Improved Inapproximability Results for the Shortest Superstring and the Bounded Metric TSP

Marek Karpinski* Richard Schmied†

Abstract

We present a new method for proving explicit approximation lower bounds for the Shortest Superstring problem, the Maximum Compression problem, Maximum Asymmetric TSP problem, the $(1,2)$–ATSP problem, the $(1,2)$–TSP problem, the $(1,4)$–ATSP problem and the $(1,4)$–TSP problem improving on the best up to now known approximation lower bounds for those problems.

1 Introduction

We study explicit inapproximability bounds of several combinatorial optimization problems connected to the Shortest Superstring and TSP problems. We start with the definitions of the underlying problems.

The **Shortest Superstring Problem (SSP)** is the following problem: Given a finite set S of strings and the objective is to construct their shortest superstring, which is the shortest possible string such that every string in S is a proper substring of it.
The task of computing a shortest common superstring appears in a wide variety of application related to computational biology [L90]. Vassilevska [V05] proved that approximating the SSP with less than $1217/1216$ is **NP**-hard. The currently best known approximation algorithm is due to Mucha [M12] and yields an approximation factor of $2\frac{11}{23}$.

In this paper, we prove that the SSP is **NP**-hard to approximate within any constant approximation ratio less than $333/332$.

The **Asymmetric Traveling Salesman Problem (ATSP)** is defined as follows: We

*Dept. of Computer Science and the Hausdorff Center for Mathematics, University of Bonn. Supported in part by DFG grants and the Hausdorff Grant EXC59-1/2. Email: marek@cs.uni-bonn.de

†Dept. of Computer Science, University of Bonn. Work supported by Hausdorff Doctoral Fellowship. Email: schmied@cs.uni-bonn.de

are given an asymmetric metric space (V, d), that is, d is not necessarily symmetric, and we would like to construct a shortest tour visiting every vertex exactly once.

The best known algorithm for the ATSP approximates the solution within $O(\log n/\log\log n)$ [AGM+10], where n is the number of vertices in the given metric space. On the other hand, in [KLS13], it was proved that the ATSP is **NP**-hard to approximate to within any approximation factor less than $75/74$. It is conceivable that the special cases with bounded metric are easier to approximate than the cases when the distance between two points grows with the size of the instance. Clearly, the $(1, B)$-ATSP, in which the distance function is taking values in the set $\{1, \ldots, B\}$, can be approximated within B by just picking any tour as the solution. When we restrict the problem to distances one and two, it can be approximated within $5/4$ due to Bläser [B04]. Furthermore, it is **NP**-hard to approximate this problem with an approximation factor less than $321/320$ [EK06]. For the case $B = 8$, in [EK06], a reduction was constructed yielding the approximation lower bound of $135/134$ for the $(1, 8)$–ATSP.

In this paper, we prove **NP**-hardness of approximating the $(1, 2)$-ATSP and $(1, 4)$-ATSP to within any approximation factor less than $207/206$ and $141/140$, respectively.

The **Metric Traveling Salesman Problem (TSP)** is a special case of the ATSP, in which we are given a metric space (V, d) and the task consists of constructing a shortest tour visiting each vertex exactly once.

The TSP in metric spaces is one of the most fundamental **NP**-hard optimization problems. Christofides [C76] gave an algorithm approximating the TSP within $3/2$, that is, an algorithm that produces a tour with length being at most a factor $3/2$ from the optimum. As for lower bounds, a reduction due to Papadimitriou and Yannakakis [PY93] and the PCP Theorem [ALM+98] together imply that there exists some constant, not greater than $1 + 10^{-6}$, such that it is **NP**-hard to approximate the TSP with distances either one or two. For discussion of bounded metrics TSP, see also [T00]. Since then there was a series of results on the hardness of approximating the TSP improving the inapproximability threshold to $123/122$. (cf. [BS00], [E03], [PV06], [L12],[KLS13]).

The restricted version of the TSP, in which the distance function takes values in $\{1, \ldots, B\}$, is referred to as the $(1, B)$-*TSP*. The $(1, 2)$-TSP can be approximated in polynomial time with an approximation factor $8/7$ due to Berman and Karpinski [BK06]. On the other hand, in [EK06], it was proved that the inapproximability threshold for the $(1, B)$-TSP is $741/740$ and $389/388$ for $B = 2$ and $B = 8$, respectively.

In this paper, we prove that it is **NP**-hard to approximate the $(1, 2)$-TSP and the $(1, 4)$-TSP to within any approximation factor less than $535/534$ and $337/336$, respectively.

The **Maximum Compression Problem (MAX-CP)** is defined as follows: We are given a collection of strings $S = \{s_1, \ldots, s_n\}$. The task is to find a superstring for S with maximum compression, which is the difference between the sum of the lengths of the given strings and the length of the superstring.

In the exact setting, an optimal solution to the SSP is an optimal solution to the MAX-

CP, but the approximate solutions can differ significantly in the sense of approximation factor. The Maximum Compression problem arises in various data compression problems (cf. [S88]). The best known efficient approximation algorithm achieves an approximation factor 1.5 [KLS$^+$05] by reducing it to the MAX-ATSP defined below.

On the approximation lower bound side, Vassilevska [V05] proved an inapproximability threshold for the MAX-CP of $1072/1071$.

In this paper, we prove that approximating the MAX-CP to within any approximation factor less than $204/203$ is **NP**-hard.

The **Maximum Asymmetric Traveling Salesman Problem (MAX-ATSP)** is defined as follows: We are given a complete directed graph G and a weight function w assigning each edge of G a non-negative weight. The task is to find a tour of maximum weight visiting every vertex of G exactly once .

This problem is well-known and motivated by several applications and in particular, a good approximation algorithm for the MAX-ATSP implies a good approximation algorithm for many other optimization problems such as the SSP, the MAX-CP and the $(1,2)$-ATSP (cf. [BGS02]).

The MAX-$(0,1)$-ATSP is the restricted version of the MAX-ATSP, in which the weight function w takes values in the set $\{0,1\}$. Vishwanathan [V92] constructed an approximation preserving reduction proving that every $(1/\alpha)$-approximation algorithm for the MAX-$(0,1)$-ATSP yields a $(2-\alpha)$- approximation algorithm for the (1,2)-ATSP. Therefore, all negative results concerning the approximation of the $(1,2)$-ATSP transform into hardness of approximation results for the MAX-$(0,1)$-ATSP. Due to the explicit approximation lower bound for the (1,2)-ATSP given in [EK06], it is **NP**-hard to approximate the MAX-(0,1)-ATSP to within every approximation factor less than $320/319$. The best known approximation algorithm for the restricted version of this problem is due to Bläser [B04] and achieves an approximation factor $5/4$.

For the unrestricted version of the problem, Kaplan et al. [KLS$^+$05] designed an algorithm with the best up to now known approximation upper bound of $3/2$. Elbassioni, Paluch and Zuylen [EPZ12] gave a simpler approximation algorithm for the problem with the same approximation ratio. In this paper, we prove that approximating the MAX-ATSP within every approximation factor less than $204/203$ is **NP**-hard.
The preliminary versions of this paper appeared in [KS11] and [KS12].

2 The Proof Methods and the Summary of Results

The results of the paper depend on several new reductions from a bounded occurrence CSP problem (cf. [K01]) called the Hybrid problem, which was proven to be approximation hard to within some constant (cf. [BK99]). The reduction method defines for each problem so called parity gadgets that are simulating the variables of the Hybrid problem and also *transmit* the parity information to the gadgets that are simulating the linear equation mod 2 with exactly three variables in the instance of the Hybrid problem. The

crucial point of the reduction is that we make essential use of the underlying structure of the equations in the Hybrid problem, which are induced by a 3-regular amplifier graph. This could be a more widely useful method for improving the approximation lower bounds of other problems (see [KLS13] and [KS13] for two extensions of this idea).

The explicit approximation lower bounds are summarized in Table 1.

Problem	Our Results	Previously known
$(1,2)$–ATSP	$207/206$ (Theorem 1 (i))	$321/320$ [EK06]
$(1,2)$–TSP	$535/534$ (Theorem 1 (ii))	$741/740$ [EK06]
$(1,4)$–ATSP	$141/140$ (Theorem 1 (iii))	$321/320$ [EK06]
$(1,4)$–TSP	$337/336$ (Theorem 1 (iv))	$741/740$ [EK06]
MAX–ATSP	$204/203$ (Corollary 1)	$320/319$ [EK06]
MAX–CP	$204/203$ (Theorem 1 (v))	$1072/1071$ [V05]
SSP	$333/332$ (Theorem 1 (vi))	$1217/1216$ [V05]

Table 1: Comparison of our results and previously known inapproximability factors.

3 Outline of the Paper

The organization of the paper is as follows. In Section 5, we formulate our main results. In Section 6, we give the definition of the Hybrid problem and state some results concerning its hardness of approximation. In Section 7, we construct the reduction from the Hybrid problem to the $(1,2)$-ATSP. In Section 8, we give the proof of the approximation lower bound for the $(1,4)$-ATSP. In Section 9, we present the inapproximability result for the $(1,2)$-TSP. In Section 10, we consider the $(1,4)$-TSP. In Section 11, we give the proof of the claimed inapproximability thresholds for the SPP and the MAX-CP.

4 Preliminaries

For $i \in \mathbb{N}$, we use the abbreviation $[i]$ for the set $\{1,\ldots,i\}$. Given a finite alphabet Σ, a string is an element of Σ^*. Given a string v, we denote the length of v by $|v|$. For two strings x and y, we define the overlap of x and y, denoted $ov(x,y)$, as the longest suffix of x that is also a prefix of y. Furthermore, we define the prefix of x with respect to y, denoted $pref(x,y)$, as the string u with $x = u\,ov(x,y)$. Let x and y be two strings with $i = |ov(x,y)| > 0$. We denote the string that is obtained by overlapping x with y by i letters as $x \overset{i}{\rightarrow} y$.

Throughout in this paper, an instance (V,d) of the $(1,2)$-ATSP problem is specified by means of a directed graph D_V, where $(x,y) \in A(D_V)$ if and only if $d(x,y) = 1$. A *tour*

$\sigma \subset V \times V$ in an asymmetric metric space (V, d) is a directed Hamiltonian cycle in the associated directed graph $(V, V \times V)$ with total cost $\sum_{a \in \sigma} d(a)$. For $n \in \mathbb{N}$, we refer to an arc $a \in V \times V$ as a *n-arc* if $d(a) = n$ holds.
Let S be a collection of strings over Σ such that no string is a proper substring of another string in S. Then, we define an instance of the ASTP problem by (V_S, d_S), where $V_S = S \cup \{\Gamma\}$ with $\Gamma \notin \Sigma$ and $d_S(s_i, s_j) = |pref(s_i, s_j)|$ for all $s_i, s_j \in V_S$. Note that we can construct from a shortest tour in (V_S, d_S) of length $(\ell + 1)$ a shortest superstring for S of length ℓ and vice versa (e.g. [M12]).
Given S and a superstring s for S, we introduce the notion of compression of s with respect to S denoted $comp(s, S)$ and defined by $comp(s, S) = \sum_{s_i \in S} |s_i| - |s|$.

5 Main Results

In the subsequent sections, we are going to prove the following Theorem.

Theorem 1. *It is* **NP***-hard to approximate*

- (i) *the* $(1, 2)$*-ATSP to within every constant factor less than* $207/206$*,*
- (ii) *the* $(1, 2)$*-TSP to within every constant factor less than* $535/534$*,*
- (iii) *the* $(1, 4)$*-ATSP to within every constant factor less than* $141/140$*,*
- (iv) *the* $(1, 4)$*-TSP to within every constant factor less than* $337/336$*,*
- (v) *the MAX-CP to within every constant factor less than* $204/203$ *and*
- (vi) *the SSP to within every constant factor less than* $333/332$*.*

Since instances of the MAX-CP can be seen as restricted instances of the MAX-ATSP (cf. [BGS02]), we obtain immediately the following corollary.

Corollary 1. *It is* **NP***-hard to approximate the MAX-ATSP within every constant factor less than* $204/203$*.*

6 Hybrid Problem

In this section, we introduce the so called Hybrid problem and present the reduction which was constructed in [BK99] in order to prove first explicit approximation lower bounds for a number of bounded occurrence CSP optimization problems.

In order to prove hardness of approximation for the the Hybrid problem, we first consider the MAX-E3LIN2 problem which is defined as follows.

Definition 1 (MAX-E3LIN2 problem)**.** *We have given a system I of linear equations mod 2 and we want to find an assignment to the variables of I that satisfies the maximum number of equations.*

For the MAX-E3LIN2 problem, Håstad [H01] gave an optimal inapproximability result stated below.

Theorem 2 ([H01])**.** *For every $\epsilon \in (0, 1/2)$, there exists a constant B_ϵ and instances I of the MAX-E3LIN2 problem with $2 \cdot \gamma$ equations such that:*
(i) Each variable in the instance I appears in at most B_ϵ number of equations.
(ii) It is **NP***-hard to decide whether there is an assignment leaving at most $\epsilon \cdot \gamma$ equations unsatisfied, or every assignment leaves at least $(1 - \epsilon)\gamma$ equations unsatisfied.*

The Hybrid problem can be seen as a generalization of the MAX-E3LIN2 problem. Let us first give the definition of the Hybrid problem following [BK99].

Definition 2 (Hybrid problem)**.** *We have given a system $I_{\mathcal{H}}$ of linear equations mod 2 involving n variables, equations with exactly two variables, equations with exactly three variables and we are supposed to find an assignment to the variables of $I_{\mathcal{H}}$ that satisfies the maximum number of equations.*

In the following, we are going to present the construction of restricted instances of the Hybrid problem, in which every variable in the instance occurs in exactly 3 equations. In particular, we are going to give the proof of the theorem due to Berman and Karpinski [BK99] stated below.

Theorem 3 ([BK99])**.** *For every constant $\epsilon \in (0, 1/2)$, there exist instances of the Hybrid problem $I_{\mathcal{H}}$ involving 60ν equations with exactly two variables of the form $x \oplus y = 0$ and 2ν equations of the form $x \oplus y \oplus z = b$ with $b \in \{0, 1\}$ such that: (i) Each variable occurs exactly three times (ii) It is* **NP***-hard to decide whether there is an assignment to the variables that leaves at most $\epsilon \cdot \nu$ equations unsatisfied, or else every assignment leaves at least $(1 - \epsilon)\nu$ equations unsatisfied. (iv) An assignment to the variables in $I_{\mathcal{H}}$ can be transformed efficiently into an assignment satisfying all 60ν equations with two variables without decreasing the total number of satisfied equations in $I_{\mathcal{H}}$.*

We are going to describe briefly the reduction from the MAX-E3LIN2 problem to the Hybrid problem and give the proof of Theorem 3. For this, let us first give some definitions and in particular, review some properties of amplifier graphs (see also [BK01] and [BK03]). One application of amplifier graphs is proving hardness of approximation for Constraint Satisfaction problems, in which every variable occurs a bounded number of times.

Definition 3 (Regular amplifier)**.** *Let G be a graph and $X \subset V(G)$. The graph G is a d-regular amplifier for the set X if the following two conditions hold:*

(i) *The vertices in X have degree $(r - 1)$, whereas all vertices in $V(G) \backslash X$ have degree r.*

(ii) *For every subset $S \subset V(G)$ with $S \neq \emptyset$, we have the condition that*

$$|E(S, V(G)\backslash S)| \geq \min\{|S \cap X|, |(V(G)\backslash S) \cap X|\},$$

where $E(S, V(G)\backslash S)$ is defined as $\{e \in E(G) \mid |S \cap e| = 1\}$.

We refer to the set $X \subset V(G)$ as the set of contact vertices, whereas $V(G)\backslash X$ is the set of checker vertices.

In [BK01], a very special 3-regular amplifier graph was constructed and called *wheel amplifier*. Let us give the description of a wheel amplifier.
A *wheel amplifier* $\mathcal{W}$ with $2n$ contact vertices is constructed by creating a Hamiltonian cycle on $14n$ vertices $1, 2, ..., 14n$ with edge set $C(\mathcal{W})$. Afterwards, a perfect matching $M(\mathcal{W})$ is chosen uniformly at random from the vertices whose number is not a multiple of 7. The vertices, that are matched, are our checker vertices and the remaining vertices are the contact vertices. The union of $M(\mathcal{W})$ and $C(\mathcal{W})$ is the edge set of $\mathcal{W}$.

Berman and Karpinski [BK01] proved the following theorem on the existence of 3-regular amplifier graphs.

Theorem 4 ([BK01])**.** *With high probability, wheel amplifier are 3-regular amplifier.*

We are ready and give the proof of Theorem 3.

Proof of Theorem 3. For a fixed $\epsilon \in (0, 1/2)$, let I an instance of the MAX-E3LIN2 problem, in which the number of occurences of each variable is bounded by a constant B_ϵ.

For each variable x_i in I, we denote by n_i the number of equations in which x_i appear in I. For each variable x_i, we introduce a set of $\mu := 7 \cdot n_i$ new variables $V_i = \{x_j^i\}_{j=1}^{\mu}$. Furthermore, we create a wheel amplifier $\mathcal{W}_i$ on μ vertices with n_i contact vertices. Since d_i is bounded by B_ϵ, $\mathcal{W}_i$ can be constructed in constant time. In the remainder, we call $x_j \in V_i$ a *contact* variable if the corresponding index is a contact vertex of $\mathcal{W}_i$. The remaining variable in V_i are called *checker* variables.

Let us now define the equations of the instance $I_{\mathcal{H}}$ of the Hybrid problem: For each edge $\{j, k\} \in M(\mathcal{W}_i)$, we introduce $x_j^i \oplus x_k^i = 0$ and refer to those equations as *matching equations*. For each edge $\{l, t\}$ in $C(\mathcal{W}_i)$, we create $x_l^i \oplus x_t^i = 0$. We refer to the equation $x_1 \oplus x_2 = 0$ as the *cycle border equation*. All other remaining are called *cycle equations*. Finally, we substitute the ℓ-th appearance of x_i in I by the contact variable x_ξ^i with $\xi = 7 \cdot \ell$. Summarizing, we have 2ν equations with three variables in $I_{\mathcal{H}}$, 60ν equations with two variables and each variable occurs in exactly 3 equations.

We call an assignment ϕ to the variables of $I_{\mathcal{H}}$ *consistent* if for each variable x_i in I, there is a $b_i \in \{0, 1\}$ such that $x_j^i = b_i$ for all $j \in [\mu]$. By the properties of a amplifier constructed in Theorem 4, we can transform efficiently an assignment to the variables of $I_{\mathcal{H}}$ into a consistent one without decreasing the number of satisfied equations and the proof of Theorem 3 follows. □

7 The $(1,2)$–ATSP

Given an instance of the Hybrid problem $I_{\mathcal{H}}$, we want to transform $I_{\mathcal{H}}$ into an instance of the $(1,2)$-ATSP. Fortunately, the special structure of the linear equations in the Hybrid problem, produced by the reduction due to Berman and Karpinski [BK99], is particularly well-suited for our reduction, since a part of the equations with two variables form a Hamiltonian cycle and every variable occurs exactly three times. The main idea of our reduction is to make use of the special structure of the wheels in $I_{\mathcal{H}}$. Every wheel $\mathcal{W}_l$ in $I_{\mathcal{H}}$ corresponds to a subgraph D_l in the instance $D^{12}_{\mathcal{H}}$ of the $(1,2)$-ATSP. Moreover, D_l forms almost a cycle, which can be traversed in two directions. An assignment to the variable x^l will have a natural interpretation in this reduction. The parity of x^l corresponds to the direction of movement in D_l of the underlying tour.

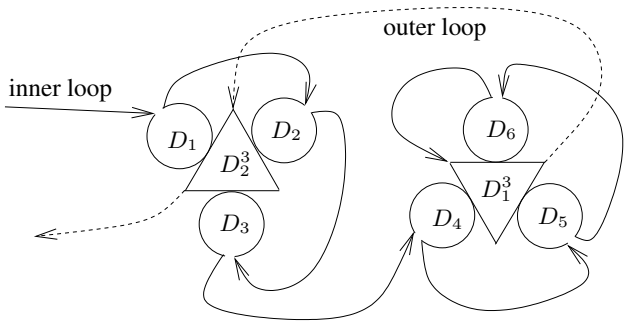


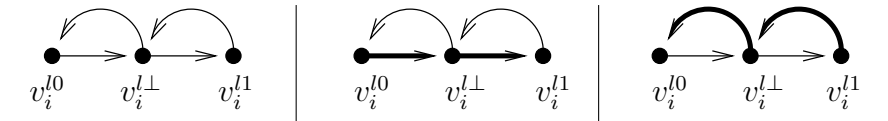
Figure 1: An illustration of a tour in $D^{12}_{\mathcal{H}}$.

The wheel graphs $D_1, \ldots, D_n$ of $D^{12}_{\mathcal{H}}$ are connected and build together the *inner loop* of $D^{12}_{\mathcal{H}}$ (Figure 1). Every variable x^l_i in a wheel $\mathcal{W}_l$ possesses an associated parity gadget P^l_i (Figure 2 (a)) in D_l as a subgraph. The two natural ways to traverse a parity gadget will be called $0/1$-traversals and correspond to the parity of the variable x^l_i. Some of the parity gadgets in D_l are also contained in graphs $D^{3\oplus}_c$ (Figure 4 and Figure 5 for a more detailed view) corresponding to equations with three variables of the form $x \oplus y \oplus z = 0 = b^3_c$. Note that we may assume that equations with three variables are all of the form $x \oplus y \oplus z = 0$, where x, y and z are variables or negated variables. Those graphs are connected and build the *outer loop* of $D^{12}_{\mathcal{H}}$. In the outer loop of the tour, we check whether the $0/1$-traversals of the parity gadgets correspond to an assignment that satisfies the underlying equation with three variables. If an underlying equation is not satisfied by the assignment defined via $0/1$-traversals of the associated parity graphs, we will have to use an arc with weight 2.

7.1 Constructing a Tour from an Assignment

Given a instance of the Hybrid problem $I_{\mathcal{H}}$, we are going to construct the corresponding instance $D^{12}_{\mathcal{H}}$ of the $(1,2)$-ATSP. For every type of equation in $I_{\mathcal{H}}$, we will introduce a specific graph or a specific way to connect the so far constructed subgraphs. In particular,

we will distinguish between graphs corresponding to cycle equations, matching equations, cycle border equations and equations with three variables. First of all, we introduce graphs corresponding to variables in $I_{\mathcal{H}}$.



(a) Parity gadget P_i^l | (b) 1-traversal of P_i^l | (c) 0-traversal of P_i^l

Figure 2: Traversals of the parity gadget P_i^l. Traversed arcs are illustrated by thick arrows.

Variable Graphs: For every variable x_i^l in $I_{\mathcal{H}}$, we introduce the parity gadget P_i^l consisting of the vertices $\{v_i^{l1}, v_i^{l\perp}, v_i^{l0}\}$, which is displayed in Figure 2(a).

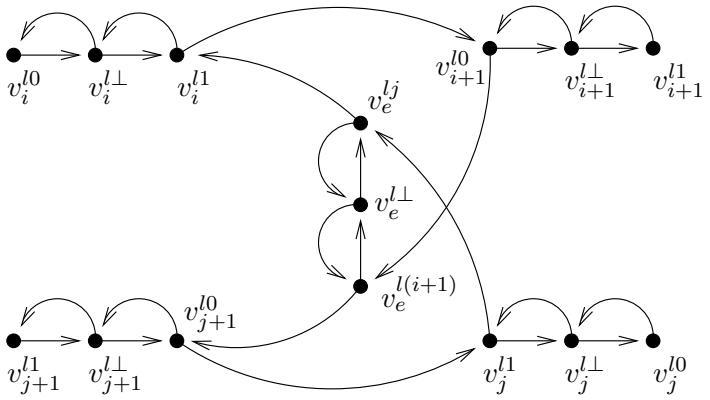


Figure 3: Connecting the parity gadget P_e^l

Matching and Cycle Equations: Let $I_{\mathcal{H}}$ be an instance of the hybrid problem, $\mathcal{W}_l$ a wheel of $I_{\mathcal{H}}$ and M_l the associated perfect matching. Furthermore, let $x_i^l \oplus x_j^l = 0$ with $e = \{i, j\} \in M(\mathcal{W}_l)$ and $i < j$ be an matching equation. For $x_i^l \oplus x_{i+1}^l = 0$ and $x_j^l \oplus x_{j+1}^l = 0$, we introduce the associated parity gadget P_e^l consisting of the vertices v_e^{lj}, $v_e^{l\perp}$ and $v_e^{l(i+1)}$. In addition to it, we connect the parity gadgets P_i^l, P_{i+1}^l, P_j^l, P_{j+1}^l and P_e^l as displayed in Figure 3.

Equations with three variables: Let $x_i^l \oplus x_j^s \oplus x_t^k = 0 = b_c^3$ be an equation with three variables in $I_{\mathcal{H}}$. Then, we introduce the graph $D_c^{3\oplus}$ (Figure 4), which consists of the vertices s_c, v_c^1, v_c^2, v_c^3 and s_{c+1}. Papadimitriou and Vempala [PV06] used this graph in their reduction and proved the following statement.

Lemma 1 ([PV06])**.** *There is a Hamiltonian path from s_c to s_{c+1} in Figure 4 if and only if an even number of dashed arcs is traversed.*

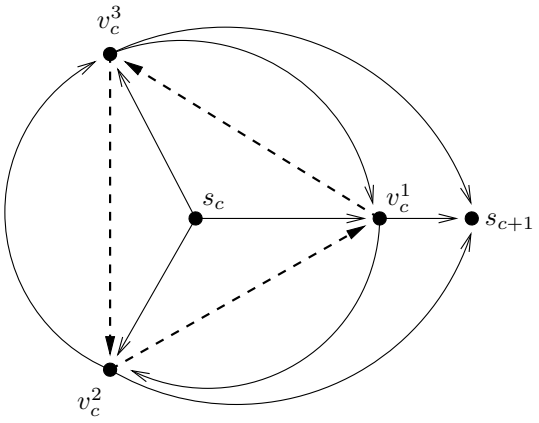


Figure 4: Gadget simulating $x \oplus y \oplus z = 0$.

This construction is extended by replacing the dashed arcs with the parity gadgets P_e^l, P_b^s and P_a^k, where $e = \{i, i+1\}$, $b = \{j, j+1\}$ and $a = \{t, t+1\}$. In Figure 5, we display $D_c^{3\oplus}$ with its connections to the graph corresponding to the cycle equation $x_i^l \oplus x_{i+1}^l = 0$.

If x_i appears negated in the equation with three variables, we connect the parity gadgets via (v_i^{l1}, v_e^{l0}), (v_{i+1}^{l0}, v_i^{l1}) and (v_e^{l1}, v_{i+1}^{l0}).

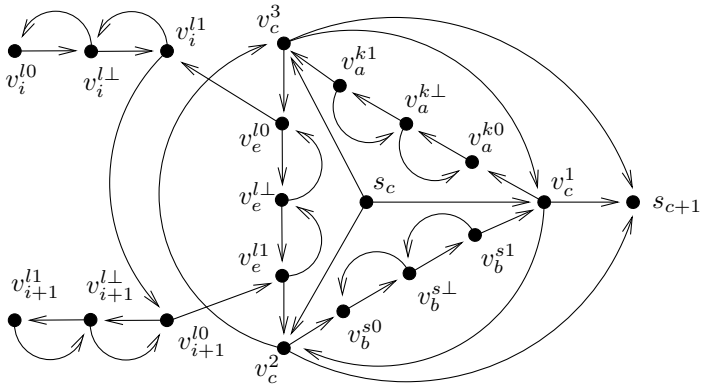


Figure 5: Graph $D_c^{3\oplus}$ corresponding to $x_i^l \oplus x_j^s \oplus x_t^u = 0 = b_c^3$ connected to gadgets simulating to $x_i^l \oplus x_{i+1}^l = 0$.

Cycle Border Equations: Let $\mathcal{W}_l$ and $\mathcal{W}_{l+1}$ be wheels of $I_{\mathcal{H}}$. In addition, let $x_1^l \oplus x_2^l = 0$ be the wheel border equation of $\mathcal{W}_l$. Then, we introduce the vertex b_l and connect it to v_2^{l0} and v_1^{l1}. Let b_{l+1} be the vertex corresponding to the wheel $\mathcal{W}_{l+1}$. We draw an arc from v_2^{l0} and v_1^{l1} to b_{l+1}.

7.2 Constructing a Tour from an Assignment

In this section, we are going to prove one direction of the reduction. In particular, we are going to prove the following lemma.

Lemma 2. *Let $\delta \in (0,1)$ be a constant, $I_{\mathcal{H}}$ an instance with n wheels, $60 \cdot \nu$ equations with two variables and $2 \cdot \nu$ equations with exactly three variables and ϕ an assignment, which leaves $\delta \cdot \nu$ equations in $I_{\mathcal{H}}$ unsatisfied. Then there exists a tour in the corresponding instance $D^{12}_{\mathcal{H}}$ with total cost at most $206\nu + n + 2 + \delta \cdot \nu$.*

Proof. Let $I_{\mathcal{H}}$ be an instance of the Hybrid problem consisting of the wheels $\mathcal{W}_1$, $\mathcal{W}_2$, ..., $\mathcal{W}_n$, 60ν equations with 2 variables and 2ν equations with three variables. Suppose we have given an assignment ϕ to the variables of $I_{\mathcal{H}}$ leaving $\delta \cdot \nu$ equations unsatisfied for a constant $\delta \in (0,1)$. We are going to construct the associated Hamiltonian tour σ_ϕ in $D^{12}_{\mathcal{H}}$. According to Theorem 3, we may assume that all equations with 2 variables in $I_{\mathcal{H}}$ are satisfied by ϕ. Thus, all variables associated to a wheel take the same value under ϕ. The Hamiltonian tour σ_ϕ in $D^{12}_{\mathcal{H}}$ starts at the vertex b_1. From a high-level view, σ_ϕ traverses all graphs corresponding to equations associated with the wheel $\mathcal{W}_1$ using the $\phi(x^1_1)$-traversal of all parity graphs that correspond to variables of $\mathcal{W}_1$ ending with the vertex b_2. Successively, it passes all graphs for each wheel in $I_{\mathcal{H}}$ until it reaches the vertex $b_m = s_1$ as s_1 is the starting vertex of the graph $D^{3\oplus}_1$.

At this point, the tour begins to traverse the remaining graphs $D^{3\oplus}_c$, which are simulating the equations with three variables in $I_{\mathcal{H}}$. By now, some of the parity graphs appearing in graphs $D^{3\oplus}_c$ already have been traversed in the *inner loop* of σ_ϕ. The *outer loop* checks whether for each graph $D^{3\oplus}_c$, an odd number of parity graphs has been traversed in the inner loop. In every situation, in which ϕ leaves the underlying equation unsatisfied, the tour needs to use a 2-arc. Hence, the tour in $D^{12}_{\mathcal{H}}$ has total cost at most $3 \cdot 60\nu + (3 \cdot 3 + 4) \cdot 2\nu + (n+2) + \delta \cdot \nu$. □

7.3 Constructing an Assignment from a Tour

Let $I_{\mathcal{H}}$ be an instance of the Hybrid problem, $D^{12}_{\mathcal{H}}$ the associated instance of the $(1,2)$-ATSP and σ a tour in $D^{12}_{\mathcal{H}}$. We are going to define the assignment ψ_σ to variables in $I_{\mathcal{H}}$. In addition to it, we establish a connection between the total cost of σ and the number of satisfied equations by ψ_σ. Let us first introduce the notion of consistent tours.

Definition 4 (Consistent Tour)**.** *Let $I_{\mathcal{H}}$ be an instance of the Hybrid problem and $D^{12}_{\mathcal{H}}$ the associated instance of the $(1,2)$-ATSP. A tour in $D^{12}_{\mathcal{H}}$ is called consistent if every parity graph in $D^{12}_{\mathcal{H}}$ is traversed by means of a $0/1$-traversals.*

Due to the following lemma, we may assume that the underlying tour is consistent.

Lemma 3. *Let $I_{\mathcal{H}}$ be an instance of the Hybrid problem and $D^{12}_{\mathcal{H}}$ the associated instance of the $(1,2)$-ATSP. Any tour σ in $D^{12}_{\mathcal{H}}$ can be transformed in polynomial time into a consistent tour with at most the same cost as σ.*

Proof. For every parity graph contained in $D^{12}_{\mathcal{H}}$, it can be seen by considering all possibilities exhaustively that every tour in $D^{12}_{\mathcal{H}}$ that is not using the corresponding $0/1$-traversals can be modified into a tour with at most the same number of 2-arcs. Some cases are shown in Figure 6 and Figure 7. □

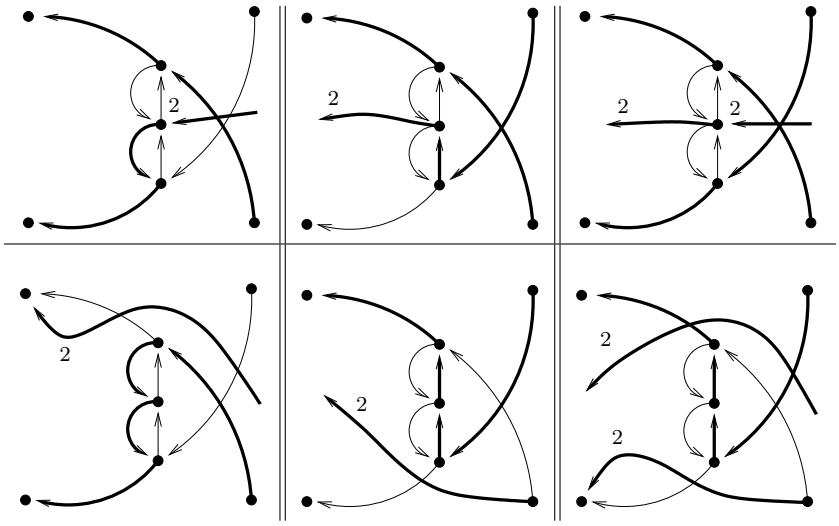


Figure 6: It is possible to transform the traversals in the upper row into the traversals in the lower row without increasing the total cost of the tour.

Let us define the assignment ψ_σ given a consistent tour σ in $D^{12}_{\mathcal{H}}$.

Definition 5 (Assignment ψ_σ)**.** *Let $I_{\mathcal{H}}$ be an instance of the Hybrid problem, $D^{12}_{\mathcal{H}}$ the associated instance of the $(1,2)$-ATSP. Given a consistent tour σ in $D^{12}_{\mathcal{H}}$, the corresponding assignment $\psi_\sigma : Var(I_{\mathcal{H}}) \to \{0,1\}$ is defined as $\psi_\sigma(x^l_i) = 1$ if σ uses a 1-traversal of P^l_i and 0 otherwise.*

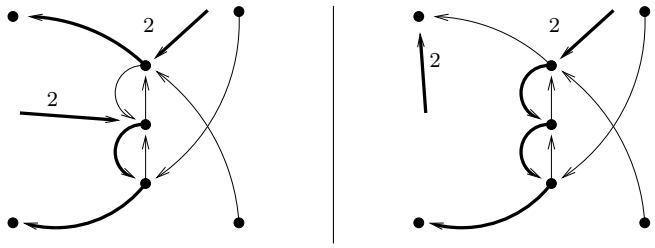


Figure 7: It is possible to transform the traversal in the left column into the traversal in the right column without increasing the total cost of the tour.

We are going to prove the other direction of the reduction and give the proof of the following lemma.

Lemma 4. *Let $\delta \in (0,1)$ be a constant, $I_{\mathcal{H}}$ an instance with n wheels, $60 \cdot \nu$ equations with two variables and $2 \cdot \nu$ equations with exactly three variables. Let σ be a tour in the corresponding instance $D^{12}_{\mathcal{H}}$ of the $(1,2)$-ATSP with length $206\nu + (n+2) + \delta \cdot \nu$. Then, we can convert σ into a tour π without increasing its length such that the corresponding assignment ψ_π leaves at most $\delta \cdot \nu$ equations in $I_{\mathcal{H}}$ unsatisfied.*

Proof. We are going to analyze different parts of the underlying tour. In particular, we consider different subgraphs of $D^{12}_{\mathcal{H}}$ simulating equations and define a lower bound on the sum of the length of arcs in the subgraph that are needed to define a tour in $D^{12}_{\mathcal{H}}$ in dependence whether the associated assignment ψ_π is satisfying the underlying equation. Let us start with graphs corresponding to matching equations.

Matching equations: Suppose we have given the equations $x_i \oplus x_{i+1} = 0$, $x_i \oplus x_j = 0$, $x_j \oplus x_{j+1} = 0$ and a tour σ passing the graphs displayed in Figure 3, we are going to analyze the relation between the cost of the tour and the number of satisfied equations by ψ_σ. In some cases, we have to transform σ without increasing the cost in order to obtain an assignment that satisfies more equations. The following definition will be useful in this context. We introduce the notion of local cost of a tour σ in $V(e,l) = \{v_e^{lj}, v_e^{l\perp}, v_e^{l(i+1)}, v_{i+1}^{l0}, v_i^{l1}, v_{j+1}^{l0}, v_j^{l1}\}$, where $e = \{i,j\}$. Futhermore, for an arc (x,y), we define $V\big((x,y)\big) = \{x,y\}$.

$$c_\sigma(V(e,l)) = \sum_{v \in V(e,l)} \quad \sum_{a \in \sigma\colon\, v \in V(a)} \frac{d(a)}{2}$$

In the following, we prove that given a tour with local cost $c_\sigma(V(e,l)) = (7+u)$, then, there is an assignment that leaves at most u equations unsatisfied out of $x_i \oplus x_{i+1} = 0$, $x_i \oplus x_j = 0$ and $x_j \oplus x_{j+1} = 0$.

1.Case $\psi_\sigma(x_i) \oplus \psi_\sigma(x_{i+1}) = 0$, $\psi_\sigma(x_i) \oplus \psi_\sigma(x_j) = 0$ and $\psi_\sigma(x_j) \oplus \psi_\sigma(x_{j+1}) = 0$: Notice that the local cost of σ in $V(e,l)$ can be bounded from below by 7, since each of the 7 vertices needs to be visited. Assuming $\psi_\sigma(x_i) = \psi_\sigma(x_j) = \psi_\sigma(x_j) = \psi_\sigma(x_{j+1}) = 1$, the tour σ traverses (v_i^{l1}, v_{i+1}^{l0}), (v_j^{l1}, v_e^{lj}), $(v_e^{lj}, v_e^{l\perp})$, $(v_e^{l\perp}, v_e^{l(i+1)})$ and $(v_e^{l(i+1)}, v_{j+1}^{l0})$. The case $\psi_\sigma(x_i)$=$\psi_\sigma(x_{i+1})$=$\psi_\sigma(x_j)$=$\psi_\sigma(x_{j+1})$=0 can be discussed analogously. In both cases, we obtain $c_\sigma(V(e,l)) = 7$ while ψ_σ is satisfying all 3 equations.

2.Case $\psi_\sigma(x_i) \oplus \psi_\sigma(x_{i+1}) = 0$, $\psi_\sigma(x_i) \oplus \psi_\sigma(x_j) = 1$ and $\psi_\sigma(x_j) \oplus \psi_\sigma(x_{j+1}) = 0$: In both cases, we have $c_\sigma(V(e,l)) \geq 8$, which corresponds to the fact that ψ_σ leaves the equation $x_i \oplus x_j = 0$ unsatisfied. Note that a similar situation holds in case of $\psi_\sigma(x_i) = \psi_\sigma(x_{i+1}) = 0$ and $\psi_\sigma(x_j) = \psi_\sigma(x_{j+1}) = 1$.

3.Case $\psi_\sigma(x_i) \oplus \psi_\sigma(x_{i+1}) = 0$, $\psi_\sigma(x_i) \oplus \psi_\sigma(x_j) = 0$ and $\psi_\sigma(x_j) \oplus \psi_\sigma(x_{j+1}) = 1$: Given $\psi_\sigma(x_i) = \psi_\sigma(x_{i+1}) = 1$ and $\psi_\sigma(x_j) \neq \psi_\sigma(x_{j+1}) = 0$, we are forced to use two 2-arcs increasing the cost by 1. Thus, we obtain $c_\sigma(V(e,l)) \geq 8$. The case $\psi_\sigma(x_i) = \psi_\sigma(x_{i+1}) = 0$

and $\psi_\sigma(x_j) \neq \psi_\sigma(x_{j+1}) = 1$ can be analyzed analogously. A similar argumentation holds for $\psi_\sigma(x_i) \oplus \psi_\sigma(x_{i+1}) = 1$, $\psi_\sigma(x_i) \oplus \psi_\sigma(x_j) = 0$ and $\psi_\sigma(x_j) \oplus \psi_\sigma(x_{j+1}) = 0$.

4.Case $\psi_\sigma(x_i) \oplus \psi_\sigma(x_{i+1}) = 1$**,** $\psi_\sigma(x_i) \oplus \psi_\sigma(x_j) = 0$ **and** $\psi_\sigma(x_j) \oplus \psi_\sigma(x_{j+1}) = 1$**:** Given $\psi_\sigma(x_i) \neq \psi_\sigma(x_{i+1}) = 0$ and $\psi_\sigma(x_j) \neq \psi_\sigma(x_{j+1}) = 0$, we are forced to use four 2-arcs in order to connect all vertices. Consequently, it yields $c_\sigma(V(e,l)) \geq 9$. The case, in which $\psi_\sigma(x_i) \neq \psi_\sigma(x_{i+1}) = 0$ and $\psi_\sigma(x_j) \neq \psi_\sigma(x_{j+1}) = 0$ holds, can be discussed analogously.

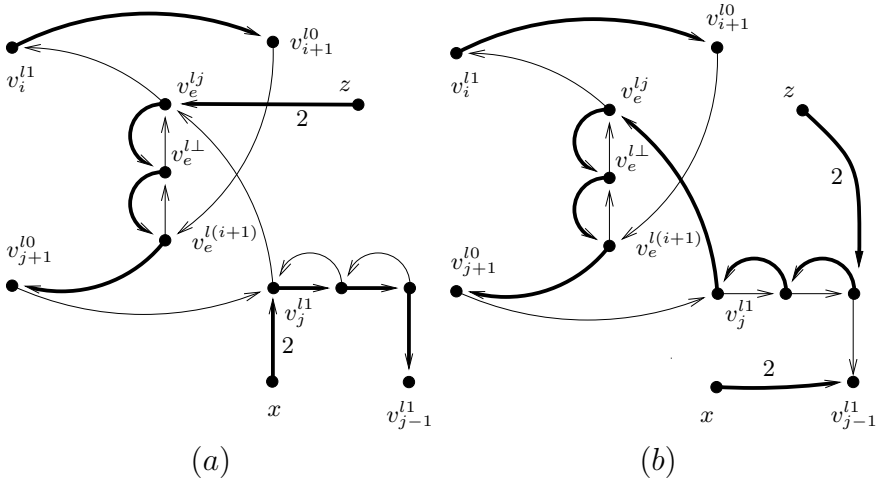


Figure 8: 5.Case with $\psi_\sigma(x_i) = \psi_\sigma(x_{i+1}) = 1$ and $\psi_\sigma(x_j) \neq \psi_\sigma(x_{j+1}) = 1$.

5.Case $\psi_\sigma(x_i) \oplus \psi_\sigma(x_{i+1}) = 0$**,** $\psi_\sigma(x_i) \oplus \psi_\sigma(x_j) = 1$ **and** $\psi_\sigma(x_j) \oplus \psi_\sigma(x_{j+1}) = 1$**:** Let σ be characterized by $\psi_\sigma(x_i) = \psi_\sigma(x_{i+1}) = 1$ and $\psi_\sigma(x_j) \neq \psi_\sigma(x_{j+1}) = 1$. We may assume that σ uses the arc (v_i^{l1}, v_{i+1}^{l0}). The corresponding situation is illustrated in Figure 8 (a). We transform σ such that it traverses the parity graph P_j^l in the other direction and obtain $\psi_\sigma(x_j) = 1$. This transformation induces a tour with at most the same cost. On the other hand, the corresponding assignment ψ_σ satisfies at least $2 - 1$ more equations since $x_j^l \oplus x_{j-1}^l = 0$ might get unsatisfied. In this case, we associate the local costs of 6 with σ. In the other case, in which $\psi_\sigma(x_i) = \psi_\sigma(x_{i+1}) = 0$ and $\psi_\sigma(x_j) \neq \psi_\sigma(x_{j+1}) = 0$ or $\psi_\sigma(x_i) \oplus \psi_\sigma(x_{i+1}) = 1$, $\psi_\sigma(x_i) \oplus \psi_\sigma(x_j) = 1$ and $\psi_\sigma(x_j) \oplus \psi_\sigma(x_{j+1}) = 0$ holds, we may argue similarly.

6.Case $\psi_\sigma(x_i) \oplus \psi_\sigma(x_{i+1}) = 1$**,** $\psi_\sigma(x_i) \oplus \psi_\sigma(x_j) = 1$ **and** $\psi_\sigma(x_j) \oplus \psi_\sigma(x_{j+1}) = 1$**:** Given a tour σ with $\psi_\sigma(x_i) \neq \psi_\sigma(x_{i+1}) = 1$ and $\psi_\sigma(x_j) \neq \psi_\sigma(x_{j+1}) = 0$, we transform σ such that it traverses the parity graph P_j^l in the opposite direction meaning $\psi_\sigma(x_j) = 0$. This transformation enables us to use the arc (v_{j+1}^{l0}, v_j^{l1}). Furthermore, it yields at least one more satisfied equation in $I_\mathcal{H}$. In order to connect the remaining vertices, we are forced to use at least two 2-arcs. In summary, we associate the local cost 9 with this situation in conformity with the at most 2 unsatisfied equations by ψ_σ. The case, in which

$\psi_\sigma(x_i) \neq \psi_\sigma(x_{i+1}) = 0$ and $\psi_\sigma(x_j) \neq \psi_\sigma(x_{j+1}) = 1$ holds, can be discussed analogously.

In summary, given a tour σ passing the graphs displayed in Figure 3 with local cost $(7+u)$, then, we can efficiently modify σ without increasing the total cost such that the assignment ψ_σ leaves at most u equations unsatisfied out of $x_i \oplus x_{i+1} = 0$, $x_i \oplus x_j = 0$ and $x_j \oplus x_{j+1} = 0$.

As for the next step, we are going to analyze the parts of a tour passing graphs corresponding to equations with exactly three variables.

Equations with three variables: Let $x_i^l \oplus x_j^s \oplus x_k^r = 0 = b_c^3$ be an equation with three variables in $I_{\mathcal{H}}$. Let $x_i^l \oplus x_{i+1}^l = 0$, $x_j^s \oplus x_{j+1}^s = 0$ and $x_k^r \oplus x_{k+1}^r = 0$ be the associated cycle equations in $I_{\mathcal{H}}$. For notational simplicity, we set $e = \{i, i+1\}$, $d = \{k, k+1\}$ and $b = \{j, j+1\}$. Let us also define local cost of σ in $V_c^{3\oplus} = \{s_c, s_{c+1}\} \cup V_c$, where $V_c = \{v_c^1, v_c^2, v_c^3\} \cup V(P_e^l) \cup V(P_a^k) \cup V(P_b^s)$.

$$c_\sigma(V_c^{3\oplus}) = \sum_{(s_c,v)\in\sigma} \frac{d((s_c,v))}{2} + \sum_{(v',s_{c+1})\in\sigma} \frac{d((v',s_{c+1}))}{2} + \sum_{v\in V_c} \sum_{a\in\sigma\colon v\in V(a)} \frac{d(a)}{2}$$

We are going to analyze the the number of satisfied equations by ψ_σ in dependence to the local cost of σ in V_c^3. In the first step, we transform the tour traversing the graphs P_i^l, P_{i+1}^l and P_e^l such that σ uses the $\psi_\sigma(x_i^l)$-traversal of P_e^l. We also apply similar modifications to the graphs P_b^s and P_d^r. Afterwards, due to the construction of $D_c^{3\oplus}$ and Lemma 1, we are able to transform σ such that it has local cost in $V_c^{3\oplus}$ of $3 \cdot 3 + 4$ if it passes an even number of parity graphs $P \in \{P_e^l, P_d^k, P_b^s\}$ while using a simple path through $D_c^{3\oplus}$. Otherwise, it yields $c_\sigma(V_c^{3\oplus}) \geq 13 + 1$.

In the following, we are going to analyze the gadgets corresponding to the equation $x_i^l \oplus x_{i+1}^l = 0$, where x_i^l also appears in $x_i^l \oplus x_j^s \oplus x_k^r = 0$. For this, we introduce local cost of σ in $V(e,l) = \{v_i^{l1}, v_{i+1}^{l0}\}$.

$$c_\sigma(V(e,l)) = \sum_{v\in V(e,l)} \sum_{a\in\sigma\colon v\in V(a)} \frac{d(a)}{2}$$

In particular, we are going to prove that given a tour σ with local cost $c_\sigma(V(e,l)) = 2$, then, we can efficiently modify σ without increasing the total cost such that the assignment ψ_σ satisfies $x_i^l \oplus x_{i+1}^l = 0$.

1. Case ($\psi_\sigma(x_i^l) \oplus \psi_\sigma(x_{i+1}^l) = 0$): Note that $c_\sigma(V(e,l))$ is bounded from below by 2. In both cases, we transform the tour such that it uses the $\psi_\sigma(x_i^l)$-traversal of P_e^l without increasing its total cost. Exemplary, we display such a scenario for the case $\psi_\sigma(x_i^l) = 1$ and $\psi_\sigma(x_{i+1}^l) = 1$ in Figure 9 (a) and the transformed tour in Figure 9 (b). In both cases, we have $c_\sigma(V(e,l)) = 2$.

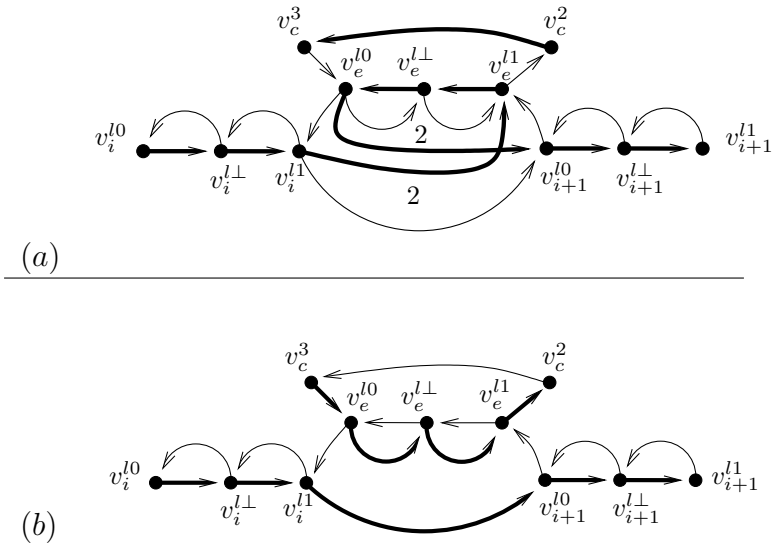


Figure 9: Case ($\psi_\sigma(x_i^l) = 1$ and $\psi_\sigma(x_{i+1}^l) = 1$).

2.Case ($\psi_\sigma(x_i^l) = 1$ and $\psi_\sigma(x_{i+1}^l) = 0$): Let us assume that $\psi_\sigma(x_i^l) \oplus \psi_\sigma(x_j^s) \oplus \psi_\sigma(x_k^r) = 0$ holds. Due to Lemma 1, it is possible to transform the tour such that it uses the 0-traversal of the parity graph P_e^l without increasing the total cost. In the other case, i.e. $\psi_\sigma(x_i^l) \oplus \psi_\sigma(x_j^s) \oplus \psi_\sigma(x_k^r) = 1$, we will change the value of $\psi_\sigma(x_i^l)$ achieving in this way at least $2 - 1$ more satisfied equation. Let us examine the scenario and the corresponding transformation in Figure 10 (a) and (b), respectively. The tour uses the 0-traversal of the parity graph P_e^l, which enables σ to pass the parity check in $D_c^{3\oplus}$. In both cases, we obtain $c_\sigma(V(e,l)) \geq 3$ in conformity with the 1 unsatisfied equation.

3.Case $\psi_\sigma(x_i^l) = 0$ and $\psi_\sigma(x_{i+1}^l) = 1$: Assuming $\psi_\sigma(x_i^l) \oplus \psi_\sigma(x_i^l) \oplus \psi_\sigma(x_i^l) = 0$, the tour will be modified such that the parity graphs P_i^l and P_e^l are traversed in the same direction. Since we have $\psi_\sigma(x_i^l) \oplus \psi_\sigma(x_i^l) \oplus \psi_\sigma(x_i^l) = 0$, we are able to uncouple the parity graph P_e^l from the tour through $D_c^{3\oplus}$ without increasing its total cost.

Assuming $\psi_\sigma(x_i^l) \oplus \psi_\sigma(x_i^l) \oplus \psi_\sigma(x_i^l) = 1$, we transform σ such that the parity graph P_e^l is traversed when σ is passing through $D_c^{3\oplus}$ meaning $v_c^3 \to v_e^{l0} \to v_e^{l\perp} \to v_e^{l1} \to v_c^2$ is a part of the tour. In addition, we change the value of $\psi_\sigma(x_i^l)$ yielding at least $2 - 1$ more satisfied equations. In both cases, we associate the local cost of 3 with σ. On the other hand, ψ_σ leaves at most one equation unsatisfied.

In summary, given a tour σ passing the graphs corresponding to $x_i^l \oplus x_{i+1}^l = 0$, $x_j^s \oplus x_{j+1}^s = 0$, $x_k^r \oplus x_{k+1}^r = 0$ and $x_i^l \oplus x_j^s \oplus x_k^r = 0$, it is possible to transform σ in polynomial time into a tour π without increasing the total cost such that sum of the local cost of π in $V_c^{3\oplus}$, $V(e,l)$, $V(a,k)$ and $V(b,s)$ is $4+3\cdot 3+3\cdot 2+u$ and the number of unsatisfied equation by ψ_π out of $x_i^l \oplus x_{i+1}^l = 0$, $x_j^s \oplus x_{j+1}^s = 0$, $x_k^r \oplus x_{k+1}^r = 0$ and $x_i^l \oplus x_j^s \oplus x_k^r = 0$ is bounded by u.

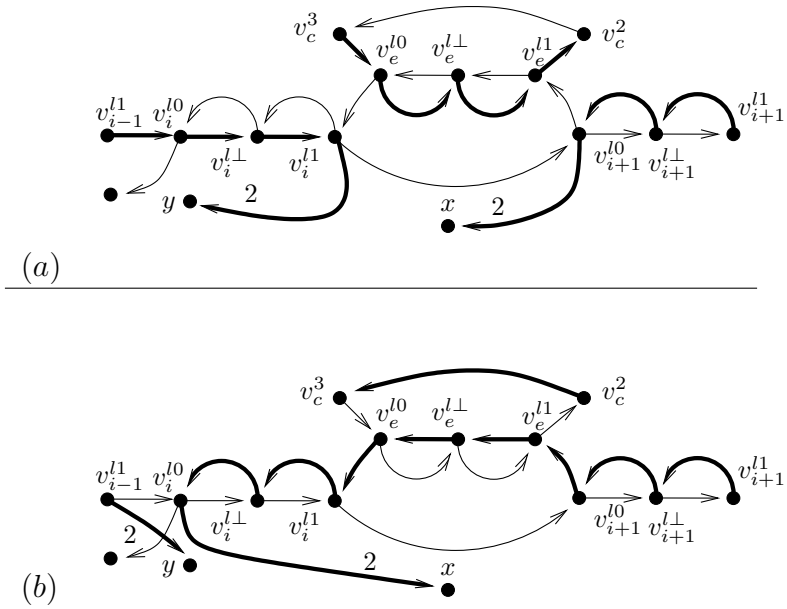


Figure 10: Case $\psi_\sigma(x_i^l) = 1$, $\psi_\sigma(x_{i+1}^l) = 0$ and $\psi_\sigma(x_i^l) \oplus \psi_\sigma(x_j^s) \oplus \psi_\sigma(x_k^r) = 1$.

Cycle border equations: Let $x_1^l \oplus x_2^l = 0$ be a cycle border equation. We also define local cost of a tour σ in $V_b^l = \{b_l, b_{l+1}\} \cup V_l^{12}$, where $V_l^{12} = \{v_2^{l0}, v_1^{l1}\}$.

$$c_\sigma(V_b^l) = \sum_{(b_l,v_2)\in\sigma} \frac{d((b_l,w))}{2} + \sum_{(v_1,b_{l+1})\in\sigma} \frac{d((b_l,w))}{2} + \sum_{v\in V_l^{12}} \sum_{a\in\sigma:\, v\in V(a)} \frac{d(a)}{2}$$

Let us analyze the local cost of σ.
Case $\psi_\sigma(x_1^l) \oplus \psi_\sigma(x_2^l) = 0$**:** Since we account half the length of arcs that are leaving b_l and entering b_{l+1} and also half the length of arcs that are leaving and entering v_2^{l0} and v_1^{l1}, $c_\sigma(V_b^l)$ is bounded from below by $1 + 1 + 2 \cdot 0.5$. In both cases ($\psi_\sigma(x_1^l) = \psi_\sigma(x_2^l) \in \{0, 1\}$), it is possible to construct a tour with $c_\sigma(V_b^l) = 3$.
Case $\psi_\sigma(x_1^l) \neq \psi_\sigma(x_2^l)$**:** Let us assume that $\psi_\sigma(x_1^l) \neq \psi_\sigma(x_2^l) = 0$ holds. We note that there is a vertex $v \in \{v_2^{l0}, v_1^{l1}\}$ such that σ is using a 2-arc to enter v. Also, in order to enter the vertex b_{l+1}, we see that we need to use another 2-arc. It implies that $c_\sigma(V_b^l) \geq 4$. In the case ($\psi_\sigma(x_1^l) \neq \psi_\sigma(x_2^l) = 1$), we may argue similarly and get $c_\sigma(V_b^l) \geq 4$.

In summary, given a tour σ passing the graphs corresponding to $x_1^l \oplus x_2^l = 0$, it is possible to transform σ in polynomial time into a tour π without increasing the total cost such that π has local cost in V_b^l of 3 if the equation $x_1^l \oplus x_2^l = 0$ is satisfied by ψ_σ and at least 4 otherwise.
Finally, we note that half of the weight of the arcs entering and leaving a vertex $v^\perp$ of a parity gadget is bounded from below by 1.

Suppose that we have given a tour σ in $D_{\mathcal{H}}^{12}$ with total cost $206\nu + n + 2 + \delta \cdot \nu$.

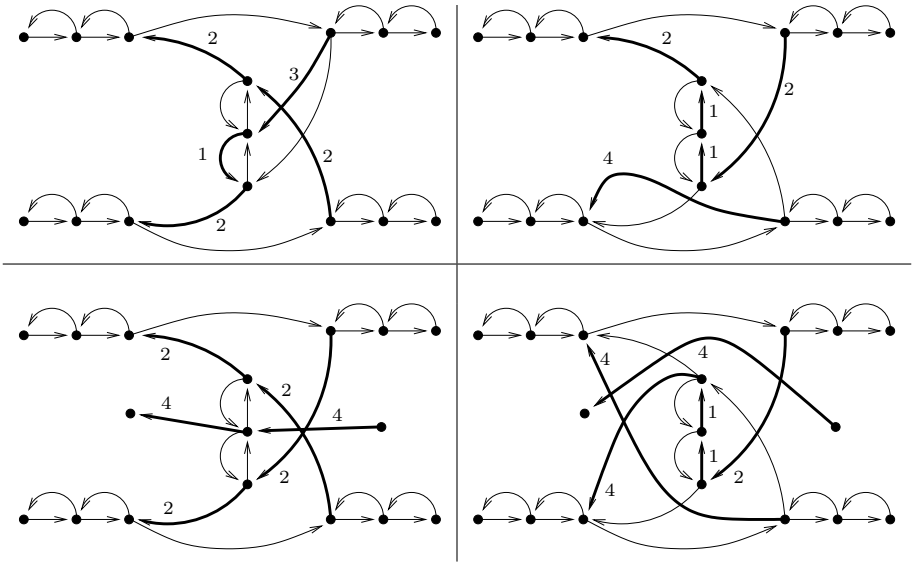


Figure 11: It is possible to transform the traversals in the left columns into the traversals in the right columns without increasing the total cost of the tour.

We are going to subtract the lower bound on the local cost for each part of the tour. It implies that the number of unsatisfied equations is bounded from above by

$$206\nu + n + 2 + \delta \cdot \nu - (3 \cdot 60\nu + 2\nu \cdot 13 + n + 2) = \delta \cdot \nu$$

and the proof of the lemma follows. □

We are ready to give the proof of Theorem 1.(i).

Proof of Theorem 1.(i). Given an instance I_1 of the MAX-E3LIN2 problem with n variables and 2γ equations, for all $\delta > 0$, there exists a constant k_δ such that if we repeat each equation k_δ time, we get an instance $I_1^{(k)}$ with $2\nu = k2\gamma$ equations and n variables such that $(n+2)/2\nu \leq \delta$ holds.

Then, from $I_1^{(k)}$, we generate an instance $I_{\mathcal{H}}$ of the Hybrid problem and the corresponding directed graph $D_{\mathcal{H}}^{12}$. Due to Lemmata 2 and 4, and Theorem 3, we know that for all $\epsilon > 0$, it is **NP**-hard to tell whether there is a tour with cost at most $206\nu + n + 2 + \epsilon \cdot \nu \leq 206 \cdot \nu + \alpha\nu$ or all tours have cost at least $206 \cdot \nu + (1-\epsilon)\nu + n + 2 \geq 207 \cdot \nu - \alpha \cdot \nu$, for some α that depends only on α and δ. Finally, we note that the ratio between these two cases can get arbitrarily close to $207/206$ by appropriate choices for ϵ and δ. □

8 The $(1,4)$-ATSP

In order to prove the claimed hardness results for the $(1,4)$-ATSP, we use the same construction as described in Section 7 with the difference that all arcs in parity graphs have

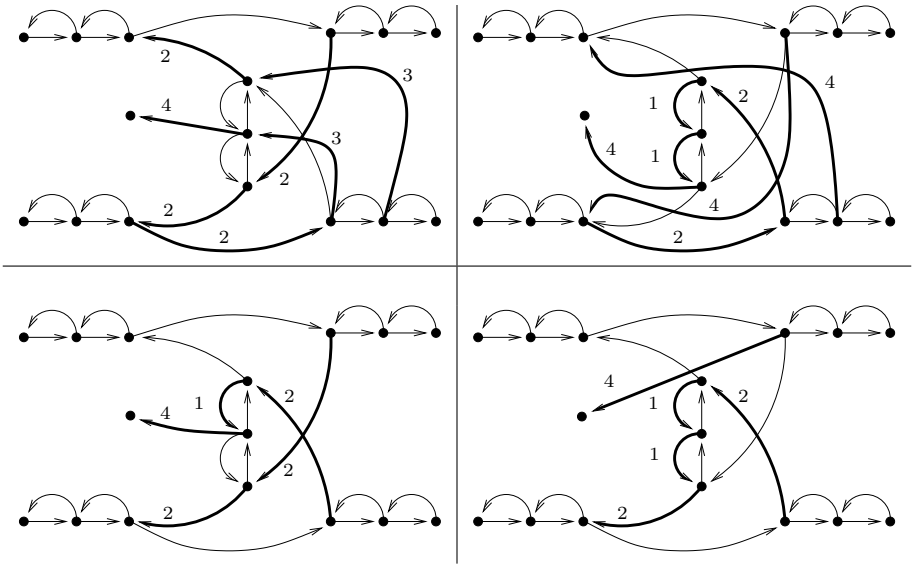


Figure 12: It is possible to transform the traversals in the left columns into the traversals in the right columns without increasing the total cost of the tour.

weight 1, whereas all other arcs contained in the directed graph $D^{12}_{\mathcal{H}}$ obtain the weight 2. Let us call this new weighted graph $D^{14}_{\mathcal{H}}$. The induced asymmetric metric space $(V_{\mathcal{H}}, d_{\mathcal{H}})$ is given by $V_{\mathcal{H}} = V(D^{14}_{\mathcal{H}})$ with distance function $d_{\mathcal{H}}$ which is defined by the shortest path metric in $D^{14}_{\mathcal{H}}$ bounded by the value 4.
In other words, given $x, y \in V_{\mathcal{H}}$, the distance between x and y in $V_{\mathcal{H}}$ is

$$d_{\mathcal{H}}(x, y) = \min\{\text{length of a shortest path from } x \text{ to } z \text{ in } D^{14}_{\mathcal{H}},\ 4\}.$$

The only difficulty that remains is to prove that every tour in $(V_{\mathcal{H}}, d_{\mathcal{H}})$ can be transformed efficiently into a consistent tour without increasing the total cost. This statement can be proved by considering all possibilities exhaustively. Some cases are displayed in Figure 11 and 12.

We are ready to give the proof of Theorem 1.(iii).

Proof of Theorem 1.(iii). Suppose we are given $I_{\mathcal{H}}$ an instance of the Hybrid problem consisting of n wheels, 60ν equations with two variables and 2ν equations with three variables, we construct in polynomial time the associated instance $(V_{\mathcal{H}}, d_{\mathcal{H}})$ of the $(1, 4)$-ATSP.

Let ϕ an assignment to the variables of $I_{\mathcal{H}}$ leaving $\delta \cdot \nu$ equations unsatisfied in $I_{\mathcal{H}}$ with $\delta \in (0, 1)$. Then, there exists a tour with total cost at most

$$60\nu \cdot (2 + 2) + 2\nu \cdot (3 \cdot 4 + 2 \cdot 4) + 2 \cdot (\delta \cdot \nu) + 2(n + 2).$$

On the other hand, if we are given a tour σ in $(V_{\mathcal{H}}, d_{\mathcal{H}})$ with total cost $280 \cdot \nu + 2 \cdot (\delta \cdot \nu) + 2(n + 2)$, it is possible to transform σ in polynomial time into a tour π without increasing

the total cost such that the associated assignment ψ_π leaves at most $\delta \cdot \nu$ equations in $I_\mathcal{H}$ unsatisfied.

Similarly to the proof of Theorem 1, we may assume that $2(n+2)/\nu \leq \tau$ holds. According to Theorem 3, we know that for all $\epsilon > 0$, it is **NP**-hard to decide whether there is a tour with total cost at most $280\nu + 2(n+2) + 2\epsilon \cdot \nu \leq 280 \cdot \nu + 2\alpha\nu$ or all tours have total cost at least $280 \cdot \nu + 2(1-\epsilon)\nu + 2(n+2) \geq 282 \cdot \nu - 2\alpha \cdot \nu$, for some α which depends only on ϵ and τ. The ratio between these two cases can get arbitrarily close to $282/280 = 141/140$ by appropriate choices for ϵ and τ. □

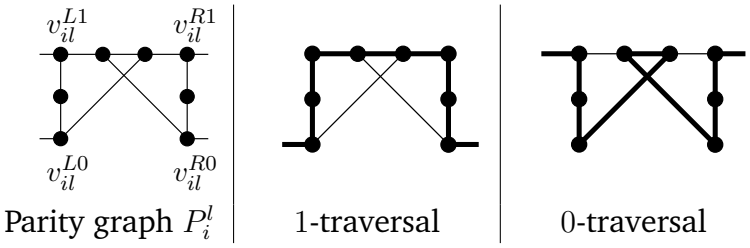


Figure 13: $0/1$-Traversals of the parity gadget P_i^l.

9 The $(1,2)$-TSP

In order to prove Theorem 1 (ii), we apply the reduction method used in the previous section to the $(1,2)$-TSP. As for parity gadgets, we use the graph depicted in Figure 13 with its corresponding traversals. The traversed edges are displayed by thick lines. The vertices in the set $\{v_{il}^{L1}, v_{il}^{R1}, v_{il}^{L0}, v_{il}^{R0}\}$ are called contact vertices of P_i^l.

Let $I_\mathcal{H}$ be an instance of the hybrid problem produced in the reduction in Theorem 3. Let $x_i^l \oplus x_j^l = 0$ be a matching equation in $I_\mathcal{H}$, and $x_i^l \oplus x_{i+1}^l = 0$ and $x_j^l \oplus x_{j+1}^l = 0$ the corresponding the cycle equations. We connect the associated parity gadgets P_i^l, P_{i+1}^l, $P_{\{i,j\}}^l$, P_j^l and P_{j+1}^l as displayed in Figure 14.

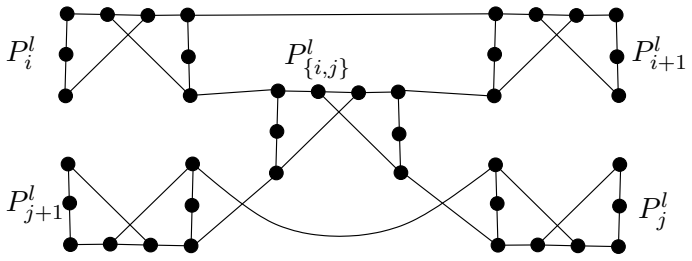


Figure 14: The construction simulating $x_i^l \oplus x_j^l = 0$, $x_i^l \oplus x_{i+1}^l = 0$ and $x_j^l \oplus x_{j+1}^l = 0$.

For equations with three variables $x \oplus y \oplus z = 0 = b_c^3$ in $I_{\mathcal{H}}$, we use the graph $G_c^{3\oplus}$ depicted in Figure 15. Engebretsen and Karpinski [EK06] used this graph in their reduction and proved the following statement.

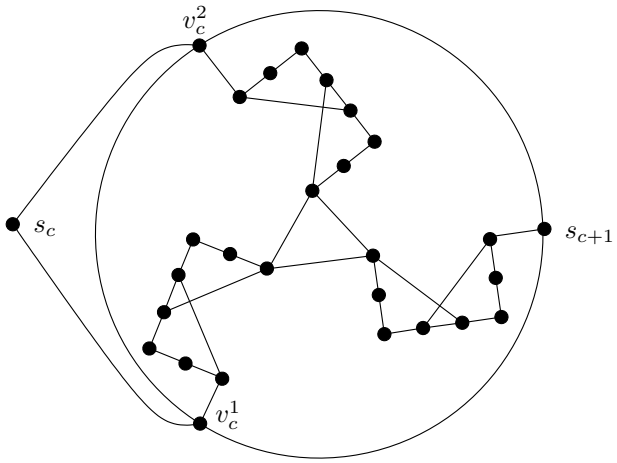


Figure 15: Graph $G_c^{3\oplus}$ simulating to $x \oplus y \oplus z = 0$.

Lemma 5 ([EK06])**.** *There is a simple path from s_c to s_{c+1} in Figure 15 containing the vertices $v \in \{v_c^1, v_c^2\}$ if and only if an even number of parity gadgets is traversed.*

Let $x_i \oplus x_{i+1} = 0$ be a cycle equation and $x_i \oplus y \oplus z = 0$ an equation with three variables in $I_{\mathcal{H}}$. We denote the parity graphs that simulate $x_i \oplus x_{i+1} = 0$ by P_i^l and P_{i+1}^l. Furthermore, we denote the parity gadget for x_i as $P_{(x,i)}$, which is a subgraph in $G_c^{3\oplus}$ simulating $x_i \oplus y \oplus z = 0$. Then, we connect v_{il}^{R1} with $v_{(i+1)l}^{L1}$ and v_{il}^{R0} with $v_{(x,i)}^{R1}$. Finally, we create $\{v_{(i+1)l}^{L0}, v_{(x,i)}^{L1}\}$.

For each cycle border equation $x_1^l \oplus x_2^l = 0$ in $I_{\mathcal{H}}$, we introduce the path $p_l = b_l^1 - b_l^2 - b_l^3$. In addition, we connect b_l^3 and b_{l+1}^1 to the parity gadgets P_1^l and P_2^l in a similar way as in the reduction from the Hybrid problem to the $(1,2)$-ATSP. More precisely, we add the edges $\{b_l, v_{2l}^{L0}\}$, $\{b_{l+1}, v_{1l}^{R0}\}$, $\{b_{l+1}, v_{2l}^{L1}\}$ and $\{b_l, v_{1l}^{R1}\}$.

For the last cycle equation $x_1^n \oplus x_2^n = 0$, we create the path $b_{n+1}^1 - b_{n+1}^2 - s_1$, where s_1 is a vertex of $G_1^{3\oplus}$. This is the whole description of the corresponding graph $G_{\mathcal{H}}$.

We are ready to give the proof of Theorem 1.(ii).

Proof of Theorem 1.(ii). Let $I_{\mathcal{H}}$ be an instance of the Hybrid problem consisting of n wheels, 60ν equations with two variables, 2ν equations with three variables. and $G_{\mathcal{H}}^{12}$ the associated instance of the $(1,2)$-TSP.

Suppose we are given an assignment ϕ to the variables of $I_{\mathcal{H}}$ leaving $\delta \cdot \nu$ equations unsatisfied with $\delta \in (0,1)$, then, we can construct a tour $G_{\mathcal{H}}^{12}$ with total cost at most $8 \cdot 60\nu + (3 \cdot 8 + 3) \cdot 2\nu + 3(n+1) + 1 + \delta \cdot \nu$.

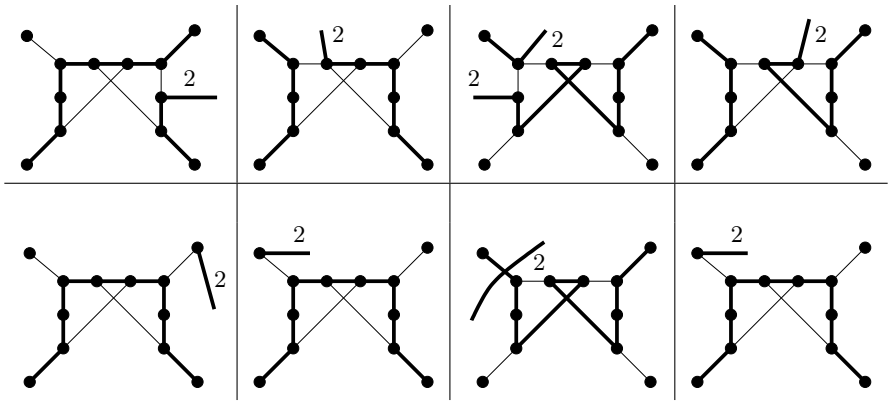
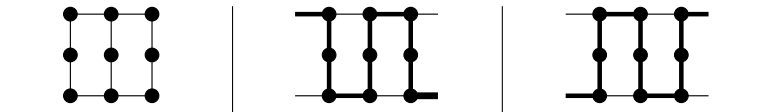


Figure 16: It is possible to transform the traversals in the upper row into the traversals in the lower row without increasing the total cost of the tour.

On the other hand, if we are given a tour σ in $G^{12}_{\mathcal{H}}$ with total cost $534\nu+3(n+1)+1+\delta\cdot\nu$, it is possible to transform σ in polynomial time into a tour π without increasing the total cost such that all parity gadgets in $G^{12}_{\mathcal{H}}$ are traversed by means of $0/1$-traversals. Some cases are displayed in Figure 16. Moreover, we are able to construct in polynomial time an assignment to the variables of $I_{\mathcal{H}}$, which leaves at most $\delta\nu$ equations in $I_{\mathcal{H}}$ unsatisfied.

Similarly to the proof of Theorem 1, we may assume that $(3n+4)/\nu \leq \tau$ holds. According to Theorem 3, we know that for all $\epsilon > 0$, it is **NP**-hard to decide whether there is a tour with total cost at most $534\nu + 3(n+1) + 1 + \epsilon \cdot \nu \leq 534 \cdot \nu + \alpha\nu$ or all tours have total cost at least $534 \cdot \nu + (1-\epsilon)\nu + 3(n+1) + 1 \geq 535 \cdot \nu - \alpha \cdot \nu$, for some α depending only on ϵ and τ. The ratio between these two cases can get arbitrarily close to $535/534$ by appropriate choices for ϵ and τ. □



Parity graph P_i^l | 1-Traversal of P_i^l. | 0-Traversal of P_i^l.

Figure 17: $0/1$-Traversals of the graph P_i^l.

10 Approximation Hardness of the $(1,4)$-TSP

In order to prove the claimed approximation hardness results for the $(1,4)$-TSP, we cannot use the same parity graphs as in the construction in the previous section since tours are

not necessarily consistent in this metric. For this reason, we introduce the parity gadgets displayed in Figure 17 with the corresponding traversals.

In order to define the new instance of the $(1,4)$-TSP, we replace all parity gadgets in $G^{12}_{\mathcal{H}}$ by graphs displayed in Figure 17. In the remainder, we refer to the whole construction as the graph $G^{14}_{\mathcal{H}}$.

The construction for a matching equation $x^l_i \oplus x^l_j = 0$, and its associated cycle equations $x^l_i \oplus x^l_{i+1} = 0$ and $x^l_j \oplus x^l_{j+1} = 0$ are displayed in Figure 18.

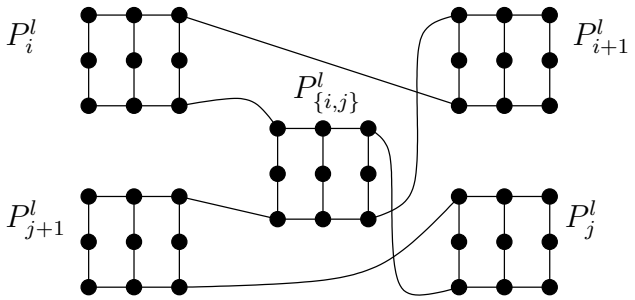


Figure 18: Construction simulating $x^l_i \oplus x^l_j = 0$, $x^l_i \oplus x^l_{i+1} = 0$ and $x^l_j \oplus x^l_{j+1} = 0$.

All edges contained in a parity graph have weight 1, whereas all other edges have weight 2. The remaining distances in the associated metric space $V_{\mathcal{H}}$ are induced by the metric in $G^{14}_{\mathcal{H}}$ bounded by the value 4 meaning

$$d_{\mathcal{H}}(\{x,y\}) = \min\{\text{length of a shortest path from } x \text{ to } y \text{ in } G^{14}_{\mathcal{H}},\ 4\}.$$

This is the whole description of the associated instance $(V_{\mathcal{H}}, d_{\mathcal{H}})$ of the $(1,4)$-TSP. We are ready to give the proof of Theorem 1.(iv).

Proof of Theorem 1.(iv) . Suppose we are given $I_{\mathcal{H}}$ an instance of the Hybrid problem consisting of n wheels, 60ν equations with two variables and 2ν equations with three variables, we construct in polynomial time the associated instance $(V_{\mathcal{H}}, d_{\mathcal{H}})$ of the $(1,4)$–TSP problem.

For a constant $\delta \in (0,1)$, we are given an assignment ϕ to the variables of $I_{\mathcal{H}}$ leaving $\delta \cdot \nu$ equations unsatisfied in $I_{\mathcal{H}}$. Then, we can construct a tour in $(V_{\mathcal{H}}, d_{\mathcal{H}})$ with total cost at most $60\nu \cdot (2+8) + 2\nu \cdot (3 \cdot 10 + 2 \cdot 3) + 6n + 8 + 2 \cdot \delta\nu$.

On the other hand, if we are given a tour σ in $(V_{\mathcal{H}}, d_{\mathcal{H}})$ with length $672\nu + 6n + 8 + 2 \cdot \delta\nu$, it is possible to transform σ in polynomial time into a tour π such that it uses $0/1$-traversals of all parity graphs contained in $G^{14}_{\mathcal{H}}$ without increasing the total cost. Some cases are displayed in Figure 19 and 20. Furthermore, we are able to construct in polynomial time an assignment to the variables of $I_{\mathcal{H}}$, which leaves at most $2\delta\nu$ equations in $I_{\mathcal{H}}$ unsatisfied.

Similarly to the proof of Theorem 1.(i), we may assume that $(6n+8)/\nu \leq \tau$ holds. According to Theorem 3, we know that for all $\epsilon > 0$, it is **NP**-hard to decide whether there is a tour with total cost at most $672\nu + 6n + 8 + \epsilon \cdot 2 \cdot \nu \leq 672 \cdot \nu + \alpha \cdot 2\nu$ or all tours have cost at least $672 \cdot \nu + (1-\epsilon) \cdot 2\nu + 6n + 8 \geq 674 \cdot \nu - \alpha 2 \cdot \nu$, for some α that depends only on

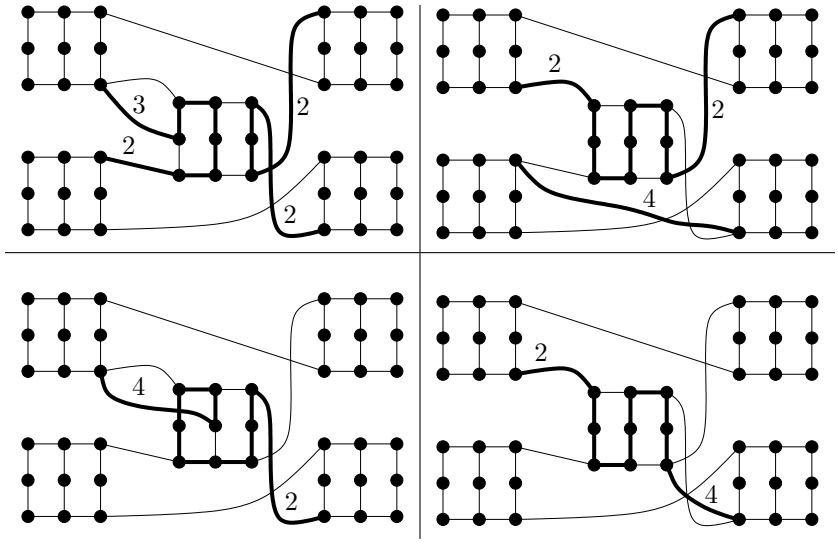


Figure 19: It is possible to transform the traversals in the left column into the traversals in the right column without increasing the total cost of the tour.

ϵ and τ. The ratio between these two cases can get arbitrarily close to $674/672 = 337/336$ by appropriate choices for ϵ and τ. □

11 The Shortest Superstring Problem

In this section, we construct a reduction from the Hybrid problem to the SSP and give the poof of Theorem 1.(vi). Let us first give a high-level view of the reduction in order to build some intuition.

11.1 Main Idea of the Reduction

Let $x_i \oplus x_{i+1} = 0$ be a cycle equation of a given instance $I_{\mathcal{H}}$ of the Hybrid problem. A parity gadget for $x_i \oplus x_{i+1} = 0$ in the instance $\mathcal{S}_{\mathcal{H}}$ of the SSP is a subset $S_i \subset \mathcal{S}_{\mathcal{H}}$ containing two strings s_i^1 and s_i^2. The strings s_i^1 and s_i^2 can be overlapped by two letters in two different ways: $s_i^1 \xrightarrow{2} s_i^2$ or $s_i^2 \xrightarrow{2} s_i^1$. We refer to those special alignments as $0/1$-*alignments* and they define the value that we will assign to x_i. The basic idea of the construction is the following: The $0/1$-alignments of two consecutive cycle equations are structured in such a way that they can be overlapped by one letter if they use the same $0/1$-alignment. The parity information of the variables in the Hybrid problem will be transported by means of $0/1$-alignments to the gadgets that are simulating equations with exactly three variables. If the underlying equation with three variables is left unsatisfied by the assignment defined

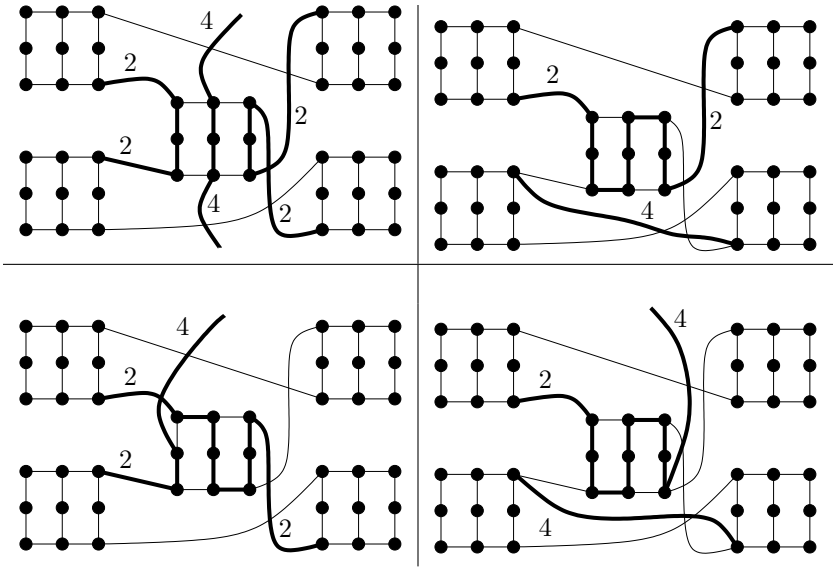


Figure 20: It is possible to transform the traversals in the left column into the traversals in the right column without increasing the total cost of the tour.

via $0/1$-alignments, it is not possible to overlap all the $0/1$-alignments of the variables, that are involved, by one letter implying a longer superstring for $\mathcal{S}_{\mathcal{H}}$.

11.2 Description of the Instance $\mathcal{S}_{\mathcal{H}}$ of the SSP

Given an instance $I_{\mathcal{H}}$ of the Hybrid problem, we define a set of corresponding strings for every equation in $I_{\mathcal{H}}$ and their specific alignment. We assume that all variables in $I_{\mathcal{H}}$ appear unnegated in equations. Therefore, equations with three variables are of the form $x \oplus y \oplus z = 0$ or $x \oplus y \oplus z = 1$. We begin with the description of the strings corresponding to cycle border equations.

Strings for cycle border equations: Given a cycle border equation $x_1 \oplus x_2 = 0$, we introduce six associated strings. The string $L_x C_x^l$ is used as the initial part of the superstring corresponding to this wheel, whereas $C_x^r R_x$ is used as the end part. Furthermore, we introduce strings that represent an assignment that sets either the variable x_1 to 1 or the variable x_2 to 0: $C_x^l x_2^{m0} x_1^{l1} C_x^r$ and $x_1^{l1} C_x^r C_x^l x_2^{m0}$. The strings, that represent an assignment that sets either the variable x_1 to 0 or the variable x_2 to 0, are given by $C_x^l x_2^{r1} x_1^{m0} C_x^r$ and $x_1^{m0} C_x^r C_x^l x_2^{r1}$. The following two alignments are called the 0-*alignment* of the four strings. $C_x^l x_2^{m0} x_1^{l1} C_x^r \overset{2}{\rightarrow} x_1^{l1} C_x^r C_x^l x_2^{m0}$ and $x_1^{m0} C_x^r C_x^l x_2^{r1} \overset{2}{\rightarrow} C_x^l x_2^{r1} x_1^{m0} C_x^r$. On the other hand, we the define the 1-*alignment* as $x_1^{l1} C_x^r C_x^l x_2^{m0} \overset{2}{\rightarrow} C_x^l x_2^{m0} x_1^{l1} C_x^r$ and $C_x^l x_2^{r1} x_1^{m0} C_x^r \overset{2}{\rightarrow} x_1^{m0} C_x^r C_x^l x_2^{r1}$. Next, we describe the strings corresponding to matching equations.

Strings for matching equations: Let $x_i \oplus x_j = 0$ be a matching equation in $I_{\mathcal{H}}$ with $i < j$.

Then, we introduce two strings of the form $x_j^{r0}x_j^{l0}x_i^{r1}x_i^{l1}$ and $x_i^{r1}x_i^{l1}x_j^{r0}x_j^{l0}$. We define the 0-*alignment* and 1-*alignment* as $x_j^{r0}x_j^{l0}x_i^{r1}x_i^{l1} \xrightarrow{2} x_i^{r1}x_i^{l1}x_j^{r0}x_j^{l0}$ and $x_i^{r1}x_i^{l1}x_j^{r0}x_j^{l0} \xrightarrow{2} x_j^{r0}x_j^{l0}x_i^{r1}x_i^{l1}$, respectively.

Strings for Equations with Three Variables: We now concentrate on equations with exactly three variables. Let $x \oplus y \oplus z = b_j^3$ be an equation with three variables in $I_{\mathcal{H}}$. For every equation $x \oplus y \oplus z = b_j^3$, we define two corresponding sets $S^{\alpha}(b_j^3)$ and $S^{\beta}(b_j^3)$, both containing three strings. Finally, the set $S(b_j^3)$ is defined as the union $S^{\alpha}(b_j^3) \cup S^{\alpha}(b_j^3)$. We distinguish whether $x \oplus y \oplus z = b_j^3$ is of the form $x \oplus y \oplus z = 1$ or $x \oplus y \oplus z = 0$. An equation of the form $x \oplus y \oplus z = 0$ is represented by $S^{\alpha}(b_j^3)$ containing the strings $x^{r1\alpha}x^{l1}y^{r1}y^{l1}$, $y^{r1}y^{l1}x^{m0\alpha}C_j$ and $x^{m0\alpha}C_jx^{r1\alpha}x^{l1}$. $S^{\beta}(b_j^3)$ consists of $x^{r1\beta}x^{l1}z^{r1}z^{l1}$, $z^{r1}z^{l1}C_jx^{m0}$ and $C_jx^{m0}x^{r1\beta}x^{l1}$. The strings in $S^{\alpha}(b_j^3)$ can be overlapped by two letters in a cyclic fashion to obtain three different constellations. A suitable constellation can be used to connect with $0/1$-alignments corresponding to cycle equations. The string $x^{r1\alpha}x^{l1}y^{r1}y^{l1} \xrightarrow{2} y^{r1}y^{l1}x^{m0\alpha}C_j \xrightarrow{2} x^{m0\alpha}C_jx^{r1\alpha}x^{l1} = s_{cj}^{1x}$ represents the assignment $x = 1$, whereas the constellation $y^{r1}y^{l1}x^{m0\alpha}C_j \xrightarrow{2} x^{m0\alpha}C_jx^{r1\alpha}x^{l1} \xrightarrow{2} x^{r1\alpha}x^{l1}y^{r1}y^{l1} = s_{cj}^{1y}$ is representing $y = 1$. Finally, the string $x^{m0\alpha}C_jx^{r1\alpha}x^{l1}y^{r1\alpha}y^{l1}x^{m0\alpha}C_j = s_{cj}^{0xl}$ can be used to overlap with $C_jx^{m0}x^{r1\beta}x^{l1}z^{r1}z^{l1}C_jx^{m0} = s_{cj}^{0xr}$ consisting of the strings in $S^{\beta}(b_j^3)$ in the case $x = 0$, $y = 0$, and $z = 0$. $z^{r1}z^{l1}C_jx^{m0} \xrightarrow{2} C_jx^{m0}x^{r1\beta}x^{l1} \xrightarrow{2} x^{r1\beta}x^{l1}z^{r1}z^{l1} = s_{cj}^{1z}$ is used in the case $z = 1$.
The sets $S^{\alpha}(b_j^3)$ and $S^{\beta}(b_j^3)$ representing equations of the form $x \oplus y \oplus z = 1$ can be constructed analogously. Next, we describe the strings corresponding to cycle equations.

Strings for cycle equations: Let $\mathcal{W}_x$ be a wheel in $I_{\mathcal{H}}$ and $M(\mathcal{W}_x)$ its associated perfect matching. We suppose that $\{i, j\}$ and $\{i+1, j'\}$ are both contained in $M(\mathcal{W}_x)$ with $i < j$. Then, we introduce the corresponding strings for $x_i \oplus x_{i+1} = 0$: If $i + 1 < j'$ holds, we have $x_i^{m0}x_{i+1}^{m0}x_i^{l1}x_{i+1}^{r1}$ and $x_i^{l1}x_{i+1}^{r1}x_i^{m0}x_{i+1}^{m0}$. We define the 0-*alignment* and 1-*alignment* as $x_i^{m0}x_{i+1}^{m0}x_i^{l1}x_{i+1}^{r1} \xrightarrow{2} x_i^{l1}x_{i+1}^{r1}x_i^{m0}x_{i+1}^{m0}$ and $x_i^{l1}x_{i+1}^{r1}x_i^{m0}x_{i+1}^{m0} \xrightarrow{2} x_i^{m0}x_{i+1}^{m0}x_i^{l1}x_{i+1}^{r1}$, respectively. In the case $(i + 1 > j')$, we use $x_i^{m0}x_{i+1}^{r0}x_i^{l1}x_{i+1}^{m1}$ and $x_i^{l1}x_{i+1}^{m1}x_i^{m0}x_{i+1}^{r0}$. The strings for the remaining cases can be defined analogously.
If the variable x_i appears in an equation of the form $x_i \oplus y \oplus z = 0$, we introduce three strings for $x_{i-1} \oplus x_i = 0$: $x_{i-1}^{l1}x_i^{r1\beta}x_{i-1}^{l1}x_i^{r1\alpha}$, $x_{i-1}^{l1}x_i^{r1\alpha}x_{i-1}^{m0}x_i^{m0}$ and $x_{i-1}^{m0}x_i^{m0}x_{i-1}^{l1}x_i^{r1\beta}$. The strings for the case ($x_i \oplus y \oplus z = 1$) can be defined analogously.

11.3 Constructing the Superstring from an Assignment

In this section, we are going to construct a superstring given an assignment to the variables of $I_{\mathcal{H}}$ and give the proof of the following lemma.

Lemma 6. *Let $\delta \in (0, 1)$ be a constant, $I_{\mathcal{H}}$ an instance with n wheels, $60 \cdot \nu$ equations with two variables and $2 \cdot \nu$ equations with exactly three variables and ϕ an assignment leaving*

$\delta \cdot \nu$ equations in $I_{\mathcal{H}}$ unsatisfied. Then, it is possible to construct efficiently a superstring for $\mathcal{S}_{\mathcal{H}}$ with length at most $332 \cdot \nu + 8 \cdot n + \delta \cdot \nu$ and compression at least $4n + 204 \cdot \nu - \delta \cdot \nu$.

Proof. For a fixed constant $\delta \in (0, 1)$, let ϕ be an assignment to the variables of $I_{\mathcal{H}}$ leaving $\delta \cdot \nu$ equations unsatisfied. According to Theorem 3, we may assume that all variables that are associated to a wheel take the same value under ϕ. Let $\{\mathcal{W}_p\}_{p=1}^{n}$ be the set of wheels associated with $I_{\mathcal{H}}$. For each wheel $\mathcal{W}_p$, we are going to construct the corresponding string s_ϕ^p according to the assignment ϕ. The superstring for $\mathcal{S}_{\mathcal{H}}$ is defined by the concatenation $s_\phi^1 s_\phi^2 \cdots s_\phi^n$. Let $\{x_i^p\}_{i=1}^{z}$ be the set of variables that is associated to the wheel $\mathcal{W}_p$. Let us assume that $\phi(x_2^p) = 1$ holds. Then, we use the 1-alignments of all sets associated to equations with exactly two variables in the wheel $\mathcal{W}_p$. The string s_ϕ^p begins with the following substring.

$$L_p C_p^l \xrightarrow{1} C_p^l x_2^{r1} x_1^{m0} C_p^r \xrightarrow{2} x_1^{m0} C_p^r C_p^l x_2^{r1}$$

Let us assume that $x_2 \oplus x_j = 0$ is a matching equation in $I_{\mathcal{H}}$. Then, we align $x_1^{m0} C_p^r C_p^l x_2^{r1}$ with the 1-alignment of the strings associated with $x_2 \oplus x_j = 0$, $x_2 \oplus x_3 = 0$ and so on.

$$x_1^{m0} C_p^r C_p^l x_2^{r1} \xrightarrow{1} x_2^{r1} x_2^{l1} x_j^{r0} x_j^{l0} \xrightarrow{2} x_j^{r0} x_j^{l0} x_2^{r1} x_2^{l1} \xrightarrow{1} x_2^{l1} x_3^{r1} x_2^{m0} x_3^{m0} \xrightarrow{2} x_2^{m0} x_3^{m0} \ldots$$

Let us consider the case that $x_{i-1} \oplus x_i = 0$ is an equation with two variables and x_i also appears in an equations with three variables of the form $x_i \oplus y \oplus z \oplus = 0 = b_j$. We assume that we have $x_i = 1$, $z = 1$ and $y = 0$. Then, we make use of the following alignments.

$$x_{i-1}^{l1} x_i^{r1\beta} x_{i-1}^{l1} x_i^{r1\alpha} \xrightarrow{2} x_{i-1}^{l1} x_i^{r1\alpha} x_{i-1}^{m0} x_i^{m0} \xrightarrow{2} x_{i-1}^{m0} x_i^{m0} x_{i-1}^{l1} x_i^{r1\beta} \xrightarrow{1} x_i^{r1\beta} x_i^{l1} z^{r1} z^{l1} \xrightarrow{2}$$

$$z^{r1} z^{l1} C_j x_i^{m0} \xrightarrow{2} C_j x_i^{m0} x_i^{r1\beta} x_i^{l1} \xrightarrow{1} x_i^{l1} x_{i+1}^{r1} x_i^{m0} x_{i+1}^{m0} \xrightarrow{2} x_i^{m0} x_{i+1}^{m0} x_i^{l1} x_{i+1}^{r1} \ldots$$

The string s_ϕ^p ends with the substring $x_1^{l1} C_p^r C_p^l x_2^{m0} \xrightarrow{2} C_p^l x_2^{m0} x_1^{l1} C_p^r \xrightarrow{1} C_p^r R_p$. If we have given $\phi(x_2^p) = 0$, then, we use the 0-alignments of the sets corresponding to equations with two variables.

We are going to describe which strings we use for equations with three variables. Let $x \oplus y \oplus z = 0$ be an equation with exactly three variables in $I_{\mathcal{H}}$. Let us first consider the two cycle equations $x_{i-1} \oplus x = 0$ and $x \oplus x_{i+1} = 0$. We note that the two associated $0/1$-alignments can be overlapped by one letter if both use the 0-alignment. Let us assume that the associated strings are using 1-alignments. Then, we want to overlap the constellation for $x = 1$ of the strings in $S^\alpha(b_j^3)$ with each 1-alignment by one letter. That means if we have an assignment to x, y and z that satisfies $x \oplus y \oplus z = 0$, we can use the corresponding constellations (for which the associated variable takes the value 1) of the strings in the sets $S^\alpha(b_j^3)$ and $S^\beta(b_j^3)$ such that the two resulting fragments can be overlapped by one letter from each side with strings corresponding to equations with two variables.

On the other hand, if the assignment is not satisfying, then, we can use at least one constellation to overlap with 1-alignments of strings. But, all in all, we lose an overlap of one letter compared to the number of overlapped letters in case of a satisfying assignment.

In summary, for each equation with two variables, we use a $0/1$-alignment, which yields 59ν strings with length 6. (Except in the case when the variable appears in an equation with

three variables. We will add 2 for each equation with three variables to the length of the superstring.) For each equation with three variables, we use an appropriate constellation and obtain $2 \cdot 2\nu$ strings with length 8. For cycle border equations, we use $0/1$-alignments as well. It yields two strings of length 6 and two strings of length 2.

All these fragments can be overlapped each by one letter except in the case when the equation with three variables is not satisfied. In this case, we lose an overlap of one letter. Thus, we obtain a string with length at most $(6-1) \cdot 60\nu + 2 \cdot (8-1) \cdot 2\nu + 2 \cdot 2\nu + 8n + \delta\nu$. This string achieves a compression at least

$$60\nu(2 \cdot 4) + 2\nu(6 \cdot 4 + 2 \cdot 4) + (2 \cdot 4 + 2 \cdot 2)n - (332\nu + 8n + \delta\nu) = 204\nu + 4n - \delta\nu$$

and the proof of the lemma follows. □

11.4 Defining an Assignment from a Superstring

In this section, we are going to prove the other direction of our reduction. Given a superstring for $\mathcal{S}_{\mathcal{H}}$, we are going to define an assignment to the variables of $I_{\mathcal{H}}$ and give the proof of the following lemma.

Lemma 7. *Let $\delta \in (0,1)$ be a constant, $I_{\mathcal{H}}$ an instance with n wheels, $60 \cdot \nu$ equations with two variables and $2 \cdot \nu$ equations with exactly three variables. If there is a superstring for $\mathcal{S}_{\mathcal{H}}$ with length $344 \cdot \nu + 8 \cdot n + \delta \cdot \nu$ and compression $4n + 204 \cdot \nu - \delta \cdot \nu$, then, it is possible to construct efficiently an assignment that leaves at most $\delta \cdot \nu$ equations in $I_{\mathcal{H}}$ unsatisfied.*

Proof. Let s be a superstring for $\mathcal{S}_{\mathcal{H}}$ with length $344 \cdot \nu + 8 \cdot n + \delta \cdot \nu$. First, we are going to apply local transformation in order to define our assignment to the variables of $I_{\mathcal{H}}$. Let us consider the two strings $s_i = x_i^{r1} x_i^{l1} x_j^{r0} x_j^{l0}$ and $s_j = x_j^{r0} x_j^{l0} x_i^{r1} x_i^{l1}$ that are associated to a matching equation. Note that s_i as well as s_j can have an overlap of at most one letter with strings in $\mathcal{S}_{\mathcal{H}} \backslash \{s_i, s_j\}$. Since a $0/1$-alignment of s_i and s_j yields an overlap of two letters, we can rearrange the strings in s such that s' is a superstring for $\mathcal{S}_{\mathcal{H}}$ with at most the same length as s and s' contains a $0/1$-alignment of s_i and s_j as substring. Due to this transformation, we may assume that the underlying superstring contains only $0/1$-alignments of strings corresponding to equations with two variables. In addition, the $0/1$-alignment of the strings corresponding to $x_i \oplus x_{i-1} = 0$ defines the value that we assign to the variable x_i.

Let us analyze gadgets for matching equations: Suppose we are given the five equations $x_{i-1} \oplus x_i = 0$, $x_i \oplus x_{i+1} = 0$, $x_{j-1} \oplus x_j = 0$, $x_i \oplus x_j = 0$ and $x_j \oplus x_{j+1} = 0$. Let S be the set of strings corresponding to the equations above. Since the length of the strings is exactly 4, we can convert S into an instance (V_S, d_S) of the $(1,4)$-ATSP. We notice that we obtain a similar structure as in the reduction to the $(1,2)$-ATSP (see Figure 21). Analogously to Lemma 4, we define transformation such that if $(3-u)$ of the $0/1$-alignments can be overlapped by one letter, then, the associated assignment leaves at most u equations unsatisfied out of $x_i \oplus x_{i+1} = 0$, $x_i \oplus x_j = 0$ and $x_j \oplus x_{j+1} = 0$.

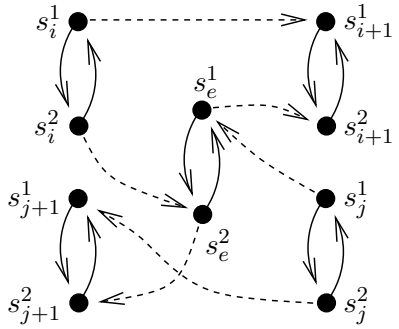


Figure 21: The construction simulating $x_i \oplus x_{i+1} = 0$, $x_i \oplus x_j = 0$ and $x_j \oplus x_{j+1} = 0$. Dotted and straight arrows represent arcs a with $d(a) = 3$ and $d(a) = 2$, respectively. For all other arcs a, we have $d(a) = 4$.

Let us now consider gadgets corresponding to equations with three variables: Due to rearrangements we may assume that the superstring contains one of the constellations that are associated to the equation $x \oplus y \oplus z = 0 = b_j^3$. Therefore, we obtain two strings s_j^1 and s_j^2, which can be overlapped by one letter with $0/1$-alignments of cycle equations. For $b \in \{0, 1\}$, we denote by s_{ix}^b and $s_{(i+1)x}^b$ the b-alignment of the strings corresponding to the equation $x_{i-1} \oplus x = 0$ and $x \oplus x_{i+1} = 0$, respectively. The strings s_{ky}^b, $s_{(k+1)y}^b$, s_{tz}^b and $s_{(t+1)z}^b$ have a similar meaning.

Let us first assume that for $\gamma \in \{x, y, z\}$, the strings $s_{i\gamma}^{b_i}$ and $s_{(i+1)\gamma}^{b_{i+1}}$ are using the same $0/1$-alignment, that is, we have $b_{i+1} = b_i$. We are going to show that we can use the constellations to overlap with the $0/1$-alignments each by one letter if the equation $x \oplus y \oplus z = 0$ is satisfied: Let us assume that $x = 0$, $y = 0$ and $z = 0$ holds. Then, we get $s_{ix}^0 \xrightarrow{1} s_{cj}^{0xl} \xrightarrow{1} s_{cj}^{0xr} \xrightarrow{1} s_{(i+1)x}^0$, $s_{ky}^0 \xrightarrow{1} s_{(k+1)y}^0$ and $s_{tz}^0 \xrightarrow{1} s_{(t+1)z}^0$. We count 5 letters that can be overlapped in this case.
For $x = 0$, $y = 0$ and $z = 1$, we have $s_{tz}^0 \xrightarrow{1} s_{cj}^{1z} \xrightarrow{1} s_{(t+1)z}^0$, $s_{ix}^0 \xrightarrow{1} s_{(i+1)x}^0$ and $s_{ky}^0 \xrightarrow{1} s_{(k+1)y}^0$. We obtain only 4 letters that we overlapped in this case. Observe that by using $s_{cj}^{0xl} \xrightarrow{1} s_{cj}^{0xr}$, it is not possible to achieve more letters that we can overlap. The other cases $x + y + z = 1$ can be discussed similarly.
Assuming $(x + y + z = 2)$, it is possible to overlap s_j^1 and s_j^2 with the $0/1$-alignments of the variables that take the value 1. Note that s_{ix}^1 can be either the string $x_{i-1}^{l1}x^{r1\beta}x_{i-1}^{l1}x^{r1\alpha} \xrightarrow{2} x_{i-1}^{l1}x^{r1\alpha}x_{i-1}^{m0}x^{m0} \xrightarrow{2} x_{i-1}^{m0}x^{m0}x_{i-1}^{l1}x^{r1\beta}$ or $x_{i-1}^{l1}x^{r1\alpha}x_{i-1}^{m0}x^{m0} \xrightarrow{2} x_{i-1}^{m0}x^{m0}x_{i-1}^{l1}x^{r1\beta} \xrightarrow{2} x_{i-1}^{l1}x^{r1\beta}x_{i-1}^{l1}x^{r1\alpha}$. In each case, we obtain 5 letters that we can overlap.
Suppose we have $x + y + z = 3$. Then, it is possible to align s_j^1 and s_j^2 from both sides with $0/1$-alignments. Note that it is not possible to overlap the remaining two $0/1$-alignments. Thus, we obtain in total 4 letters that we can overlap.
Summarizing, in the cases, when $x \oplus z \oplus y = 0$ is left unsatisfied, it yields one less letter, that is overlapped.

Finally, we observe that if the two strings $s_{i\gamma}^{b_i}$ and $s_{(i+1)\gamma}^{b_{i+1}}$ corresponding to a variable γ have not the same $0/1$-alignment, that is $b_i \neq b_{i+1}$, only one of the strings $s_{i\gamma}^{b_i}$ and $s_{(i+1)\gamma}^{b_{i+1}}$ can be aligned with either s_j^1 or s_j^2. In addition, it is not possible to overlap $s_{i\gamma}^{b_i}$ with $s_{(i+1)\gamma}^{b_{i+1}}$ by one letter. This corresponds to the fact that the cycle equation is not satisfied. The gadgets for cycle border equations can be analyzed very similarly. In conclusion, given a superstring for $\mathcal{S}_\mathcal{H}$ with length $344 \cdot \nu + 8 \cdot n + \delta \cdot \nu$ and compression $4n + 204 \cdot \nu - \delta \cdot \nu$, it is possible to extract efficiently an assignment that leaves at most $\delta\nu$ equations in $I_\mathcal{H}$ unsatisfied. □

We are ready to give the proof of Theorem 1.(v) and 1.(vi).

11.5 Proof of Theorem 1.(v) and 1.(vi).

For a fixed $\delta > 0$, we are given an instance $I_\mathcal{H}$ of the Hybrid problem with n wheels, 2ν equations with three variables and 60ν equations with two variables such that $8n/\nu \leq \delta$ holds. Then, we construct the associated instance $\mathcal{S}_\mathcal{H}$. Due to Lemmata 6, 7 and Theorem 3, we know that for all $\epsilon > 0$, it is **NP**-hard to tell whether there is a superstring with length at most $344 \cdot \nu + 8 \cdot n + \epsilon \cdot \nu \leq 344 \cdot \nu + (\delta + \epsilon)\nu$ or all superstrings have length at least $344\nu + (1 - \epsilon)\nu + 8 \cdot n \geq 345 \cdot \nu - \epsilon \cdot \nu$. The ratio between these two cases can get arbitrarily close to $345/344$ by appropriate choices for ϵ and δ.

Similarly, we conclude that for all $\epsilon > 0$, it is **NP**-hard to tell whether there is a superstring with compression at least $204 \cdot \nu + 4 \cdot n - \epsilon \cdot \nu \geq 204 \cdot \nu - \epsilon \cdot \nu$ or all superstrings have compression at most $204\nu - (1 - \epsilon)\nu + 4 \cdot n \leq 203 \cdot \nu + (\epsilon + \delta) \cdot \nu$. □

Acknowledgements

We thank Michael Lampis for a number of interesting discussions.

References

[ALM+98] S. Arora, C. Lund, R. Motwani, M. Sudan and M. Szegedy, *Proof Verification and the Hardness of Approximation Problems*, J. ACM 45 (1998), pp. 501–555.

[AGM+10] A. Asadpour, M. Goemans, A. Madry, S. Gharan and A. Saberi, *An $O(\log n/\log\log n)$-Approximation Algorithm for the Asymmetric Traveling Salesman Problem*, Proc. 21st SODA (2010), pp. 379–389.

[BGS02] A. Barvinok, E. Gimadi and A. Serdyukov, *The Maximum Traveling Salesman Problem*, in: *The Traveling Salesman Problem and Its Variations*, pp. 585–607, G. Gutin and A. Punnen, eds., Kluwer, 2002.

[BK99] P. Berman, M. Karpinski, *On Some Tighter Inapproximability Results*, Proc. 26th ICALP (1999), LNCS 1644, 1999, pp. 200–209.

[BK01] P. Berman and M. Karpinski, *Efficient Amplifiers and Bounded Degree Optimization*, ECCC TR01-053, 2001.

[BK03] P. Berman and M. Karpinski, *Improved Approximation Lower Bounds on Small Occurrence Optimization*, ECCC TR03-008, 2003.

[BK06] P. Berman and M. Karpinski, *8/7-approximation algorithm for* $(1,2)$*-TSP*, 17th SODA (2006), pp. 641–648.

[B04] M. Bläser, *A 3/4-Approximation Algorithm for Maximum ATSP with Weights Zero and One*, Proc. 8th APPROX-RANDOM (2004), pp. 61–71.

[BS00] H.-J. Böckenhauer and S. Seibert, *Improved Lower Bounds on the Approximability of the Traveling Salesman Problem*, Theor. Inform. Appl. 34, pp. 213–255, 2000.

[C76] N. Christofides, *Worst-Case Analysis of a New Heuristic for the Traveling Salesman Problem*, Technical Report CS-93-13, Carnegie Mellon University, Pittsburgh, 1976.

[E03] L. Engebretsen, *An Explicit Lower Bound for TSP with Distances One and Two*, Algorithmica 35, pp. 301–318, 2003.

[EK06] L. Engebretsen and M. Karpinski, *TSP with Bounded Metrics*, J. Comput. Syst. Sci. 72 (2006), pp. 509–546.

[EPZ12] K. Elbassioni, K. Paluch and A. v. Zuylen, *Simpler Approximation of the Maximum Asymmetric Traveling Salesman Problem*, Proc. 29th STACS 2012, pp. 501–506.

[H01] J. Håstad, *Some Optimal Inapproximability Results*, J. ACM 48 (2001), pp. 798–859.

[KLS+05] H. Kaplan, M. Lewenstein, N. Shafrir and M. Sviridenko, *Approximation Algorithms for Asymmetric TSP by Decomposing Directed Regular Multigraphs*, J. ACM 52 (2005), pp. 602–626.

[K01] M. Karpinski, *Approximating Bounded Degree Instances of NP-hard Problems*, Proc. 13th Symp. on Fundamentals of Comp. Theory, LNCS 2138, Springer, pp. 24–34, 2001.

[KLS13] M. Karpinski, M. Lampis and R. Schmied, *New Inapproximability Bounds for TSP* CoRR arXiv:1303.6437, 2013.

[KS11] M. Karpinski and R. Schmied, *Improved Lower Bounds for the Shortest Superstring and Related Problems* CoRR arXiv:1111.5442, 2011; appeared also in Proc. 19th CATS (2013), CRPIT 141, pp. 27–36, 2013.

[KS12] M. Karpinski and R. Schmied, *On Approximation Lower Bounds for TSP with Bounded Metrics*, CoRR arXiv:1201.5821, 2012; appeared also in Proc. 19th CATS (2013), CRPIT 141, pp. 27–36, 2013.

[KS13] M. Karpinski and R. Schmied, *Approximation Hardness of Graphic TSP on Cubic Graphs*, CoRR arXiv:1304.6800, 2013.

[L12] M. Lampis, *Improved Inapproximability for TSP*, CoRR 1206.2497, 2012; also appeared in Proc. 15th. APPROX (2012), LNCS 7408, pp. 243–253, Springer.

[L90] M. Li, *Towards a DNA Sequencing Theory*, Proc. 31th FOCS (1990), pp. 125–134.

[M12] M. Mucha, *Lyndon Words and Short Superstrings*, CoRR arXiv:1205.6787, 2012; also appeared in Proc. 24th ACM-SIAM SODA (2013), pp. 958–972.

[PV06] C. Papadimitriou and S. Vempala, *On the Approximability of the Traveling Salesman Problem*, Combinatorica 26 (2006), pp. 101–120.

[PY93] C. Papadimitriou and M. Yannakakis, *The traveling Salesman Problem with Distances One and Two*, Math. Oper. Res. 18 (1993), pp. 1–11.

[S88] J. Storer, *Data Compression: Methods and Theory*, Computer Science Press, 1988.

[T00] L. Trevisan, *When Hamming Meets Euclid: The Approximability of Geometric TSP and Steiner Tree*, SIAM J. Comput. 30 (2000), pp. 475–485.

[V05] V. Vassilevska, *Explicit Inapproximability Bounds for the Shortest Superstring Problem*, Proc. 30th MFCS (2005), LNCS 3618, 2005, pp. 793–800.

[V92] S. Vishwanathan, *An Approximation Algorithm for the Asymmetric Travelling Salesman Problem with Distances One and Two*, Inf. Process. Lett. 44 (1992), pp. 297–302.