On the Complexity of Determining Whether there is a Unique Hamiltonian Cycle or Path

Olivier Hudry
LTCI, Télécom ParisTech, Université Paris-Saclay
46 rue Barrault, 75634 Paris Cedex 13 - France
& Antoine Lobstein
Centre National de la Recherche Scientifique
Laboratoire de Recherche en Informatique, UMR 8623,
Université Paris-Sud, Université Paris-Saclay
Bâtiment 650 Ada Lovelace, 91405 Orsay Cedex - France

olivier.hudry@telecom-paristech.fr, antoine.lobstein@lri.fr

January 10, 2018

Abstract

The decision problems of the existence of a Hamiltonian cycle or of a Hamiltonian path in a given graph, and of the existence of a truth assignment satisfying a given Boolean formula $\mathcal{C}$, are well-known *NP*-complete problems. Here we study the problems of the *uniqueness* of a Hamiltonian cycle or path in an undirected, directed or oriented graph, and show that they have the same complexity, up to polynomials, as the problem U-SAT of the uniqueness of an assignment satisfying $\mathcal{C}$. As a consequence, these Hamiltonian problems are *NP*-hard and belong to the class *DP*, like U-SAT.

Key Words: Graph Theory, Hamiltonian Cycle, Hamiltonian Path, Travelling Salesman, Complexity Theory, *NP*-Hardness, Decision Problems, Polynomial Reduction, Uniqueness of Solution, Boolean Satisfiability Problems

1 Introduction

1.1 The Hamiltonian Cycle and Path Problems

We shall denote by $G = (V, E)$ a finite, simple, undirected graph with vertex set V and edge set E, where an *edge* between $x \in V$ and $y \in V$ is indifferently denoted by xy or yx. The *order* of the graph is its number of vertices, $|V|$.

If $V = \{v_1, v_2, \ldots, v_n\}$, a *Hamiltonian path* $\mathcal{HP} =< v_{i_1} v_{i_2} \ldots v_{i_n} >$ is an ordering of all the vertices in V, such that $v_{i_j} v_{i_{j+1}} \in E$ for all j, $1 \leq j \leq n-1$. The vertices v_{i_1} and v_{i_n} are called the *ends* of $\mathcal{HP}$. A *Hamiltonian cycle* is an ordering $\mathcal{HC} =< v_{i_1} v_{i_2} \ldots v_{i_n} (v_{i_1}) >$ of all the vertices in V, such that $v_{i_n} v_{i_1} \in E$ and $v_{i_j} v_{i_{j+1}} \in E$ for all j, $1 \leq j \leq n-1$. Note that the same Hamiltonian cycle admits $2n$ representations, e.g., $< v_{i_2} v_{i_3} \ldots v_{i_n} v_{i_1} (v_{i_2}) >$ or $< v_{i_n} v_{i_{n-1}} \ldots v_{i_2} v_{i_1} (v_{i_n}) >$.

A *directed graph* $H = (X, A)$ is defined by its set X of vertices and its set A of directed edges, also called *arcs*, an arc being an ordered pair (x, y) of vertices; with this respect, (x, y) and (y, x) are two different arcs and may coexist. A directed graph is said to be *oriented* if it is antisymmetric, i.e., if we have, for any pair $\{x, y\}$ of vertices, at most one of the two arcs (x, y) or (y, x); if $(x, y) \in A$, we say that y is the *out-neighbour* of x, and x is the *in-neighbour* of y, and we define the *in-degree* and *out-degree* of a vertex accordingly. The notions of *directed Hamiltonian cycle* and of *directed Hamiltonian path* are extended to a directed graph by considering the arcs $(v_{i_n}, v_{i_1}) \in A$ and $(v_{i_j}, v_{i_{j+1}}) \in A$ in the above definitions. When there is no ambiguity, we shall often drop the words "directed" and "Hamiltonian".

The following six problems (stated as one) are well known, in graph theory as well as in complexity theory:

Problem HAMC / HAMP (Hamiltonian Cycle / Hamiltonian Path):
Instance: An undirected, directed or oriented graph.
Question: Does the graph admit a Hamiltonian cycle / Hamiltonian path?

As we shall see (Proposition 2), they have been known to be *NP*-complete for a long time. In this paper, we shall be interested in the following problems, and shall locate them in the complexity classes:

Problem U-HAMC[U] (Unique Hamiltonian Cycle in an Undirected graph):
Instance: An undirected graph $G = (V, E)$.
Question: Does G admit a *unique* Hamiltonian cycle?

Problem U-HAMP[U] (Unique Hamiltonian Path in an Undirected graph):
Instance: An undirected graph $G = (V, E)$.
Question: Does G admit a *unique* Hamiltonian path?

Problem U-HAMC[D] (Unique directed Hamiltonian Cycle in a Directed graph):
Instance: A directed graph $H = (X, A)$.
Question: Does H admit a *unique* directed Hamiltonian cycle?

Problem U-HAMP[D] (Unique directed Hamiltonian Path in a Directed graph):
Instance: A directed graph $H = (X, A)$.
Question: Does H admit a *unique* directed Hamiltonian path?

Problem U-HAMC[O] (Unique directed Hamiltonian Cycle in an Oriented graph):
Instance: An oriented graph $H = (X, A)$.
Question: Does H admit a *unique* directed Hamiltonian cycle?

Problem U-HAMP[O] (Unique directed Hamiltonian Path in an Oriented graph):
Instance: An oriented graph $H = (X, A)$.
Question: Does H admit a *unique* directed Hamiltonian path?

We shall prove in Section 2 that these problems have the same complexity, up to polynomials, as the problem of the uniqueness of a truth assignment satisfying a Boolean formula (U-SAT). As a consequence, all are *NP*-hard and belong to the class *DP*. The closely related problem Unique Optimal Travelling Salesman has been investigated in [13], see Remark 8.

In a forthcoming work, we similarly reexamine some famous problems, from the viewpoint of uniqueness of solution: Boolean Satisfiability and Graph Colouring [9], Vertex Cover and Dominating Set (as well as its generalization to domination within distance r) [8], and r-Identifying Code together with r-Locating-Dominating Code [10]. We shall re-use here results from [9], and modify a construction from [8].

In the sequel, we shall need the following tools, which constitute classical definitions related to graph theory or to Boolean satisfiability. A *vertex cover* in an undirected graph G is a subset of vertices $V^* \subseteq V$ such that for every edge $e = uv \in E$, $V^* \cap \{u, v\} \neq \emptyset$. We denote by $\phi(G)$ the smallest cardinality of a vertex cover of G; any vertex cover V^* with $|V^*| = \phi(G)$ is said to be *optimal.*

Next we consider a set $\mathcal{X}$ of n *Boolean variables* x_i and a set $\mathcal{C}$ of m *clauses* ($\mathcal{C}$ is also called a *Boolean formula*); each clause c_j contains κ_j *literals*, a literal being a variable x_i or its complement $\overline{x}_i$. A *truth assignment* for $\mathcal{X}$ sets the variable x_i to TRUE, also denoted by T, and its complement to FALSE (or F), or *vice-versa.* A truth assignment is said to *satisfy* the clause c_j if c_j contains at least one true literal, and to satisfy the set of clauses $\mathcal{C}$ if every clause contains at least one true literal. The following decision problems are classical problems in complexity.

Problem VC (Vertex Cover with bounded size):
Instance: An undirected graph G and an integer k.
Question: Does G admit a vertex cover of size at most k?

Problem SAT (Satisfiability):
Instance: A set $\mathcal{X}$ of variables, a collection $\mathcal{C}$ of clauses over $\mathcal{X}$, each clause containing at least two different literals.
Question: Is there a truth assignment for $\mathcal{X}$ that satisfies $\mathcal{C}$?

The following problem is stated for any fixed integer $k \geq 2$.

Problem k-SAT (k-Satisfiability):
Instance: A set $\mathcal{X}$ of variables, a collection $\mathcal{C}$ of clauses over $\mathcal{X}$, each clause containing exactly k different literals.
Question: Is there a truth assignment for $\mathcal{X}$ that satisfies $\mathcal{C}$?

Problem 1-3-SAT (One-in-Three Satisfiability):
Instance: A set $\mathcal{X}$ of variables, a collection $\mathcal{C}$ of clauses over $\mathcal{X}$, each clause containing exactly three different literals.
Question: Is there a truth assignment for $\mathcal{X}$ such that each clause of $\mathcal{C}$ contains *exactly one* true literal?

We shall say that a clause (respectively, a set of clauses) is 1-3-*satisfied* by an assignment if this clause (respectively, every clause in the set) contains exactly one true literal. We shall also consider the following variants of the above problems:

U-VC (Unique Vertex Cover with bounded size),
U-SAT (Unique Satisfiability),
U-k-SAT (Unique k-Satisfiability),
U-1-3-SAT (Unique One-in-Three Satisfiability).

They have the same instances as VC, SAT, k-SAT and 1-3-SAT respectively, but now the question is "Is there a *unique* vertex cover / truth assignment...?".

We shall give in Propositions 3–7 what we need to know about the complexities of these problems.

1.2 Some Classes of Complexity

We refer the reader to, e.g., [1], [6], [11] or [14] for more on this topic. A *decision problem* is of the type "Given an instance I and a property $\mathcal{PR}$ on I, is $\mathcal{PR}$ true for I?", and has only two solutions, "yes" or "no". The class P will denote the set of problems which can be solved by a *polynomial* (time) algorithm, and the class NP the set of problems which can be solved by a *nondeterministic polynomial* algorithm. A *polynomial reduction* from a decision problem π_1 to a decision problem π_2 is a polynomial transformation that maps any instance of π_1 into an equivalent instance of π_2, that is, an instance of π_2 admitting the same answer as the instance of π_1; in this case, we shall write $\pi_1 \rightarrow_p \pi_2$. Cook [4] proved that there is one problem in *NP*, namely SAT, to which every other problem in *NP* can be polynomially reduced. Thus, in a sense, SAT is the "hardest" problem inside *NP*. Other problems share this property in *NP* and are called *NP-complete* problems; their class is denoted by *NP-C*. The way to show that a decision problem π is *NP*-complete is, once it is proved to be in *NP*, to choose some *NP*-complete problem π_1 and to polynomially reduce it to π. From a practical viewpoint, the *NP*-completeness of a problem π implies that we do not know any polynomial algorithm solving π, and that, under the assumption $P \neq NP$, which is widely believed to be true, no such algorithm exists: the time required can grow exponentially with the size of the instance (when the instance is a graph, its size is polynomially linked to its order; for a Boolean formula, the size is polynomially linked to, e.g., the number of variables plus the number of clauses).

The *complement* of a decision problem, "Given I and $\mathcal{PR}$, is $\mathcal{PR}$ true for I?", is "Given I and $\mathcal{PR}$, is $\mathcal{PR}$ false for I?". The class *co-NP* (respectively, *co-NP-C*) is the class of the problems which are the complement of a problem in *NP* (respectively, *NP-C*).

For problems which are not necessarily decision problems, a *Turing reduction* from a problem π_1 to a problem π_2 is an algorithm $\mathcal{A}$ that solves π_1 using a (hypothetical) subprogram $\mathcal{S}$ solving π_2 such that, if $\mathcal{S}$ were a polynomial algorithm for π_2, then $\mathcal{A}$ would be a polynomial algorithm for π_1. Thus, in this sense, π_2 is "at least as hard" as π_1. A problem π is *NP-hard* (respectively, *co-NP-hard*) if there is a Turing reduction from some *NP*-complete (respectively, co-*NP*-complete) problem to π [6, p. 113].

Remark 1 *Note that with these definitions, NP-hard and co-NP-hard coincide [6, p. 114].*

The notions of completeness and hardness can of course be extended to classes other than *NP* or co-*NP*. *NP*-hardness is defined differently in [5] and [7]: there, a problem π is *NP-hard* if there is a *polynomial* reduction from some *NP*-complete problem to π; this may lead to confusion (see Section 3).

We also introduce the classes P^{NP} (also known as Δ_2 in the hierarchy of classes) and L^{NP} (also denoted by $P^{NP[O(\log n)]}$ or Θ_2), which contain the decision problems which can be solved by applying, with a number of calls which is polynomial (respectively, logarithmic) with respect to the size of the instance, a subprogram able to solve an appropriate problem in *NP* (usually, an *NP*-complete problem); and the class *DP* [15] (or DIF^P [2] or BH_2 [11], [17], ...) as the class of languages (or problems) L such that there are two languages $L_1 \in NP$ and $L_2 \in$ co-*NP* satisfying $L = L_1 \cap L_2$. This class is not to be confused with $NP \cap$ co-*NP* (see the warning in, e.g., [14,

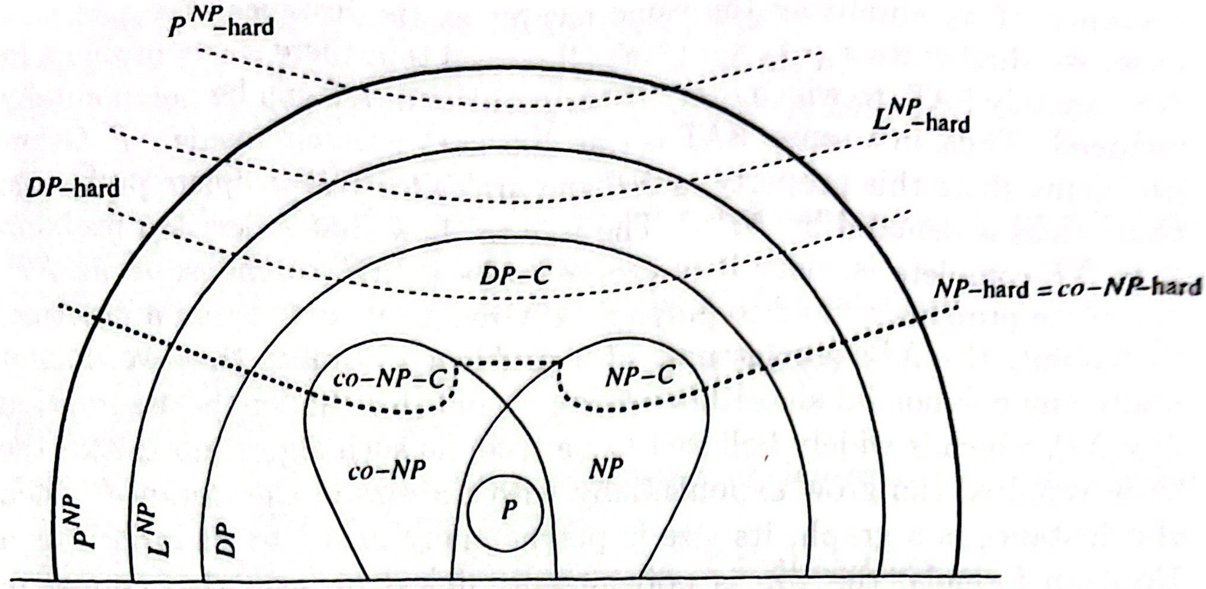

Figure 1: Some classes of complexity.

p. 412]); actually, DP contains $NP \cup$ co-NP and is contained in L^{NP}. See Figure 1.

Membership to P, NP, co-NP, DP, L^{NP} or P^{NP} gives an upper bound on the complexity of a problem (this problem is not more difficult than ...), whereas a hardness result gives a lower bound (this problem is at least as difficult as ...). Still, such results are conditional in the sense that we do not know whether or where the classes of complexity collapse.

We now consider some of the problems from Section 1.1.

Proposition 2 *[12], [6, pp. 56–60 and pp. 199-200] The decision problems HAMC and HAMP, in an undirected, directed or oriented graph, are NP-complete.* ◇

The problems VC, SAT and 3-SAT are also three of the basic and most well-known NP-complete problems [4], [6, p. 39, p. 46, p. 190 and p. 259]. More generally, k-SAT is NP-complete for $k \geq 3$ and polynomial for $k = 2$. The problem 1-3-SAT, which is obvioulsy in NP, is also NP-complete [16, Lemma 3.5], [6, p. 259], [9, Rem. 3].

The following results will be used in the sequel.

Proposition 3 *[9] For every integer $k \geq 3$, the decision problems U-SAT, U-k-SAT and U-1-3-SAT have equivalent complexity, up to polynomials.* ◇

Using the previous proposition and results from [2] and [14, p. 415], it is rather simple to obtain the following two results.

Proposition 4 *For every integer $k \geq 3$, the decision problems U-SAT, U-k-SAT and U-1-3-SAT are NP-hard (and co-NP-hard by Remark 1).* ◇

Proposition 5 *For every integer $k \geq 3$, the decision problems U-SAT, U-k-SAT and U-1-3-SAT belong to the class DP.* ◊

Remark 6 *It is not known whether these problems are DP-complete. In [14, p. 415], it is said that "U-SAT is not believed to be DP-complete".*

Proposition 7 *[8] The decision problems U-SAT and U-VC have equivalent complexity, up to polynomials. In particular, there exists a polynomial reduction from U-1-3-SAT to U-VC: U-1-3-SAT $\to_p$ U-VC.* ◊

After the following remark is made, we shall be ready to investigate the problems of uniqueness of Hamiltonian cycle or path.

Remark 8 *In [13], it is shown that the following problem is P^{NP}-complete (or Δ_2-complete).*

Problem *U-OTS (Unique Optimal Travelling Salesman):*
Instance: *A set of n vertices, a $n \times n$ symmetric matrix $[c_{ij}]$ of (nonnegative) integers giving the distance between any two vertices i and j.*
Question: *Is there a unique optimal tour, that is, a unique way of visiting every vertex exactly once and coming back, with the smallest distance sum?*

At best, a polynomial reduction from any instance $G = (V, E)$ of U-HAMC[U] to U-OTS would show that U-HAMC[U] belongs to P^{NP}, but we have a better result in Theorem 15(b), with U-HAMC[U] belonging to DP; no useful information for our Hamiltonian problems can be induced from this result on U-OTS.

2 Locating the Problems of Uniqueness

We prove that our six Hamiltonian problems have the same complexity as any of the three problems U-SAT, U-k-SAT ($k \geq 3$) and U-1-3-SAT by proving the chain of polynomial reductions given by Figure 2.

Theorem 9 *There exists a polynomial reduction from U-1-3-SAT to U-HAMP[O]: U-1-3-SAT $\to_p$ U-HAMP[O].*

Proof. We describe a polynomial reduction from the problem U-1-3-SAT to U-HAMP[O], via U-VC; it is an elaborate variation on the polynomial reduction from 3-SAT to VC in [12], [6, pp. 54–56] and the polynomial reduction from VC to HAMC[U] (see [6, pp. 56–60]). The complete proof can be found in the Appendix. ◊

Proposition 10 *There exists a polynomial reduction from U-HAMP[O] to U-HAMC[O]: U-HAMP[O] $\to_p$ U-HAMC[O].*

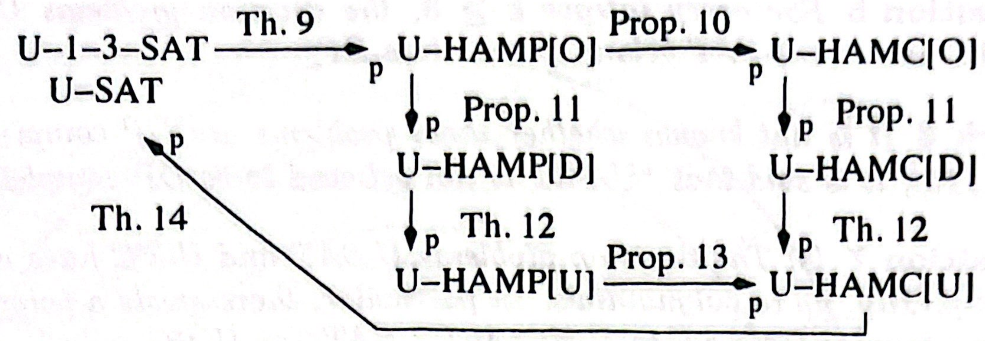

Figure 2: The chain of polynomial reductions.

Proof. We start from an oriented graph $H = (X, A)$ which is an instance of U-HAMP[O] and build a graph which is an instance of U-HAMC[O] by adding two extra vertices y, z, together with the arc (y, z) and all the arcs (x, y) and (z, x), $x \in X$. This transformation is polynomial and clearly preserves the number of solutions, in particular the uniqueness. ◊

Proposition 11 *There is a polynomial reduction from U-HAMP[O] to U-HAMP[D] and from U-HAMC[O] to U-HAMC[D]:*
U-HAMP[O] $\rightarrow_p$ *U-HAMP[D] and U-HAMC[O]* $\rightarrow_p$ *U-HAMC[D].*

Proof. It suffices to consider the identity as the polynomial reduction. ◊

Theorem 12 *There is a polynomial reduction from U-HAMP[D] to U-HAMP[U] and from U-HAMC[D] to U-HAMC[U]:*
U-HAMP[D] $\rightarrow_p$ *U-HAMP[U] and U-HAMC[D]* $\rightarrow_p$ *U-HAMC[U].*

Proof. The method is borrowed from [12].

Consider any instance of U-HAMP[D] or U-HAMC[D], i.e., a directed graph $H = (X, A)$ on n vertices. We build the undirected graph $G = (V, E)$, the instance of U-HAMP[U] or U-HAMC[U], as follows: every vertex $x \in X$ is triplicated into three vertices $x^- \in V$ (a *minus-type vertex*), $x^* \in V$ (a *star-type vertex*) and $x^+ \in V$, linked by the edges $x^-x^* \in E$ and $x^*x^+ \in E$; for every arc $(x, y) \in A$, we create the edge x^+y^- in E. The graph G thus constructed has order $3n$.

We claim that there is a unique Hamiltonian cycle (respectively, path) in G if and only if there is a unique directed Hamiltonian cycle (respectively, path) in H.

(1) Assume first that H admits a directed Hamiltonian cycle $< x_1x_2 \ldots x_n(x_1) >$. Then

$$< x_1^- x_1^* x_1^+ x_2^- x_2^* x_2^+ \ldots x_{n-1}^+ x_n^- x_n^* x_n^+ (x_1^-) >$$

is a Hamiltonian cycle in G. Moreover, two different directed Hamiltonian cycles in H provide two different Hamiltonian cycles in G.

Conversely, assume that G admits a Hamiltonian cycle $\mathcal{HC}$. This cycle must go through all the star-type vertices x^*, so it necessarily goes through all the edges x^-x^* and x^*x^+. Without loss of generality, $\mathcal{HC}$ reads:

$$\mathcal{HC} =< x_1^- x_1^* x_1^+ x_2^- x_2^* x_2^+ \dots x_{n-1}^+ x_n^- x_n^* x_n^+ (x_1^-) >; \tag{1}$$

indeed, we may assume that we "start" with the edge $x_1^- x_1^*$, then $x_1^* x_1^+$; now, because the edges which have no star-type vertex as one of their extremities are necessarily of the type x^+y^-, the other neighbour of x_1^+ is a minus-type vertex, say x_2^-; step by step, we see that $\mathcal{HC}$ has necessarily the previous form (1). Now we claim that $< x_1 x_2 \dots x_{n-1} x_n (x_1) >$ is a directed Hamiltonian cycle in H.

Indeed, for every $i \in \{1, \dots, n-1\}$, the edge $x_i^+ x_{i+1}^-$ in G implies the existence of the arc (x_i, x_{i+1}) in H; the same is true for the arc (x_n, x_1) in H, thanks to the edge $x_n^+ x_1^-$ in G. Furthermore, observe that two different Hamiltonian cycles in G provide two different directed Hamiltonian cycles in H.

So, G admits a unique Hamiltonian cycle if and only if H admits a unique directed Hamiltonian cycle.

(2) Exactly the same argument works with paths, apart from the fact that we need not consider the arc (x_n, x_1) in H, nor the edge $x_n^+ x_1^-$ in G. $\Diamond$

Proposition 13 *There exists a polynomial reduction from U-HAMP[U] to U-HAMC[U]: U-HAMP[U] $\to_p$ U-HAMC[U].*

Proof. We start from an undirected graph $G = (V, E)$ which is an instance of U-HAMP[U] and build a graph which is an instance of U-HAMC[U] by adding the extra vertex y, together with all the edges xy, $x \in V$. This transformation is polynomial and clearly preserves the number of solutions, in particular the uniqueness. $\Diamond$

Theorem 14 *There exists a polynomial reduction from U-HAMC[U] to U-SAT: U-HAMC[U] $\to_p$ U-SAT.*

Proof. We start from an instance of U-HAMC[U], an undirected graph $G = (V, E)$ with $V = \{x^1, \dots, x^{|V|}\}$; we assume that $|V| \geq 3$. We create the set of variables $\mathcal{X} = \{x_j^i : 1 \leq j \leq |V|, 1 \leq i \leq |V|\}$ and the following clauses:

(a_1) for $1 \leq i \leq |V|$, clauses of size $|V|$: $\{x_1^i, x_2^i, \dots, x_{|V|}^i\}$;
(a_2) for $1 \leq i \leq |V|$, $1 \leq j < j' \leq |V|$, clauses of size two: $\{\overline{x}_j^i, \overline{x}_{j'}^i\}$;
(b_1) for $1 \leq j \leq |V|$, clauses of size $|V|$: $\{x_j^1, x_j^2, \dots, x_j^{|V|}\}$;
(b_2) for $1 \leq i < i' \leq |V|$, $1 \leq j \leq |V|$, clauses of size two: $\{\overline{x}_j^i, \overline{x}_j^{i'}\}$;

(c) for $1 \leq i < i' \leq |V|$ such that $x^i x^{i'} \notin E$, for $1 \leq j \leq |V|$, clauses of size two: $\{\overline{x}^i_j, \overline{x}^{i'}_{j+1}\}$ and $\{\overline{x}^i_j, \overline{x}^{i'}_{j-1}\}$, with computations performed modulo $|V|$;

(d_1) $\{x^1_1\}$;

(d_2) for $2 \leq j < j' \leq |V|$, clauses of size two: $\{\overline{x}^2_{j'}, \overline{x}^3_j\}$.

Assume that we have a unique Hamiltonian cycle in G, $\mathcal{HC}_1 =< x^{p_1} x^{p_2} x^{p_3} \ldots x^{p_{|V|-1}} x^{p_{|V|}} (x^{p_1}) >$. Note that for the time being, we could also write $\mathcal{HC}_1 =< x^{p_1} x^{p_{|V|}} x^{p_{|V|-1}} \ldots x^{p_3} x^{p_2} (x^{p_1}) >$, or "start" on a vertex other than x^{p_1}, cf. Introduction. This is why, without loss of generality, we set $p_1 = 1$, i.e., we "fix" the first vertex, and we also choose the "direction" of the cycle, by deciding, e.g., that x^2 appears "before" x^3 in the cycle —cf. (d_1)-(d_2). Define the assignment $\mathcal{A}_1$ by $\mathcal{A}_1(x^{p_q}_q) =$T for $1 \leq q \leq |V|$, and all the other variables are set FALSE by $\mathcal{A}_1$. We claim that $\mathcal{A}_1$ satisfies all the clauses.

(a_1) for $1 \leq i \leq |V|$, if the vertex x^i has position j in the cycle, then the variable x^i_j satisfies the clause; (a_2) if $\{\overline{x}^i_j, \overline{x}^i_{j'}\}$ is not satisfied by $\mathcal{A}_1$ for some i, j, j', then $\mathcal{A}_1(x^i_j) = \mathcal{A}_1(x^i_{j'}) =$T, which means that the vertex x^i appears at least twice in the cycle;

(b_1) for $1 \leq j \leq |V|$, if the position j is occupied by the vertex x^i, then the variable x^i_j satisfies the clause; (b_2) if $\{\overline{x}^i_j, \overline{x}^{i'}_j\}$ is not satisfied by $\mathcal{A}_1$ for some i, i', j, then two different vertices are the j-th vertex in the cycle.

(c) If one of the two clauses is not satisfied, say the first one, then the positions j and $j+1$ in the cycle are occupied by two vertices not linked by any edge in G.

(d_1) $\{x^1_1\}$ is satisfied by $\mathcal{A}_1$ thanks to the assumption on the first vertex of the cycle; (d_2) if for some $j < j'$, the clause $\{\overline{x}^2_{j'}, \overline{x}^3_j\}$ is not satisfied, then the vertex x^3 occupies a position j smaller than the position j' of x^2, which contradicts our assumption on x^2 and x^3.

Is $\mathcal{A}_1$ unique? Assume on the contrary that another assignment, $\mathcal{A}_2$, also satisfies the constructed instance of U-SAT. Then by (a_1) and (a_2), for every $i \in \{1, \ldots, |V|\}$, there is at least, then at most, one $j = j(i)$ such that $\mathcal{A}_2(x^i_j) =$T; by (b_1) and (b_2), for every $j \in \{1, \ldots, |V|\}$, there is at least, then at most, one $i = i(j)$ such that $\mathcal{A}_2(x^i_j) =$T; so we have "a place for everything and everything in its place", with exactly $|V|$ variables which are TRUE by $\mathcal{A}_2$ and an ordering of the vertices according to the one-to-one correspondence given by $\mathcal{A}_2$: the vertex x^i is in position j if and only if $\mathcal{A}_2(x^i_j) =$T. Next, thanks to the clauses (c), two vertices following each other in this ordering, including the last and first ones, are necessarily neighbours, so that this ordering is a Hamiltonian cycle, $\mathcal{HC}_2$. Since we have assumed the uniqueness of the Hamiltonain cycle $\mathcal{HC}_1$ in G, the two cycles can differ only by their starting points or their "directions". However these differences are ruled out by the clauses (d_1) and (d_2), so that the two cycles coincide vertex to vertex, and $\mathcal{A}_1 = \mathcal{A}_2$. So a YES answer for U-HAMC[U] leads to a YES answer for U-SAT.

Assume now that the answer to U-HAMC[U] is negative. If it is negative because there are at least two Hamiltonian cycles, then we have at least two assignments satisfying the instance of U-SAT: we have seen above how to construct a suitable assignment from a cycle, and different cycles obviously lead to different assignments. If there is no Hamiltonian cycle, then there is no assignment satisfying U-SAT, because such an assignment would give a cycle, as we have seen above with $\mathcal{A}_2$. So in both cases, a NO answer to U-HAMC[U] implies a NO answer to U-SAT. ◊

Gathering all our previous results, we obtain the following theorem.

Theorem 15 *For every integer $k \geq 3$, the decision problems U-SAT, U-k-SAT and U-1-3-SAT have the same complexity as U-HAMP[U], U-HAMC[U], U-HAMP[O], U-HAMC[O], U-HAMP[D], and U-HAMC[D], up to polynomials. Therefore,*

(a) the decision problems U-HAMP[U], U-HAMC[U], U-HAMP[O], U-HAMC[O], U-HAMP[D], and U-HAMC[D] are NP-hard (and co-NP-hard by Remark 1);

(b) the decision problems U-HAMP[U], U-HAMC[U], U-HAMP[O], U-HAMC[O], U-HAMP[D], and U-HAMC[D] belong to the class DP. ◊

Note that the membership to *DP* could have been proved directly.

3 Conclusion

By Theorem 15, for every integer $k \geq 3$, the three decision problems U-SAT, U-k-SAT, U-1-3-SAT have the same complexity, up to polynomials, as the problem of the uniqueness of a path or of a cycle in a graph, undirected, directed, or oriented; all are *NP*-hard (and co-*NP*-hard by Remark 1) and belong to the class *DP*, and it is thought that they are not *DP*-complete. Anyway, they can be found somewhere in the hatched area of Figure 3.

Open problem. Find a better location for any of these problems inside the hierarchy of complexity classes.

In [2], the authors wonder whether

(A) U-SAT is *NP*-hard, but here we believe that what they mean is: does there exist a *polynomial* reduction from an *NP*-complete problem to U-SAT? i.e., they use the *second* definition of *NP*-hardness;

finally, they show that (A) is true if and only if

(B) U-SAT is *DP*-complete.

So, if one is careless and considers that U-SAT is *NP*-hard without checking according to which definition, one might easily jump too hastily to the

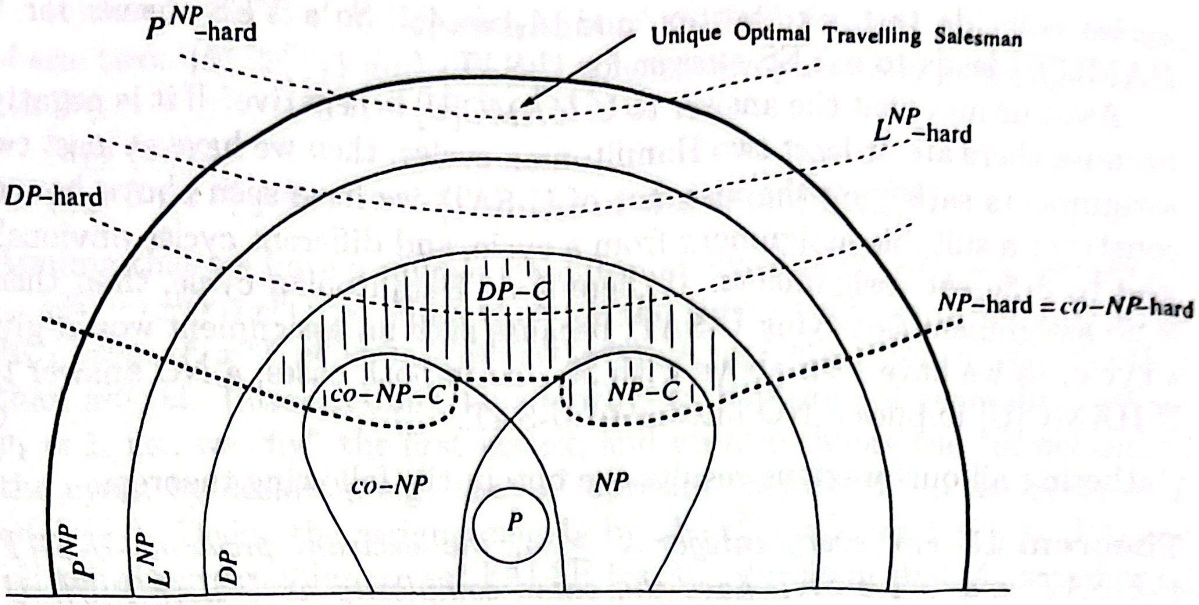

Figure 3: Some classes of complexity: Figure 1 re-visited.

conclusion that U-SAT is *DP*-complete, which, to our knowledge, is not known to be true or not. As for U-3-SAT, we do not know where to locate it more precisely either; in [3] the problems U-k-SAT and more particularly U-3-SAT are studied, but it appears that they are versions where the given set of clauses has zero or one solution, which makes quite a difference with our problem.

Appendix: the Proof of Theorem 9

A) From U-1-3-SAT to U-VC

From an arbitrary instance of U-1-3-SAT with m clauses and n variables, we mimick the reduction from 3-SAT to VC in [12], [6, pp. 54–56], and we construct the instance $G_{VC} = (V_{VC}, E_{VC})$ of U-VC as follows (see Figure 4 for an example): we construct for each clause c_j a triangle $T_j = \{a_j, b_j, d_j\}$, and for each variable x_i a component $G_i = (V_i = \{x_i, \overline{x}_i\}, E_i = \{x_i\overline{x}_i\})$. Then we link the components G_i on the one hand, and the triangles T_j on the other hand, according to which literals appear in which clauses ("membership edges"). For each clause $c_j = \{\ell_1, \ell_2, \ell_3\}$, we also add the triangular set of edges $E'_j = \{\overline{\ell}_1\overline{\ell}_2, \overline{\ell}_1\overline{\ell}_3, \overline{\ell}_2\overline{\ell}_3\}$. Finally, we set $k = n + 2m$.

The order of G_{VC} is $3m + 2n$ and its number of edges is at most $n + 9m$ (the edge sets E'_j are not necessarily disjoint).

Note already that if V^* is a vertex cover, then each triangle T_j contains at least two vertices, each component G_i at least one vertex, and $|V^*| \geq 2m + n = k$; if $|V^*| = 2m + n$, then each triangle contains exactly two vertices, and each component G_i exactly one vertex. We can also observe that, because of the edge sets E'_j, at least two vertices among $\overline{\ell}_1, \overline{\ell}_2, \overline{\ell}_3$ belong to any vertex cover.

(a) Let us first assume that the answer to U-1-3-SAT is YES: there is a unique

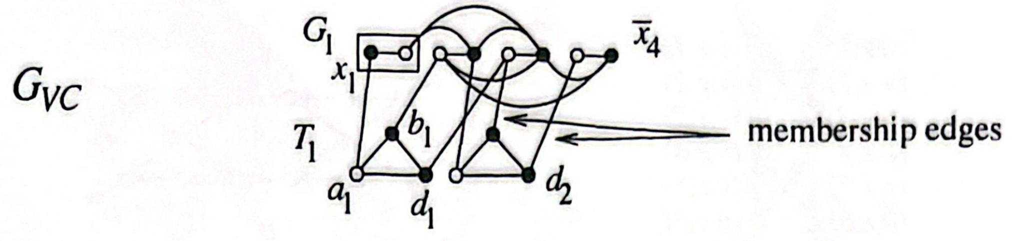

Figure 4: Illustration of the undirected graph constructed for the reduction from U-1-3-SAT to U-VC, with four variables and two clauses, $c_1 = \{x_1, x_2, x_3\}$, $c_2 = \{\overline{x}_2, x_3, x_4\}$. Here, $k = 8$, and the black vertices form the (not unique) vertex cover V^* of size eight corresponding to the (not unique) truth assignment $x_1 = $T, $x_2 = $F, $x_3 = $F, $x_4 = $F 1-3-satisfying the clauses. As soon as we set $V^* \cap (V_1 \cup V_2 \cup V_3 \cup V_4) = \{x_1, \overline{x}_2, \overline{x}_3, \overline{x}_4\}$, the other vertices in V^* are forced.

truth assignment 1-3-satisfying the clauses of $\mathcal{C}$. Then, by taking, in each G_i, the vertex corresponding to the literal which is TRUE, and in every triangle T_j, the two vertices which are linked to the two false literals of c_j, we obtain a vertex cover V^* whose size is equal to k. Moreover, once we have put the n vertices corresponding to the true literals in the vertex cover V^* in construction, we have *no choice* for the completion of V^* with $k - n = 2m$ vertices: when we take two vertices in T_j, we *must* take the two vertices which cover the membership edges linked to the two false literals (in the example of Figure 4, the vertices b_1, d_1 and b_2, d_2). So, if another vertex cover V^+ of size k exists, it must have a different distribution of its vertices over the components G_i, still with exactly one vertex in each G_i; this in turn defines a valid truth assignment, by setting $x_i = $T if $x_i \in V^+$, $x_i = $F if $\overline{x}_i \in V^+$. Now this assignment 1-3-satisfies $\mathcal{C}$, thanks in particular to our observation on the covering of the edges in E'_j. So we have two truth assignments 1-3-satisfying $\mathcal{C}$, contradicting the YES answer to U-1-3-SAT; therefore, V^* is the only vertex cover of size k.

(b) Assume next that the answer to U-1-3-SAT is NO: this may be either because no truth assignment 1-3-satisfies the instance, or because at least two assignments do; in the latter case, this would lead, using the same argument as in the previous paragraph, to at least two vertex covers of size k, and a NO answer to U-VC. So we are left with the case when the set of clauses $\mathcal{C}$ cannot be 1-3-satisfied. But again, we have already seen that this would imply that no vertex cover of size (at most) k exists, since such a hypothetical vertex cover V^+ would imply the existence of a suitable assignment.

We are now ready to construct an instance of U-HAMP[O]. In the sequel, we shall say "path" for "directed Hamiltonian path".

B) Construction of the Instance of U-HAMP[O]

We look deeper into the proof of the *NP*-completeness of the problem Hamiltonian Cycle (see [6, pp. 56–60]), which uses a polynomial reduction from VC to HAMC[U] that, due to the so-called "selector vertices", cannot cope with the

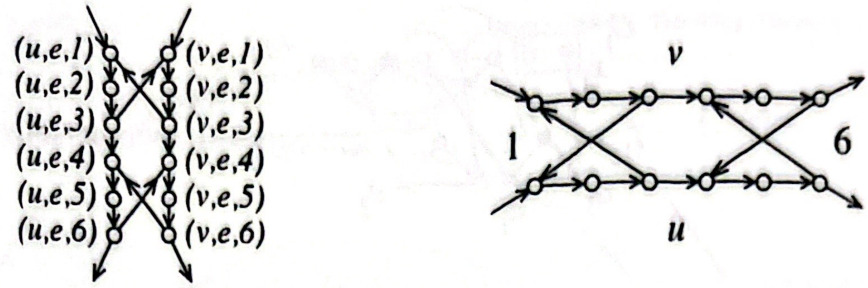

Figure 5: Two possible representations of the same component H_e for the edge $e = uv \in E_{VC}$ (Step 1).

problem of uniqueness; step by step, we construct an oriented graph $H = (X, A)$ for which we will prove that:

(i) if there is a YES answer for the instance of U-1-3-SAT (which implies that there is a unique vertex cover V^* in G_{VC}, with cardinality at most k), then there is a unique path in H;

(ii) if there are at least two assignments 1-3-satisfying all the clauses (i.e., there are at least two vertex covers in G_{VC}, with cardinality at most k), then there are at least two paths in H;

(iii) if there is no assignment 1-3-satisfying the clauses (and no vertex cover in G_{VC} with cardinality at most k), then there is no path in H.

Step 1. For each edge $e = uv \in E_{VC}$, we build one component $H_e = (X_e, A_e)$ with 12 vertices and 14 arcs: $X_e = \{(u,e,i), (v,e,i) : 1 \le i \le 6\}$, $A_e = \{((u,e,i),(u,e,i+1)), ((v,e,i),(v,e,i+1)) : 1 \le i \le 5\} \cup \{((v,e,3),(u,e,1)), ((u,e,3),(v,e,1))\} \cup \{((v,e,6),(u,e,4)), ((u,e,6),(v,e,4))\}$; see Figure 5, which is the oriented copy of Figure 3.4 in [6, p. 57].

In the completed construction, the only vertices from this component that will be involved in any additional arcs are the vertices $(u,e,1)$, $(u,e,6)$, $(v,e,1)$, and $(v,e,6)$. This, together with the fact that there will be two particular vertices, α_1 and δ, which will necessarily be the ends of any path, will imply that any path in the final graph H will have to meet the vertices in X_e in exactly one of the three configurations shown in Figure 6, which is the oriented copy of Figure 3.5, in [6, p. 58]. Thus, when the path meets the component H_e at $(u,e,1)$, it will have to leave at $(u,e,6)$ and go through either (a) all 12 vertices in the component, in which case we shall say that the component is *completely visited from the u-side*, or (b) only the 6 vertices (u,e,i), $1 \le i \le 6$, in which case we shall say that the component is visited *in parallel* and needs two visits, i.e., another section of the path will re-visit the component, meeting the 6 vertices (v,e,i), $1 \le i \le 6$.

Step 2. We create n vertices α_i, $1 \le i \le n$, and $2n$ arcs $(\alpha_i, (x_i, x_i\overline{x}_i, 1))$, $(\alpha_i, (\overline{x}_i, x_i\overline{x}_i, 1))$, that is, we link α_i to the "first" vertices of the component H_e whenever $e = x_i\overline{x}_i$. The vertices α_i can be seen as literal selectors that will choose between x_i and $\overline{x}_i$. The vertex α_1 will have no other neighbours; this means in particular that it will have no in-neighbours, thus it will necessarily be the starting vertex of any Hamiltonian path, if such a path exists.

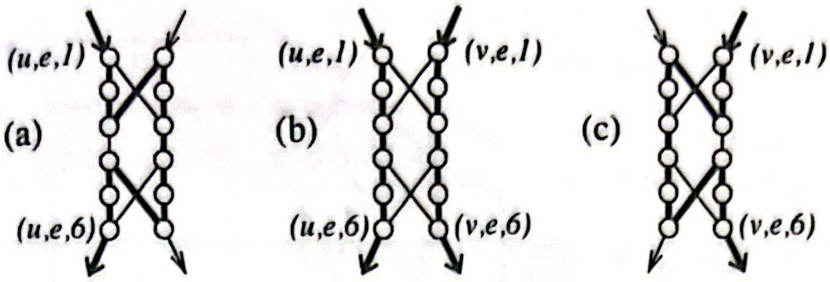

Figure 6: The three ways of going through the component H_e (Step 1). The arrows inside H_e are not represented.

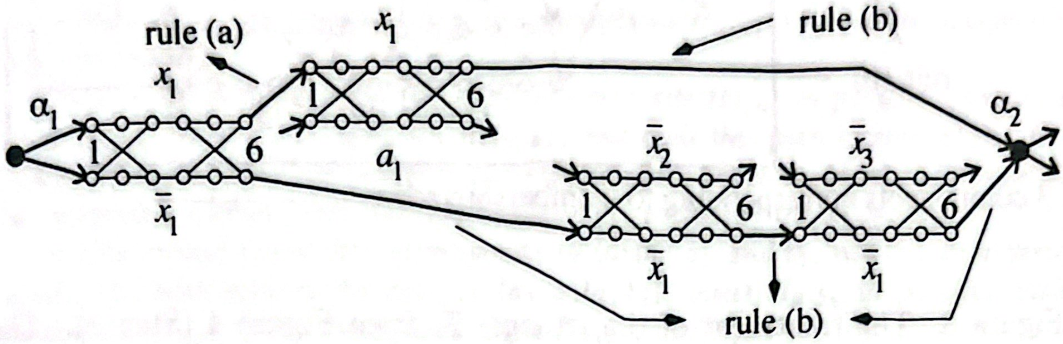

Figure 7: The example of the literals x_1 and $\overline{x}_1$ from Figure 4 (Step 2); here, Rule (a) applies with $q(x_1) = 1$, $q(\overline{x}_1) = 0$, Rule (b) with $s(\overline{x}_1) = 2$. The arrows inside H_e are not represented.

We choose an arbitrary order on the $3m$ vertices of the triangles T_j in the graph G_{VC}, say $\mathcal{O}_T =< a_1, b_1, d_1, a_2, \ldots, d_m >$ and an arbitrary order on the literals $x_i, \overline{x}_i$, say $\mathcal{O}_\ell =< x_1, x_2, \ldots, x_n, \overline{x}_1, \ldots, \overline{x}_n >$. For each literal ℓ_i equal to x_i or $\overline{x}_i$, we do the following (see Figure 7 for an example):

Rule (a): If ℓ_i appears $q = q(\ell_i) \geq 0$ times in the clauses and is linked in G_{VC} to $t_1, \ldots, t_q$ where the t's belong to the triangles T_j and follow the order $\mathcal{O}_T$, then we create the arcs $((\ell_i, \ell_i\overline{\ell}_i, 6), (\ell_i, \ell_i t_1, 1))$, $((\ell_i, \ell_i t_1, 6), (\ell_i, \ell_i t_2, 1))$, $\ldots$, $((\ell_i, \ell_i t_{q-1}, 6), (\ell_i, \ell_i t_q, 1))$.

Rule (b): We consider the triangular sets of edges E'_j described in the construction of G_{VC}.

• If ℓ_i does not belong to any such edge, we create the arc $((\ell_i, \ell_i t_q, 6), \alpha_{i+1})$ —or $((\ell_i, \ell_i\overline{\ell}_i, 6), \alpha_{i+1})$ if ℓ_i does not apppear in any clause— unless $i = n$, in which case we create $((\ell_i, \ell_i t_q, 6), \beta_1)$ or $((\ell_i, \ell_i\overline{\ell}_i, 6), \beta_1)$, where β_1 is a new vertex that will be spoken of at the beginning of Step 3.

• If ℓ_i belongs to $s = s(\ell_i) > 0$ edges from E'_j, which link ℓ_i to s literals $\ell_{i_1}, \ldots, \ell_{i_s}$ that follow the order $\mathcal{O}_\ell$, then we build the arc $((\ell_i, \ell_i t_q, 6), (\ell_i, \ell_i\ell_{i_1}, 1))$ —or the arc $((\ell_i, \ell_i\overline{\ell}_i, 6), (\ell_i, \ell_i\ell_{i_1}, 1))$ if $q = 0$; next, the arcs $((\ell_i, \ell_i\ell_{i_1}, 6), (\ell_i, \ell_i\ell_{i_2}, 1))$, $\ldots$, $((\ell_i, \ell_i\ell_{i_{s-1}}, 6), (\ell_i, \ell_i\ell_{i_s}, 1))$ and $((\ell_i, \ell_i\ell_{i_s}, 6), \alpha_{i+1})$, unless $i = n$,

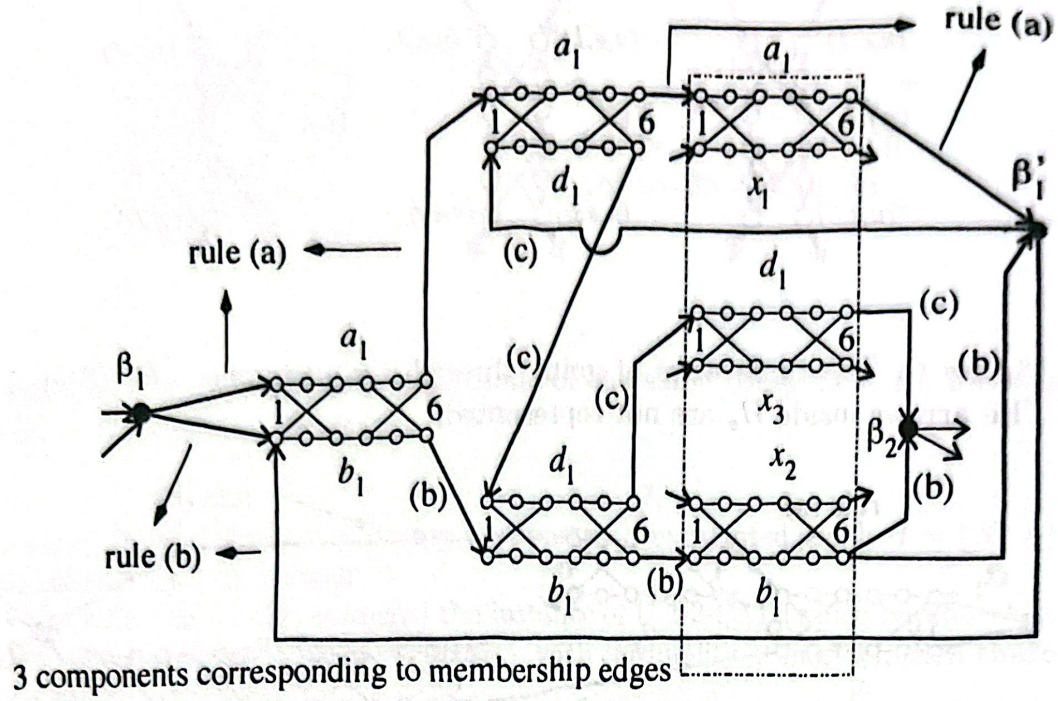

Figure 8: The treatment of the triangle T_1 from Figure 4 (Step 3). The arrows inside H_e are not represented.

in which case we create $((\ell_i, \ell_i\ell_{i_s}, 6), \beta_1)$.

Remark 16 *In the example of Figure 7, one can see that if a path takes, e.g., the arc* $(\alpha_1, (x_1, x_1\overline{x}_1, 1))$*, then it visits the vertices* $(x_1, x_1\overline{x}_1, 6)$, $(x_1, x_1 a_1, 1)$, $(x_1, x_1 a_1, 6)$, *and* α_2. *If on the other hand, we use the arc* $(\alpha_1, (\overline{x}_1, x_1\overline{x}_1, 1))$*, we also go to* α_2. *The same is true between* α_2 *and* α_3, ..., *between* α_{n-1} *and* α_n, *between* α_n *and* β_1.

We can see that so far, α_1 has (out-)degree 2, α_2, ..., α_n have degree 4 (in- and out-degrees equal to 2), and β_1 has (in-)degree 2.

Step 3. We consider the m clauses and the m corresponding triangles T_j.

We create $2m$ vertices β_j, β'_j, $1 \leq j \leq m$. As we have seen in the previous step, β_1 has already two in-neighbours, which can be $(\ell_n, \ell_n t_q, 6)$, or $(\ell_n, \ell_n\overline{\ell}_n, 6)$, or $(\ell_n, \ell_n\ell_{n_s}, 6)$. We also create one more vertex δ, which will have only in-neighbours, so that α_1 and δ will necessarily be the ends of any directed Hamiltonian path, if such a path exists.

Now for the triangle $T_j = \{a_j, b_j, d_j\}$, $1 \leq j \leq m$, associated to the clause $c_j = \{\ell_{j_1}, \ell_{j_2}, \ell_{j_3}\}$ in the graph G_{VC}, we consider the six corresponding components $H_{a_j b_j}$, $H_{a_j d_j}$, $H_{b_j d_j}$, $H_{a_j \ell_{j_1}}$, $H_{b_j \ell_{j_2}}$ and $H_{d_j \ell_{j_3}}$. The vertices β_j and β'_j can be seen as triangle selectors, intended to choose two vertices among three. With this in mind, we create the following arcs (see Figure 8), for $j \in \{1, 2, \ldots, m\}$:

Rule (a): $(\beta_j, (a_j, a_j b_j, 1))$, $((a_j, a_j b_j, 6), (a_j, a_j d_j, 1))$, $((a_j, a_j d_j, 6), (a_j, a_j \ell_{j_1}, 1))$, $((a_j, a_j \ell_{j_1}, 6), \beta'_j)$.

Rule (b): $(\beta_j, (b_j, a_jb_j, 1))$, $(\beta'_j, (b_j, a_jb_j, 1))$, $((b_j, a_jb_j, 6), (b_j, b_jd_j, 1))$, $((b_j, b_jd_j, 6), (b_j, b_j\ell_{j_2}, 1))$, $((b_j, b_j\ell_{j_2}, 6), \beta'_j)$, plus the arc $((b_j, b_j\ell_{j_2}, 6), \beta_{j+1})$, unless $j = m$, in which case it is $((b_j, b_j\ell_{j_2}, 6), \delta)$.

Rule (c): $(\beta'_j, (d_j, a_jd_j, 1))$, $((d_j, a_jd_j, 6), (d_j, b_jd_j, 1))$, $((d_j, b_jd_j, 6), (d_j, d_j\ell_{j_3}, 1))$, $((d_j, d_j\ell_{j_3}, 6), \beta_{j+1})$, unless $j = m$, in which case it is $((d_j, d_j\ell_{j_3}, 6), \delta)$.

Remark 17 *In the example of Figure 8, there are three ways for going from β_1 to β_2 through the components $H_{a_1b_1}$, $H_{a_1d_1}$ and $H_{d_1b_1}$.*

- *If a path starts by taking the arc $(\beta_1, (a_1, a_1b_1, 1))$, then there are two possibilities, according to how we visit $H_{a_1b_1}$:*

  ◦ *The first possibility corresponds to taking a_1 and d_1, not b_1, in a vertex cover: the path completely visits the component $H_{a_1b_1}$ from the a_1-side, then the component $H_{a_1d_1}$ in parallel, then the component $H_{a_1x_1}$ in a so far unspecified way, then β'_1.*

  Next, it takes the arc $(\beta'_1, (d_1, d_1a_1, 1))$, re-visits $H_{d_1a_1}$ in parallel, completely visits $H_{d_1b_1}$ from the d_1-side, then $H_{d_1x_3}$, and ends this path section at β_2. One can see that the three components corresponding to edges incident to b_1 must all be completely visited from the side opposite b_1, including the x_2-side.

  ◦ *The second possibility corresponds to taking a_1 and b_1, not d_1, in a vertex cover: the path follows the arc $(\beta_1, (a_1, a_1b_1, 1))$, visits $H_{a_1b_1}$ in parallel, visits completely $H_{a_1d_1}$ from the a_1-side, then $H_{a_1x_1}$, and β'_1.*

  Next, it takes the arc $(\beta'_1, (b_1, b_1a_1, 1))$, goes through $H_{b_1a_1}$ in parallel, goes completely through $H_{b_1d_1}$ from the b_1-side, then $H_{b_1x_2}$, and ends this path section at β_2. The component $H_{d_1x_3}$ is not yet visited.

- *Alternatively, a path can start by taking the arc $(\beta_1, (b_1, a_1b_1, 1))$; this corresponds to taking b_1 and d_1, not a_1, in a vertex cover and constitutes the third way for going from β_1 to β_2. The path then completely visits $H_{b_1a_1}$ from the b_1-side, $H_{b_1d_1}$ in parallel, $H_{b_1x_2}$, and β'_1.*

  Next, it completely visits $H_{d_1a_1}$ from the d_1-side, $H_{d_1b_1}$ in parallel, $H_{d_1x_3}$, and this path section ends at β_2. The component $H_{a_1x_1}$ is not yet visited.

It is easy to see that these are the only *three ways for going from β_1 to β_2 through the components $H_{a_1b_1}$, $H_{a_1d_1}$ and $H_{d_1b_1}$, not taking into account the ways of going through the components $H_{a_1x_1}$, $H_{b_1x_2}$ and $H_{d_1x_3}$ (this issue will be treated later on, in the general case): indeed, the only possibility left would be to follow the arc $(\beta_1, (b_1, a_1b_1, 1))$ and visit $H_{b_1a_1}$ in parallel, but then the a_1-side of $H_{b_1a_1}$ cannot be reached.*

The same will be true for the components $H_{a_jb_j}$, $H_{a_jd_j}$ and $H_{d_jb_j}$ and the corresponding triangles T_j, $1 \le j \le m$, between β_j and β_{j+1} (or between β_m and δ).

The description of the oriented graph H is complete. Now β_1 has increased its degree to 4, and β_2, ..., β_m and β'_1, ..., β'_m have degree 4. *Actually, all the selectors but α_1 have in-degree 2 and out-degree 2 in H.* These n selectors α_i, $1 \le i \le n$, and $2m$ selectors β_j, β'_j, $1 \le j \le m$, translate the choices we have to make when constructing a vertex cover with size $2m + n$: we have one choice among the n variables (take x_i or $\overline{x}_i$); as for the m triangles T_j associated to the clauses, Remark 17 has shown how the selectors β_j, β'_j, $1 \le j \le m$, can be used to choose two vertices among three. The *number* of selectors is one reason why there is no directed Hamiltonian path in H when the vertex covers in G_{VC} have size at least $2m+n+1$.

The order of H is $12|E_{VC}|+n+2m+1$, which is at most $12(n+9m)+n+2m+1$, so that the transformation is polynomial indeed.

We are now going to prove our claims about the existence or non-existence, uniqueness or non-uniqueness, of a directed Hamiltonian path in H.

C) How it Works

Assume first that there is an assignment satisfying the instance of U-1-3-SAT, and therefore that there is a vertex cover V^* in G_{VC} with size $2m+n$. We construct a path in H in a straightforward way: every component H_{uv} ($uv \in E_{VC}$) with $\{u,v\} \subset V^*$ is visited in parallel, whereas H_{uv} is completely visited from the u-side whenever $u \in V^*$, $v \notin V^*$. Let us have a closer look at how this works:

We start at α_1, and visit completely the component $H_{x_1\overline{x}_1}$ from the x_1-side if $x_1 =$T, from the $\overline{x}_1$-side if $x_1 =$F (or, equivalently, if $x_1 \in V^*$ or $\overline{x}_1 \in V^*$, respectively). If, say, $x_1 =$F, we then completely go through all the components corresponding to triangles T_j and involving $\overline{x}_1$, all from the $\overline{x}_1$-side; note that all the components just completely visited involve $\overline{x}_1$ and a vertex not in V^*, by the very construction of the vertex cover V^*, which is possible because it stems from an assignment 1-3-satisfying all the clauses. Then we go through the components constructed from the edge sets E'_j and involving $\overline{x}_1$; those involving a second vertex in V^* (i.e., a true literal) are visited in parallel, whereas those involving a vertex not in V^* are completely visited from the $\overline{x}_1$-side; then the path arrives at α_2. The components involving x_1, apart from $H_{x_1\overline{x}_1}$, remain completely unvisited for the time being, and the components that have been visited in parallel will have to be re-visited.

We act similarly between α_2 and α_3, ..., α_n and β_1; cf. Remark 16. When doing this, we re-visit all the components that had been visited only in parallel, and completely visit the components involving a literal not in V^* and corresponding to edges in E'_j. The only components not visited yet between α_1 and β_1 are those corresponding to edges between a false literal (not in V^*) and its neighbours in the triangles T_j.

Next, starting from β_1, we use Remark 17 according to the three possible cases: (a) $\{a_1,d_1\} \subset V^*$, $b_1 \notin V^*$, (b) $\{a_1,b_1\} \subset V^*$, $d_1 \notin V^*$, (c) $\{b_1,d_1\} \subset V^*$, $a_1 \notin V^*$. We give in detail only the third case, for the clause $c_1 = \{\ell_1,\ell_2,\ell_3\}$: we use the arc $(\beta_1,(b_1,a_1b_1,1))$ and completely visit the component $H_{a_1b_1}$ from the b_1-side, then the component $H_{b_1d_1}$ in parallel, then the complete component $H_{b_1\ell_2}$ from the b_1-side (because if $b_1 \in V^*$, then $\ell_2 \notin V^*$ and this component had not yet been visited) and end at β'_1. Next, we take the arc $(\beta'_1,(d_1,d_1a_1,1))$, we completely visit $H_{d_1a_1}$ from the d_1-side, re-visit $H_{d_1b_1}$ in parallel, completely visit $H_{d_1\ell_3}$ from the d_1-side, and this path section ends at β_2. Note that (a) the three components involving a_1 between β_1 and β_2 have been completely visited, from the b_1-, d_1- or ℓ_1-sides (because $a_1 \notin V^*$ implies that $\ell_1 \in V^*$); (b) any so far unvisited component involving a false literal (here, these are ℓ_2 and ℓ_3) and one of the vertices of the triangle T_1 (here b_1 and d_1) has now been completely visited from the triangle sides (here from the b_1- and d_1-sides).

We act similarly between β_2 and β_3, ..., β_m and δ; cf. the end of Remark 17. The ultimate section takes us between β_m and δ, the final vertex, and we have indeed built a directed Hamiltonian path, from α_1 to δ, in the oriented graph H.

Obviously, two different assignments 1-3-satisfying all the clauses lead, following the above process, to two different paths in H. We still want to prove that 1) if no assignment 1-3-satisfying all the clauses exists, then no path exists, and 2) a unique assignment 1-3-satisfying all the clauses leads to a unique path.

1) We assume that there is a directed Hamiltonian path $\mathcal{HP}$ in H, and exhibit an assignment 1-3-satisfying all the clauses.

Let us consider the vertex α_1; its two out-neighbours in H are $(x_1, x_1\overline{x}_1, 1)$ and $(\overline{x}_1, x_1\overline{x}_1, 1)$. So exactly one of the arcs $(\alpha_1, (x_1, x_1\overline{x}_1, 1))$, $(\alpha_1, (\overline{x}_1, x_1\overline{x}_1, 1))$ is part of $\mathcal{HP}$. The same is true for α_i, $1 < i \leq n$. As a consequence, we can define a valid assignment of the variables x_i, $1 \leq i \leq n$, by setting $x_i = \text{T}$ if and only if the arc $(\alpha_i, (x_i, x_i\overline{x}_i, 1))$ belongs to $\mathcal{HP}$.

Next, we address the vertices β_j, $1 \leq j \leq m$. The construction in Steps 2 and 3 is such that each vertex β_j, $1 \leq j \leq m$, has two out-neighbours, $(a_j, a_jb_j, 1)$ and $(b_j, a_jb_j, 1)$.

This implies that the assignment defined above is such that there is at least one true literal in each clause. Indeed, if we assume that the clause $c_j = \{\ell_{j_1}, \ell_{j_2}, \ell_{j_3}\}$ does not contain any true literal, then the component $H_{a_j\ell_{j_1}}$ is completely visited by $\mathcal{HP}$ from the a_j-side, because $\ell_{j_1} = \text{F}$ implies that the arc $(\alpha_j, (\ell_{j_1}, \ell_{j_1}\overline{\ell}_{j_1}, 1))$ is not part of $\mathcal{HP}$ and does not give access to the ℓ_{j_1}-side. Similarly, the components $H_{b_j\ell_{j_2}}$ and $H_{d_j\ell_{j_3}}$ are completely visited by $\mathcal{HP}$ from the b_j- and d_j-sides, respectively. This in turn implies that in $\mathcal{HP}$ we have the arcs $((a_j, a_j\ell_{j_1}, 6), \beta'_j)$, $(\beta_j, (a_j, a_jb_j, 1))$, $((d_j, d_j\ell_{j_3}, 6), \beta_{j+1})$ and $(\beta'_j, (d_j, d_ja_j, 1))$ — replace β_{j+1} by δ if $j = m$. Now how does $\mathcal{HP}$ go through $(b_j, b_j\ell_{j_2}, 6)$? It cannot be with the help of the ℓ_{j_2}-side of $H_{b_j\ell_{j_2}}$, so there are only two possibilities left: but if it is with the arc $((b_j, b_j\ell_{j_2}, 6), \beta'_j)$, then β'_j has three neighbours in $\mathcal{HP}$, which is impossible; and if it is with the arc $((b_j, b_j\ell_{j_2}, 6), \beta_{j+1})$, then in $\mathcal{HP}$, the vertex β_{j+1} has two in-neighbours, which is impossible —including when $j = m$ and β_{j+1} is replaced by δ. From this we can conclude that the clause $c_j = \{\ell_{j_1}, \ell_{j_2}, \ell_{j_3}\}$ contains at least one true literal.

Assume next that one clause has at least two true literals: without loss of generality, $c_j = \{\ell_{j_1}, \ell_{j_2}, \ell_{j_3}\}$ is such that $\ell_{j_1} = \ell_{j_2} = \text{T}$. Then $\mathcal{HP}$ has no access to the $\overline{\ell}_{j_1}$- and $\overline{\ell}_{j_2}$- sides of the components involving $\overline{\ell}_{j_1}$ or $\overline{\ell}_{j_2}$, but, since there is the edge $\overline{\ell}_{j_1}\overline{\ell}_{j_2}$ in G_{VC}, this means that $\mathcal{HP}$ has no way of visiting the component $H_{\overline{\ell}_{j_1}\overline{\ell}_{j_2}}$. Therefore, we have just established that the assignment derived from the path $\mathcal{HP}$ 1-3-satisfies all the clauses. This, together with the fact that two assignments 1-3-satisfying the clauses lead to two paths, shows that a NO answer to the instance of U-1-3-SAT implies a NO answer for the constructed instance H of U-HAMP[O].

2) We want to show that a unique assignment $\mathcal{A}$ 1-3-satisfying all the clauses leads to a unique path in H. This assignment leads to a unique vertex cover V^*, of size $n + 2m$, in G_{VC}, and to a path in H, as already seen. Now assume that

we have a second path, so that these two paths, which we call $\mathcal{HP}_1$ and $\mathcal{HP}_2$, both lead, with the above description in 1), to the same $\mathcal{A}$ and the same V^*.

The two paths must behave in the same way over the components $H_{x_i\overline{x}_i}$, $1 \leq i \leq n$: otherwise, from them we could define two different valid assignments, which would both, as seen previously, 1-3-satisfy the clauses.

Next, consider the clause $c_j = \{\ell_{j_1}, \ell_{j_2}, \ell_{j_3}\}$ and assume without loss of generality that $\mathcal{A}(\ell_{j_1}) = \text{T}$, $\mathcal{A}(\ell_{j_2}) = \mathcal{A}(\ell_{j_3}) = \text{F}$; this implies, for both $\mathcal{HP}_1$ and $\mathcal{HP}_2$, that the components $H_{\ell_{j_1}\overline{\ell}_{j_1}}$, $H_{\ell_{j_2}\overline{\ell}_{j_2}}$ and $H_{\ell_{j_3}\overline{\ell}_{j_3}}$ are completely visited from the ℓ_{j_1}-, $\overline{\ell}_{j_2}$- and $\overline{\ell}_{j_3}$-sides, respectively, so that both paths have no access to the ℓ_{j_2}- nor ℓ_{j_3}-sides. As a consequence, between β_j and β_{j+1} (or β_m and δ), the components $H_{b_j\ell_{j_2}}$ and $H_{d_j\ell_{j_3}}$ are completely visited from the b_j- and d_j-sides, respectively. Then necessarily the following arcs belong to $\mathcal{HP}_1$ and $\mathcal{HP}_2$, going along the d_j-side:
$((d_j, d_j\ell_{j_3}, 6), \beta_{j+1})$ —or $((d_j, d_j\ell_{j_3}, 6), \delta)$—, $((d_j, d_jb_j, 6), (d_j, d_j\ell_{j_3}, 1))$, $((d_j, d_ja_j, 6), (d_j, d_jb_j, 1))$, $(\beta_j', (d_j, d_ja_j, 1))$;

and going along the b_j-side:
$((b_j, b_j\ell_{j_2}, 6), \beta_j')$ (because β_{j+1} —or δ— cannot have two in-neighbours), and $((b_j, b_jd_j, 6), (b_j, b_j\ell_{j_2}, 1))$, $((b_j, b_ja_j, 6), (b_j, b_jd_j, 1))$.

The component $H_{b_jd_j}$ must be visited in parallel, and it is $(\beta_j, (b_j, b_ja_j, 1))$ that belongs to the two paths.

We can see that all the components containing a_j, in particular $H_{a_j\ell_{j_1}}$, must be completely visited from the sides opposite a_j. So far, we have proved that the two paths $\mathcal{HP}_1$ and $\mathcal{HP}_2$ behave identically between β_j and β_{j+1} (or β_m and δ), including on the components corresponding to membership edges (between literals and triangles).

Consider now what happens between α_i and α_{i+1} (or α_n and β_1). Assume without loss of generality that, say, $\mathcal{A}(x_i) = \text{T}$, so that $(\alpha_i, (x_i, x_i\overline{x}_i, 1))$ is part of the two paths. Consider the components involving x_i in H: there are first those involving vertices of type a, b or d, which translate the membership of x_i to a certain number of clauses, and which we called $t_1, \ldots, t_q$ in Step 2(a); we have already seen in the previous paragraph that these components must be completely visited from the x_i-side.

Then we consider the components created from the edges in E_j', cf. Step 2(b); here, some edges in G_{VC} can have both ends in V^*, but, using similar arguments as before, we can see that the two paths will visit all these components in the same way: consider the clause $c_j = \{\ell_{j_1}, \ell_{j_2}, \ell_{j_3}\}$ and the corresponding set E_j', and assume without loss of generality that $x_i = \ell_{j_1}$, so that $\mathcal{A}(\ell_{j_1}) = \text{T}$, which implies that $\mathcal{A}(\ell_{j_2}) = \mathcal{A}(\ell_{j_3}) = \text{F}$; then $H_{\overline{\ell}_{j_1}\overline{\ell}_{j_2}}$ and $H_{\overline{\ell}_{j_1}\overline{\ell}_{j_3}}$ must be completely visited from the $\overline{\ell}_{j_2}$- and $\overline{\ell}_{j_3}$-sides, respectively, and $H_{\overline{\ell}_{j_2}\overline{\ell}_{j_3}}$ in parallel, i.e., the two paths have no choice but to behave identically on all three components. As for the components with $\overline{x}_i$, they must be completely visited from the side which is not the side of $\overline{x}_i$.

So we have just proved that the two paths are identical between α_1 and α_2, ..., α_n and β_1.

Therefore, the two paths (between α_1 and δ) are one and the same. ◊

References

[1] J.-P. BARTHÉLEMY, G. D. COHEN and A. C. LOBSTEIN: *Algorithmic Complexity and Communication Problems*, London: University College of London, 1996.

[2] A. BLASS and Y. GUREVICH: On the unique satisfiability problem, *Information and Control*, Vol. 55, pp. 80-88, 1982.

[3] C. CALABRO, R. IMPAGLIAZZO, V. KABANETS and R. PATURI: The complexity of Unique k-SAT: an isolation lemma for k-CNFs, *Journal of Computer and System Sciences*, Vol. 74, pp. 386–393, 2008.

[4] S. A. COOK: The complexity of theorem-proving procedures, *Proceedings of 3rd Annual ACM Symposium on Theory of Computing*, pp. 151–158, 1971.

[5] T. CORMEN: Algorithmic complexity, in: K. H. Rosen (ed.) *Handbook of Discrete and Combinatorial Mathematics*, pp. 1077–1085, Boca Raton: CRC Press, 2000.

[6] M. R. GAREY and D. S. JOHNSON: *Computers and Intractability, a Guide to the Theory of NP-Completeness*, New York: Freeman, 1979.

[7] L. HEMASPAANDRA: Complexity classes, in: K. H. Rosen (ed.) *Handbook of Discrete and Combinatorial Mathematics*, pp. 1085–1090, Boca Raton: CRC Press, 2000.

[8] O. HUDRY and A. LOBSTEIN: Complexity of unique (optimal) solutions in graphs: Vertex Cover and Domination, *Journal of Combinatorial Mathematics and Combinatorial Computing*, to appear.

[9] O. HUDRY and A. LOBSTEIN: Some complexity considerations on the uniqueness of solutions for satisfiability and colouring problems, submitted.

[10] O. HUDRY and A. LOBSTEIN: Unique (optimal) solutions: complexity results for identifying and locating-dominating codes, submitted.

[11] D. S. JOHNSON: A catalog of complexity classes, in: *Handbook of Theoretical Computer Science, Vol. A: Algorithms and Complexity*, van Leeuwen, Ed., Chapter 2, Elsevier, 1990.

[12] R. M. KARP: Reducibility among combinatorial problems, in: R. E. Miller and J. W. Thatcher (eds.) *Complexity of Computer Computations*, pp. 85–103, New York: Plenum Press, 1972.

[13] C. H. PAPADIMITRIOU: On the complexity of unique solutions, *Journal of the Association for Computing Machinery*, Vol. 31, pp. 392-400, 1984.

[14] C. H. PAPADIMITRIOU: *Computational Complexity*, Reading: Addison-Wesley, 1994.

[15] C. H. PAPADIMITRIOU and M. YANNAKAKIS: The complexity of facets (and some facets of complexity), *Journal of Computer and System Sciences*, Vol. 28, pp. 244–259, 1984.

[16] T. J. SCHAEFER: The complexity of satisfiability problems, *Proceedings of 10th Annual ACM Symposium on Theory of Computing*, pp. 216–226, 1978.

[17] https://complexityzoo.uwaterloo.ca/Complexity_Zoo