Partial Orders Embedding is NP-Complete

Dariusz Kalociński

Department of Logic, Institute of Philosophy, University of Warsaw
ul. Krakowskie Przedmieście 3, 00-047 Warsaw, Poland
dariusz.kalocinski@gmail.com

Abstract

Following Barwise, we consider examples of natural language sentences that seem to express that there is an embedding of one partial order into the other. We prove NP-completeness of two versions of partial orders embedding problem. We show that the task of computing the truth value of such sentences in finite models is NP-complete.

Keywords

NP-completeness, partial order, embedding, natural language, computational semantics

1. Introduction

How to recognize the truth value of natural language sentences? This question may be of interest not only for philosophers, but for computer scientists and engineers as well. We are still far away from realizing the dream of artificial intelligence capable of seamless communication with human beings. Natural language processing is a big challenge. Key questions are (a) how a machine could interpret natural language sentences and (b) compute their truth values? By an interpretation of a sentence we mean assigning a logical form to it. In this paper, we ignore (a) and simply propose some reasonable interpretations for certain natural language sentences. Having a logical form of a sentence, we may approach (b) at least in two ways. One approach is to compute the truth value of a sentence by investigating its inferential meaning, namely its consequences and their logical relations to other, already evaluated, sentences [1]. Another approach is to compute the truth value of a sentence directly in a model. This is the approach we use in this paper. Some interesting results have already been obtained by various authors, see for example [1], [2], [3], [4]. Existing work on the subject clearly indicates that recognizing the truth value of some natural language constructions in finite models is intractable. Our current work supports this view. We consider some interpretations of natural language sentences and show that the problem of recognizing their truth value in finite models is NP-complete. Along the way, we show NP-completeness of two problems concerning embedding of partial orders.

2. Variations of "The…The…" Construction

In [5] Barwise considers a version of the following natural language sentence:

The richer the country the more powerful are some of its officials. (1)

Natarajan Meghanathan et al. (Eds) : ACSIT, FCST, ITCA, SE, ICITE, SIPM, CMIT - 2014
pp. 29–35, 2014. © CS & IT-CSCP 2014 DOI : 10.5121/csit.2014.4603

He observes that (1) seems to express that there is an embedding of one order into the other. Let $\mathrm{A} = (A, >_A)$ denote the set of countries A ordered by richness $>_A$. Let $\mathrm{B} = (B, >_B)$ denote the set of officials B ordered by power $>_B$. According to Barwise, an embedding of A into B is a function $f : A \to B$ such that $\forall x, y \in A(x >_A y \Rightarrow f(x) >_B f(y))$. We use different terminology. Here, functions having the above property are called homomorphisms; embedding is an injective homomorphism that preserves order in both directions: $\forall x, y \in A(x >_A y \Leftrightarrow f(x) >_B f(y))$. Hence – in our terminology – the statement (1) seems to express the fact that there is a homomorphism from A into B.

Now, consider a slightly more complicated example:

The richer the country the more powerful are some of its officials and the more powerful are these officials the richer are countries they represent. (2)

The first conjunct of (2) is the same as (1). Thus, the logical form for (2) starts by saying that there is a function $f : A \to B$ such that $\forall x, y \in A(x >_A y \Rightarrow f(x) >_B f(y))$. If we agree that "these officials" in (2) denotes officials referred to by "some of its officials", namely to elements of the image of f, then it seems that the second conjunct of (2) adds the following condition: $\forall x, y \in A(x >_A y \Leftarrow f(x) >_B f(y))$. Hence, the logical form for (2) reads as follows: there is a function $f : A \to B$ such that $\forall x, y \in A(x >_A y \Leftrightarrow f(x) >_B f(y))$. Observe that f satisfying this condition need not be injective, if we allow the same man to be an official of two countries. However, we may force f to be injective. Consider the following example:

The smarter the student the better are some of her individual presentations and the better are these presentations the smarter are students who performed them. (3)

Here, the syntactical form is exactly the same as in (2). The only difference is that we consider different types of objects and relations. Let A stand for "⌞ is a student", P for "⌞ is an individual presentation of ⌞ ", $>$ for "⌞ is smarter than ⌞ " and ϕ for "⌞ is better than ⌞ ". Observe that since any two students have different individual presentations, any function mapping students to their individual presentations must be injective. (3) could have the following logical form:

$$\exists f \forall x, y[(A(x) \wedge A(y) \wedge x > y) \Leftrightarrow (P(f(x), x) \wedge P(f(y), y) \wedge f(x) \phi f(y))]. \quad (4)$$

We shall get back to (4) and show that recognizing the truth value of (4) in finite models, where P satisfies certain conditions, is NP-complete.

3. TECHNICAL PART

Definition 1. *Let* $\mathrm{A} = (A, >_A)$ *and* $\mathrm{B} = (B, >_B)$ *be strict partial orders. We say* A *embeds in* B *if there is an injection* $f : A \to B$ *such that*

$$\forall x, y \in A(x >_A y \Leftrightarrow f(x) >_B f(y)). \quad (5)$$

We assume basic knowledge from propositional logic. The reader need to know what are propositional formulae, valuations and satisfiability. To get familiar with the subject, see any introductory book about logic, for example [6].

Now we introduce somewhat specific notions from logic, needed in our complexity analysis. A literal is a sentential variable or a negation of a sentential variable. A clause is an alternative of literals. A propositional formula is in CNF (conjunctive normal form), if it is a conjunction of clauses. A formula is in 3CNF, if it is a conjunction of clauses each of which consists of exactly three literals.

Definition 2. STISFIABILITY OF 3CNF FORMULAE (3SAT)
Input: a formula φ *in 3CNF.*
Question: is there a valuation satisfying φ *?*

3SAT is an NP-complete problem. For an exhaustive survey of NP-completeness we refer the reader to [7].

The problem of our interest is strict partial orders embedding. We denote it shortly by SPOE.

Definition 3. STRICT PARTIAL ORDERS EMBEDDING (SPOE)

Input: strict partial orders A *and* B.
Question: is there an embedding of A *into* B *?*
Theorem 1. SPOE *is* NP-*complete.*

Proof. First, we prove that SPOE is in NP (this is an easy part). Observe that given strict partially ordered sets A and B and a function $f : A \to B$, checking whether f is an embedding of A into B is in PTIME. Now consider a non-deterministic algorithm with input consisting of strict partially ordered sets A and B. Guess a function $f : A \to B$. Finally, if f is an embedding of A into B, then accept, otherwise – reject. This clearly shows that SPOE is in NP.

We proceed to demonstration that 3SAT is polynomially reducible to SPOE. We describe an algorithm that takes an arbitrary formula φ in 3CNF as input and returns an ordered pair of strict partial orders A_φ and B_φ satisfying the following condition:

$$\forall \varphi \in \text{3CNF } (\varphi \in \text{3SAT} \Leftrightarrow \mathrm{A}_\varphi \textit{ embeds in } \mathrm{B}_\varphi). \qquad (6)$$

Let φ be an arbitrary formula in 3CNF. φ has the following form:
[
$$\varphi = (a_1 \vee b_1 \vee c_1) \wedge (a_2 \vee b_2 \vee c_2) \wedge \ldots \wedge (a_k \vee b_k \vee c_k), \qquad (7)$$

where k is a natural number of clauses in φ and a_i, b_i, c_i are literals for $i = 1,2,\ldots,k$.

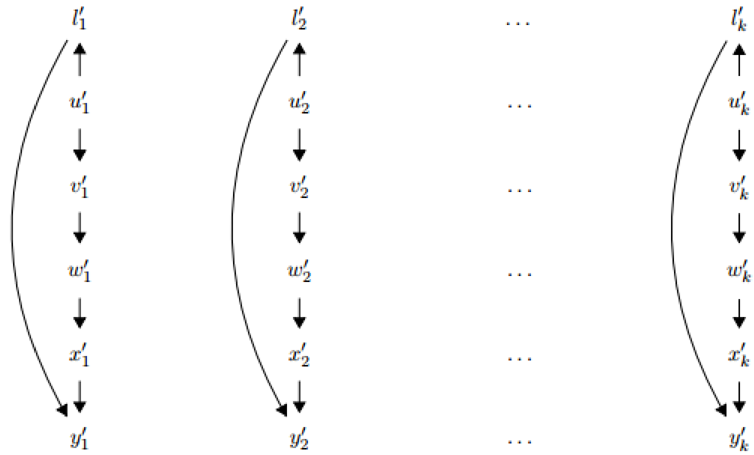

Figure 1. Construction of A_φ.

Construction of $\mathrm{A}_\varphi = (A, <_A)$. Let $A = \{l_1^{'}, l_2^{'}, \ldots, l_k^{'}\} \cup \bigcup_{i=1}^{k} \{u_i^{'}, v_i^{'}, w_i^{'}, x_i^{'}, y_i^{'}\}$. The strict partial ordering $<_A$ is the transitive closure of the relation presented in Figure 1. We adopt the convention that for any vertices s, t, the relation $s < t$ is graphically represented by an arrow from s to t. Observe that $\mathrm{A}_\varphi = (A, <_A)$ consists of k separate sub-orders. The number k is purposely the same as the number of clauses in φ.

Construction of $B_\varphi = (B, <_B)$. Let $L = \bigcup_{i=1}^{k} \{a_i, b_i, c_i\}$ be the *3k*-element set of all occurrences of literals in φ (different occurrences of the same literal are treated as different). Let $\Gamma = \bigcup_{i=1}^{k} \{u_i, v_i, w_i, x_i, y_i\}$. We put $B = L \cup \Gamma$. We proceed to the construction of $<_B$. At the beginning $<_B$ is empty. Add to $<_B$ all pairs $(s,t) \in B^2$ such that there is a solid arrow from *s* to *t*, as indicated in Figure 2. Furthermore, for every *l*, *l'* $\in L$, $l <_B l^{'}$ if and only if: *l* and *l'* does not occur in the same clause of φ and $\neg l = l^{'}$. This part of the construction is represented by dashed arrows. In this way, we obtain a relation presented in Figure 2. The desired order $<_B$ is the transitive closure of this relation.

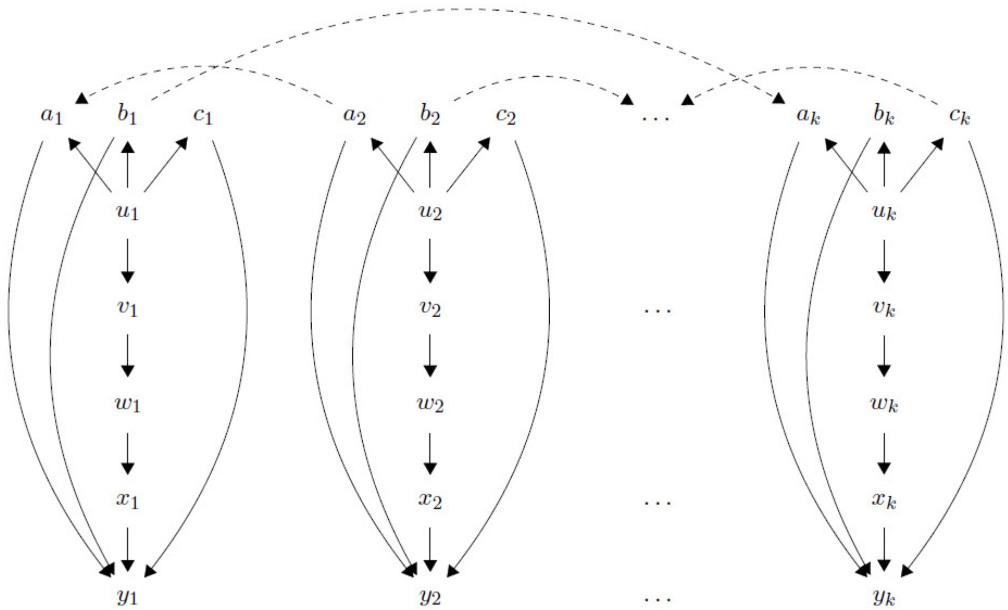

Figure 2. Construction of B_{φ}.

Proof of the equation (6). Let φ be an arbitrary formula in 3CNF with *k* clauses, *k>0*.

($\Rightarrow$) Suppose $\varphi \in$ 3SAT. Let t be a valuation satisfying φ. For each $i = 1,2,\ldots,k$ choose l_i, an occurrence of literal in the *i*-th clause, such that the value of l_i under the valuation t is 1. Let $f : A \to B$ be defined as follows: $f(l_i^{'}) = l_i$, $f(u_i^{'}) = u_i$, $f(v_i^{'}) = v_i$, $f(w_i^{'}) = w_i$, $f(x_i^{'}) = x_i$, $f(y_i^{'}) = y_i$, for $i = 1,2,\ldots,k$. We claim that f is an embedding of A_{φ} into B_{φ}. f is clearly injective. We want to show that $\forall x, y \in A(x >_A y \Leftrightarrow f(x) >_B f(y))$. This is equivalent to the condition that A_{φ} is isomorphic to $(f[A], <_B |f[A])$, where $f[A]$ denotes the image of A under the function *f* and $<_B |f[A]$ denotes the restriction of $<_B$ to the set $f[A]$. Observe that for every $1 \le i < j \le k$, neither $\neg l_i = l_j$ nor $\neg l_j = l_i$. For it were the case, then the value of l_i, (l_j) under the valuation t would be 0, which is impossible. Hence, by construction of B_{φ}, no pair of vertices $l_1, l_2, \ldots, l_k$ is joined in B_{φ} by an edge. So A_{φ} and $(f[A], <_B |f[A])$ are isomorphic.

($\Leftarrow$) Assume that A_{φ} embeds in B_{φ}. We prove that for each embedding f from A_{φ} to B_{φ} the following conditions hold:

1. For each $i = 1,2,\ldots,k$ there is $j \in \{1,2,\ldots,k\}$ such that $f(l_i^{'}) = l_j$, $f(u_i^{'}) = u_j$, $f(v_i^{'}) = v_j$, $f(w_i^{'}) = w_j$, $f(x_i^{'}) = x_j$, $f(y_i^{'}) = y_j$, where $l_j \in \{a_j, b_j, c_j\}$.

2. For all $i, j \in \{1,2,\ldots,k\}$, if $i \neq j$ then $f(u_i^{'}) \neq f(u_j^{'})$.

Let f be an arbitrary embedding from A_φ to B_φ. To prove the condition 1, note that the only paths of length four in B_φ are $u_j v_j w_j x_j y_j$, for $j = 1,2,\ldots,k$. Hence, for an arbitrary $i \in \{1,2,\ldots,k\}$, it must be that $f(u_i^{'}) = u_j$, $f(v_i^{'}) = v_j$, $f(w_i^{'}) = w_j$, $f(x_i^{'}) = x_j$, $f(y_i^{'}) = y_j$, for some $j \in \{1,2,\ldots,k\}$. Choose such a j. It remains to show that $f(l_i^{'}) \in \{a_j, b_j, c_j\}$. Suppose it is not the case. But then, by construction of B_φ, $f(l_i^{'}) <_B y_j$ does not hold. On the other hand, $l_i^{'} <_A y_i^{'}$ and since f is an embedding, we have $f(l_i^{'}) <_B f(y_i^{'}) = y_j$ which is a contradiction.

To prove the condition 2, let $i, j \in \{1,2,\ldots,k\}$ and assume $i \neq j$. For the sake contradiction, assume $f(u_i^{'}) = f(u_j^{'})$. Since $u_i^{'} <_A v_j^{'}$ does not hold, $f(u_i^{'}) <_B f(v_j^{'})$ does not hold either. However , by condition 1, $f(u_i^{'}) = f(u_j^{'}) <_B f(v_j^{'})$ which is a contradiction.

All in all, every embedding f from A_φ to B_φ chooses a set of k literals $L_f \in \{f(l_1^{'}), f(l_2^{'}),\ldots, f(l_k^{'})\}$, each literal from a different clause of φ. We claim that L_f is consistent. Observe that, for every $i, j \in \{1,2,\ldots,k\}$, $l_i^{'}$ and $l_j^{'}$ are not connected by an edge and consequently $f(l_i^{'})$ and $f(l_j^{'})$ are not connected by an edge either. This means, by construction of B_φ, that no two elements of L_f are negations of each other. Hence, they all can be made true by an appropriate valuation. This shows that $\varphi \in$ 3SAT.

Complexity. It remains to show that our construction of A_φ and B_φ from an arbitrary 3CNF formula φ is computable in polynomial time in the number of clauses in φ. The construction of the relation from Figure 1 is polynomial in the number of clauses in φ (for each clause we add six vertices and six edges). The construction of the relation from Figure 2 consists of two steps. Initially, for each clause we add eight vertices and ten edges corresponding to solid arrows. Next, we make appropriate interconnections between vertices corresponding to contradictory pairs of occurrences of literals. This can be done by searching through all pairs of occurrences of literals. Hence, the construction of the relation from Figure 2 is polynomial in the number of clauses of φ. Maybe less obvious part is the operation of transitive closure performed as a last step of the construction of A_φ and B_φ. However, given a directed graph (relations presented in Figure 1 and Figure 2 are directed graphs) one can generate its transitive closure using a Floyd-Warshall algorithm [8] which is known to work in polynomial time with respect to the number of nodes.

Definition 4. STRICT PARTIAL ORDERS EMBEDDING IN PARTITION (SPOEP)
Input: strict partial orders A *and* B, *a partition* $\{B_a\}_{a \in A}$ *such that* $\bigcup_{a \in A} B_a \subseteq B$.
Question: is there an embedding f *of* A *into* B *such that* $f(a) \in B_a$, *for every* $a \in A$ *?*

Theorem 2. SPOEP *is* NP-*complete*.

Idea of proof. We give an idea of a polynomial reduction from SPOE to SPOEP. Let (A, B) be an arbitrary instance of SPOE. We construct an instance of SPOEP $(\mathrm{A}^{'}, \mathrm{B}^{'}, \{B_a\}_{a \in A^{'}})$. Set $\mathrm{A}^{'}$ to be A. Let $\{B_a\}_{a \in A^{'}}$ be the set of $| A^{'} |$ disjoint copies of B. Let $B^{'}$ equal $\bigcup_{a \in A} B_a$. Define $<_{B^{'}}$

as follows: for every $x', y' \in B'$, $x' <_{B'} y'$ if and only if there exist $x, y \in A'$ such that x' is a copy of x, y' is a copy of y and $x <_B y$.

Theorem 3. *The problem of recognizing the truth value of* (4) *in finite models of the form* $\mathrm{M} = (U, A, B, P, >, \phi)$, *where* A, B *are unary relations,* $>, \phi$ *are strict partial orders on* A, B *respectively and* P *is a binary relation such that* $\{P_a\}_{a \in A}$ *is a disjoint family of non-empty sets, where* $P_a = \{b \in U : P(b, a)\}$, *is* NP-*complete.*

Proof. We consider only models of the form $(U, A, B, P, >, \phi)$ and satisfying the conditions stated in the theorem. We give an idea of a polynomial reduction of SPOEP to the class of finite models M that satisfy (4). The algorithm is straightforward: given an instance of SPOEP $\mathrm{X} = ((A', >_{A'}), (B', >_{B'}), \{B_a\}_{a \in A})$, construct a model $\mathrm{M}_\mathrm{X} = (U, A, B, P, >, \phi)$ in the following way: let $U = A' \cup B'$, $A = A'$, $B = B'$, $P = \bigcup_{a \in A'} \{B_a \times \{a\}\}$, $> = >_{A'}$, $\phi = \phi_{B'}$.

4. CONCLUSIONS

Assuming Edmond's Thesis and that $\mathrm{P} \neq \mathrm{NP}$, Theorem 3 indicates that the idea of artifitial intelligence capable of effective computation of truth values of some relatively simple natural language sentences in finite models is not realizable.

A somewhat specific questions about computational complexity arise when we take other types of orders into consideration, such as quasi-orders or non-strict orders. Moreover, other kinds of similarity relations between orders may be taken into account, such as homomorphisms and injective homomorphisms. These questions as well as their implications for computational semantics are being considered by the author in collaboration with M.T. Godziszewski and are going to be included in our future work.

REFERENCES

[1] M. Mostowski and D. Wojtyniak, "Computational complexity of the semantics of some natural language constructions," Annals of Pure and Applied Logic, vol. 127, no. 1-3, pp. 219–227, 2004.

[2] J. Szymanik, Quantifiers in Time and Space: Computational Complexity of Generalized Quantifiers in Natural Language. PhD thesis, University of Amsterdam, 2009.

[3] J. Szymanik, "The computational complexity of quantified reciprocals.," in TbiLLC (P. Bosch, D. Gabelaia, and J. Lang, eds.), vol. 5422 of Lecture Notes in Computer Science, pp. 139–152, Springer, 2007.

[4] M. Sevenster, Branches of imperfect information: logic, games and computation. PhD thesis, University of Amsterdam, 2006.

[5] K. J. Barwise, "On branching quantifiers in English," Journal of Philosophical Logic, vol. 8, no. 1, pp. 47–80, 1978.

[6] E. Mendelson, Introduction to Mathematical Logic. 1987.

[7] M. R. Garey and D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences). W. H. Freeman, first edition ed., 1979.

[8] T. Cormen, C. Stein, R. Rivest, and C. Leiserson, Introduction to Algorithms. McGraw-Hill Higher Education, 2nd ed., 2001.