bisimulation problem is NP-complete.

Proof: It remains to prove the NP-hardness of the problem. We reduce the HP problem to it. Let $G = \langle N, E\rangle$; we claim that $HP(G)$ is equivalent to $SB(C_n, G)$, with C_n a n-chain and $n = |N|$

[2] $HP(G)$ is the problem: given a graph $G = \langle N, E\rangle$, is there a path that visits each node of N exactly once?

(see also Fig. 2).

Assume that G admits the Hamilton path: $v_n \to v_{n-1} \to \cdots \to v_1$. By definition, the v_i's are pairwise distinct. Each edge occurring in the path occurs exactly once (otherwise, some node is repeated in the path). The path is a subgraph of G isomorphic (and hence bisimilar) to C_n.

Assume that there is a subgraph G' of G bisimilar to C_n. By Lemma 1(i) G' has at least n nodes; thus, being a subgraph of G, it has n nodes. By Lemma 1(ii) $G' \equiv C_n$, i.e. it is a n-chain describing a Hamilton path.

The reduction is trivially in determisistic $O(\log n)$ space. ■

Acknowledgements. We thank Elisa Quintarelli for the useful discussions concerning this work.

REFERENCES

[1] P. Aczel. *Non-well-founded sets*. Vol. 14 of *Lecture Notes, CLSI* Stanford, 1988.

[2] P. Buneman, S. B. Davidson, G. G. Hillebrand, and D. Suciu. A Query Language and Optimization Techniques for Unstructured Data. In Proc. of the 1996 ACM SIGMOD, pp. 505–516.

[3] M. P. Consens and A. O. Mendelzon. Graphlog: a Visual Formalism for Real Life Recursion. In Proc. of 9th ACM PODS'90, pp. 404–416, 1990.

[4] A. Cortesi, A. Dovier, E. Quintarelli, and L. Tanca. Operational and Abstract Semantics of the Query Language G-Log. *Theoretical Computer Science*, 275(1–2):521–560, 2002.

[5] A. Dovier and E. Quintarelli: Model-Checking Based Data Retrieval. In Proc. of *Database Programming Languages, 8th Int. Workshop*, LNCS Vol. 2397, pp. 62–77, 2002.

[6] M. R. Garey and D. S. Johnson. *Computers and Intractability – A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979.

[7] A. Lisitsa and V. Sazonov. Bounded Hyperset Theory and Web-like Data Bases. In Proc. of *5th Kurt Gödel Colloquium*, LNCS Vol. 1289, pp. 172–185, 1997.

[8] R. Milner. Operational and Algebraic Semantics of Concurrent Processes. In *Handbook of Theoretical Computer Science*, chapter 19. Elsevier Science 1990.

[9] R. Paige and R. E. Tarjan. Three partition refinements algorithms. *SIAM Journal on Computing*, 16(6):973–989, 1987.

[10] J. Paredaens, P. Peelman, and L. Tanca. G-Log: A Declarative Graphical Query Language. *IEEE Trans. on Knowledge and Data Eng.*, 7:436–453, 1995.