Finding Paths in Grids with Forbidden Transitions

M. M. Kanté[1], F. Z. Moataz[2,3⋆], B. Momège[2,3], and N. Nisse[3,2]

[1] Univ. Blaise Pascal, LIMOS, CNRS, Clermont-Ferrand, France
[2] Univ. Nice Sophia Antipolis, CNRS, I3S, UMR 7271, 06900 Sophia Antipolis, France
[3] INRIA, France

Abstract. A transition in a graph is a pair of adjacent edges. Given a graph $G = (V, E)$, a set of forbidden transitions $\mathcal{F} \subseteq E \times E$ and two vertices $s, t \in V$, we study the problem of finding a path from s to t which uses none of the forbidden transitions of $\mathcal{F}$. This means that it is forbidden for the path to consecutively use two edges forming a pair in $\mathcal{F}$. The study of this problem is motivated by routing in road networks in which forbidden transitions are associated to prohibited turns as well as routing in optical networks with asymmetric nodes, which are nodes where a signal on an ingress port can only reach a subset of egress ports. If the path is not required to be elementary, the problem can be solved in polynomial time. On the other side, if the path has to be elementary, the problem is known to be NP-complete in general graphs [Szeider 2003]. In this paper, we study the problem of finding an elementary path avoiding forbidden transitions in planar graphs. We prove that the problem is NP-complete in planar graphs and particularly in grids. In addition, we show that the problem can be solved in polynomial time in graphs with bounded treewidth. More precisely, we show that there is an algorithm which solves the problem in time $O((3\Delta(k+1))^{2k+2}n))$ in n-node graphs with treewidth at most k and maximum degree Δ.

1 Introduction

Driving in New-York is not easy. Not only because of the rush hours and the taxi drivers, but because of the no-left, no-right and no U-turn signs. Even in a "grid-like" city like New-York, prohibited turns might force a driver to cross several times the same intersection before eventually reaching their destination. In this paper, we give hints explaining why it is difficult to deal with forbidden-turn signs when driving in grid-like road networks.

Let $G = (V, E)$ be a graph. A transition in G is a pair of two distinct edges incident to a same vertex. Let $\mathcal{F} \subseteq E \times E$ be a set of forbidden transitions in G. We say that a path $P = (v_0, \ldots, v_q)$ is $\mathcal{F}$-valid if it contains none of the transitions of $\mathcal{F}$, i.e., $\{\{v_{i-1}, v_i\}, \{v_i, v_{i+1}\}\} \notin \mathcal{F}$ for $i \in \{1, \ldots, q-1\}$. Given G,

⋆ This author is supported by a grant from the "Conseil régional Provence Alpes-Côte d'Azur".

$\mathcal{F}$ and two vertices s and t, the Path Avoiding Forbidden Transitions (PAFT) problem consists in finding an $\mathcal{F}$-valid s-t-path.

The PAFT problem arises in many contexts. In optical networks for instance, nodes can be highly asymmetric with respect to their switching capabilities as pointed out in [2]. This means that an optical node might have some restrictions on its internal connectivity and that, consequently, signal on a certain ingress port can only reach a subset of the egress ports. As explained in [2, 4, 7], a node can be asymmetrically configured for many reasons such as the limitation on the number of physical ports of optical switch components and the low cost of asymmetric nodes compared to symmetric ones. The existence of asymmetric nodes adds some connectivity constraints in the network. This has motivated some studies to re-investigate, under the assumption of the existence of asymmetric nodes, some classical problems in optical network, such as routing [2, 4, 9] and protection with node-disjoint paths [7]. These studies do not highlight the computational complexity of the problems they consider. We point out here that the optical nodes configured asymmetrically can be modeled as vertices with forbidden transitions and the routing problem is an application of PAFT. The study of PAFT is also motivated by its relevance to vehicle routing. In road networks, it is possible that some roads are closed due to traffic jams, construction, etc. It is also frequent to encounter no-left, no-right and no U-turn signs at intersections. These prohibited roads and turns can be modeled by forbidden transitions.

When the PAFT problem is studied, a distinction has to be made according to whether the path to find is elementary (cannot repeat vertices) or non-elementary. Indeed, PAFT can be solved in polynomial time [6] for the non-elementary case while finding an elementary path avoiding forbidden transitions has been proved NP-complete in [12]. In this paper, we study the elementary version of the PAFT problem in planar graphs and more particularly in grids. Our interest for planar graphs is motivated by the fact that they are closely related to road networks. They are also an interesting special case to study while trying to capture the difficulty of the problem. Furthermore, to the best of our knowledge, this case has not been addressed before in the literature.

Related work PAFT is a special case of the problem of finding a path avoiding forbidden paths (PFP) introduced in [15]. Given a graph G, two vertices s and t, and a set $\mathcal{S}$ of forbidden paths, PFP aims at finding an s-t-path which contains no path of $\mathcal{S}$ as a subpath. When the forbidden paths are composed of exactly two edges, PFP is equivalent to PAFT. Many papers address the non-elementary version of PFP, proposing exact polynomial solutions [15, 8, 1]. The elementary counterpart has been recently studied in [10] where a mathematical formulation is given and two solution approaches are developed and tested. The computational complexity of the elementary PFP can be deduced from the complexity of PAFT which has been established in [12]. Szeider proved in [12] that finding an elementary path avoiding forbidden transitions is NP-complete and gave a complexity classification of the problem according to the types of the forbidden transitions. The NP-completeness proof in [12] does not extend to planar graphs.

PAFT is also a generalization of the problem of finding a properly colored path in an edge-colored graph (PEC). Given an edge-colored graph G^c and two vertices s and t, the PEC problem aims at finding an s-t-path such that any consecutive two edges have different colors. It is easy to see that PEC is equivalent to PAFT when the set of forbidden transitions consists of all pairs of adjacent edges that have the same color. The PEC problem is proved to be NP-complete in directed graphs [5] which directly implies that the PAFT problem is NP-complete in directed graphs[4].

Contribution Our main contribution is the proof that the PAFT problem is NP-complete in grids. We also prove that the problem can be solved in time $O((3\Delta(k+1))^{2k+2}n))$ in n-node graphs with treewidth at most k and maximum degree Δ. In other words, we prove that the PAFT problem is FPT in $k + \Delta$. Our NP-completeness result strengthens the one of Szeider [12] established in 2003 and extends to the problem of PFP.

The paper is organized as follows. The problem of PAFT is formally stated in Section 2. In Section 3, the problem is proven NP-complete in grids. A polynomial time algorithm for graphs with bounded treewidth is presented in Section 4. Finally, some directions for future work are presented in Section 5

2 Problem statement

Let $G = (V, E)$ be a graph. Given a subgraph H of G, a *transition* in H is a (not ordered) set of two distinct edges of H incident to a same vertex. Namely, $\{e, f\}$ is a transition if $e, f \in E(H)$, $e \neq f$ and $e \cap f \neq \emptyset$. Let $\mathcal{T}$ denote the set of all transitions in G. Let $\mathcal{F} \subseteq \mathcal{T}$ be a set of *forbidden transitions*. A transition in $\mathcal{A} = \mathcal{T} \setminus \mathcal{F}$ is said *allowed*. A *path* is any sequence $(v_0, v_1, \cdots, v_r)$ of vertices such that $v_i \neq v_j$ for any $0 \leq i < j \leq r$ and $e_i = \{v_i, v_{i+1}\} \in E$ for any $0 \leq i < r$. Given two vertices s and t in G, a path $P = (v_0, v_1, \cdots, v_r)$ is called an *s-t-path* if $v_0 = s$ and $v_r = t$. Finally, a path $P = (v_0, v_1, \cdots, v_r)$ is *$\mathcal{F}$-valid* if any transition in P is allowed, i.e., $\{e_i, e_{i+1}\} \notin \mathcal{F}$ for any $0 \leq i < r$.

Problem 1 (Problem of Finding a Path Avoiding Forbidden Transitions, PAFT). Given a graph $G = (V, E)$, a set $\mathcal{F}$ of forbidden transitions and two vertices $s, t \in V$. Is there an $\mathcal{F}$-valid s-t-path in G?

3 NP-completeness in grids

We start by proving that the PAFT problem is NP-complete in grids. For this purpose, we first prove that it is NP-complete in planar graphs with maximum

[4] Note that, in [5], the authors state that their result can be extended to planar graphs. However, there is a mistake in the proof of the corresponding Corollary 7: to make their graph planar, vertices are added when edges intersect. Unfortunately, this transformation does not preserve the fact that the path is elementary.

degree at most 8 by a reduction from 3-SAT. Then, we propose simple transformations to reduce the degree of the vertices and prove that the PAFT problem is NP-complete in planar graphs with degree at most 4. Finally, we prove it is NP-complete in grids.

Lemma 1. *The PAFT problem is NP-complete in planar graphs with maximum degree* 8*.*

Proof. The problem is clearly in NP. We prove the hardness using a reduction from the 3-SAT problem. Let Φ be an instance of 3-SAT, i.e., Φ is a boolean formula with variables $\{v_1, \cdots, v_n\}$ and clauses $\{C_1, \cdots, C_m\}$. We build a grid-like planar graph G where rows correspond to clauses and columns correspond to variables. In what follows, the colors are only used to make the presentation easier. Moreover, we consider undirected graphs but, since the forbidden transitions can simulate orientations, the figures are depicted with directed arcs for ease of presentation. Please note also that we use a multigraph in the reduction for the sake of simplicity. This multigraph can easily be transformed into a simple graph without changing the maximum degree.

Gadget G_{ij}. For any $i \leq n$ and $j \leq m$, we define the gadget G_{ij} depicted in Figure 1 and that consists of 4 edge-disjoint paths from s_{ij} to t_{ij}: two "blue" paths $\mathcal{B}T_{ij}$ and $\mathcal{B}F_{ij}$, and two "red" paths $\mathcal{R}T_{ij}$ and $\mathcal{R}F_{ij}$ defined as follows.

- $\mathcal{R}T_{ij} = (s_{ij}, \alpha_{ij}, true_{ij}, x_{ij}, true'_{ij}, y_{ij}, z_{ij}, t_{ij})$;
- $\mathcal{B}T_{ij} = (s_{ij}, \beta_{ij}, true_{ij}, x_{ij}, true'_{ij}, y_{ij}, z_{ij}, t_{ij})$;
- $\mathcal{R}F_{ij} = (s_{ij}, x_{ij}, y_{ij}, \gamma_{ij}, false_{ij}, z_{ij}, false'_{ij}, t_{ij})$;
- $\mathcal{B}F_{ij} = (s_{ij}, x_{ij}, y_{ij}, \delta_{ij}, false_{ij}, z_{ij}, false'_{ij}, t_{ij})$.

The forbidden transitions $\mathcal{F}_{ij}$ of the gadget G_{ij} are defined in such a way that the only way to go from s_{ij} to t_{ij} is by following one of the paths in $\{\mathcal{B}T_{ij}, \mathcal{B}F_{ij}, \mathcal{R}T_{ij}, \mathcal{R}F_{ij}\}$. It is forbidden to use any transition consisting of two edges from two different paths of the set $\{\mathcal{B}T_{ij}, \mathcal{B}F_{ij}, \mathcal{R}T_{ij}, \mathcal{R}F_{ij}\}$.

Intuitively, assigning the variable v_i to $True$ will be equivalent to choosing one of the paths $\mathcal{B}T_{ij}$ or $\mathcal{R}T_{ij}$ (called *positive* paths) depicted with full lines in Fig. 1. Respectively, assigning v_i to $False$ will correspond to choosing one of the paths $\mathcal{B}F_{ij}$ or $\mathcal{R}F_{ij}$ (called *negative* paths) and depicted by dotted line in Fig. 1.

So far, it is *a priori* not possible to start from s_{ij} by one path and arrive in t_{ij} by another path. In particular, the color by which s_{ij} is left must be the same by which t_{ij} is reached. If Variable v_i appears in Clause C_j, we add one edge to G_{ij} as follows. If v_i appears positively in C_j, we add the *brown* edge $\{\alpha_{ij}, \beta_{ij}\}$ that creates a "bridge" between $\mathcal{B}T_{ij}$ and $\mathcal{R}T_{ij}$. Similarly, if v_i appears negatively in C_j, we add the *green* edge $\{\gamma_{ij}, \delta_{ij}\}$ that creates a "bridge" between $\mathcal{B}F_{ij}$ and $\mathcal{R}F_{ij}$. When the gadget G_{ij} contains a brown (resp. green) edge, all the transitions containing the brown (resp. green) edge are allowed; this makes it possible to switch between the positive (resp. negative) paths $\mathcal{B}T_{ij}$ and $\mathcal{R}T_{ij}$ (resp. $\mathcal{B}F_{ij}$ and $\mathcal{R}F_{ij}$) when going from s_{ij} to t_{ij}. Hence, if v_i appears in C_j, it will be possible to start from s_{ij} with one color and arrive to t_{ij} with

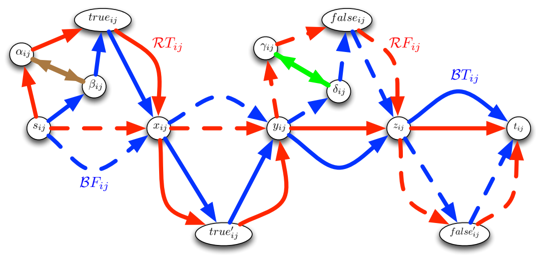


Fig. 1: Example of the Gadget-graph G_{ij} for Variable v_i, and $j \leq m$. Brown (resp. green) edge is added if v_i appears positively (resp., negatively) in C_j. If $v_i \notin C_j$, none of the green nor brown edge appear.

a different one. Note that, the type of path (positive or negative) cannot be modified between s_{ij} and t_{ij}.

We characterize the $\mathcal{F}_{ij}$-valid s_{ij}-t_{ij}-paths in G_{ij} with the following straightforward claims.

Claim 1 *The $\mathcal{F}_{ij}$-valid s_{ij}-t_{ij}-paths in G_{ij} are $\mathcal{R}T_{ij}, \mathcal{B}T_{ij}, \mathcal{R}F_{ij}, \mathcal{B}F_{ij}$ and*

- *if variable v_i appears positively in Clause C_j:*
  - *the path $\mathcal{RB}T_{ij}$ that starts with the first edge $\{s_{ij}, \alpha_{ij}\}$ of $\mathcal{R}T_{ij}$, then uses brown edge $\{\alpha_{ij}, \beta_{ij}\}$ and ends with all edges of $\mathcal{B}T_{ij}$ but the first one;*
  - *the path $\mathcal{BR}T_{ij}$ that starts with the first edge $\{s_{ij}, \beta_{ij}\}$ of $\mathcal{B}T_{ij}$, then uses brown edge $\{\alpha_{ij}, \beta_{ij}\}$ and ends with all edges of $\mathcal{R}T_{ij}$ but the first one;*
- *if variable v_i appears negatively in Clause C_j:*
  - *the path $\mathcal{RB}F_{ij}$ that starts with the subpath $(s_{ij}, x_{ij}, y_{ij}, \gamma_{ij})$ of $\mathcal{R}F_{ij}$, then uses green edge $\{\gamma_{ij}, \delta_{ij}\}$ and ends with the subpath of $\mathcal{B}F_{ij}$ that starts at δ_{ij} and ends at t_{ij};*
  - *the path $\mathcal{BR}F_{ij}$ that starts with the subpath $(s_{ij}, x_{ij}, y_{ij}, \delta_{ij})$ of $\mathcal{B}F_{ij}$, then uses green edge $\{\delta_{ij}, \gamma_{ij}\}$ and ends with the subpath of $\mathcal{R}F_{ij}$ that starts at γ_{ij} and ends at t_{ij};*

Claim 2 *Let P be a $\mathcal{F}_{ij}$-valid $s_{ij}t_{ij}$-paths in G_{ij}. Then, either*

- *P passes through $true_{ij}$ and $true'_{ij}$ and does not pass through $false_{ij}$ nor $false'_{ij}$, or*
- *P passes through $false_{ij}$ and $false'_{ij}$ and does not pass through $true_{ij}$ nor $true'_{ij}$.*

Claim 3 *Let P be a $\mathcal{F}_{ij}$-valid $s_{ij}t_{ij}$-paths in G_{ij}. Then the first and last edges of P have different colors if and only if P uses a green or a brown edge, i.e., if $P \in \{\mathcal{RB}T_{ij}, \mathcal{BR}T_{ij}, \mathcal{RB}F_{ij}, \mathcal{BR}F_{ij}\}$.*

Clause-graph G_j. For any $j \leq m$, the Clause-gadget G_j is built by combining the graphs G_{ij}, $i \leq n$, in a "line" (see Fig. 2). The subgraphs G_{ij} are combined from "left to right" (for $i = 1$ to n) if j is odd and from "right to left" (for $i = n$ to 1) otherwise. In more details, for any $j \leq m$, G_j is obtained from a copy of each gadget G_{ij}, $1 \leq i \leq n$, and two additional vertices s_j and t_j as follows:

- If j is odd, the subgraph G_j starts with a red edge $\{s_j, s_{1j}\}$ and then, for $1 < i \leq n$, the vertices s_{ij} and $t_{i-1,j}$ are identified. Finally, there is a blue edge from t_{nj} to vertex t_j.
- If j is even, the subgraph G_j starts with a blue edge $\{s_j, s_{nj}\}$ and then, for $1 < i \leq n$, the vertices t_{ij} and $s_{i-1,j}$ are identified. Finally, there is a red edge from t_{1j} to vertex t_j.

The forbidden transitions $\mathcal{F}_j$ include, besides all transitions in $\mathcal{F}_{ij}$, $i = 1, \ldots, n$, new transitions which are defined such that, when passing from a gadget G_{ij} to the next one, the same color must be used. This means that if we enter a vertex $t_{ij} = s_{i,j+1}$ by an edge with a given color, the same color must be used to leave this vertex. However, in such vertices, we can change the type (positive or negative) of path.

Note that if we enter a Clause-graph with a red (resp. blue) edge, we can only leave it with a blue (resp. red) edge. This means that a path must change its color inside the Clause-graph, and must hence use a brown or green edge in some gadget-graph. The use of a brown (resp. green) forces a variable that appears positively (resp. negatively) in the clause to be set to true (resp. false) and validates the Clause.

The key property of G_j relates to the structure of $\mathcal{F}_j$-valid paths from s_j to t_j, which we summarize in Claims 4 and 5.

Claim 4 *Any $\mathcal{F}_j$-valid path P from s_j to t_j in G_j consists of the concatenation of:*

Case j odd. *the red edge $\{s_j, s_{1j}\}$, then the concatenation of $\mathcal{F}_{ij}$-valid paths from s_{ij} to t_{ij} in G_{ij}, for $1 \leq i \leq n$ in this order (from $i = 1$ to n), and finally the blue edge $\{t_{nj}, t_j\}$;*

Case j even. *the blue edge $\{s_j, s_{nj}\}$, then the concatenation of $\mathcal{F}_{ij}$-valid paths from s_{ij} to t_{ij} in G_{ij}, for $1 \leq i \leq n$ in the reverse order (from $i = n$ to 1), and finally the red edge $\{t_{1j}, t_j\}$.*

By the previous claim, for any $\mathcal{F}_j$-valid path P from s_j to t_j, the colors of the first and last edges differ. Hence, by Claim 3 and the definition of the allowed transitions between two gadgets:

Claim 5 *Any $\mathcal{F}_j$-valid path P from s_j to t_j must use a green or a brown edge in a gadget G_{ij} for some $1 \leq i \leq n$.*

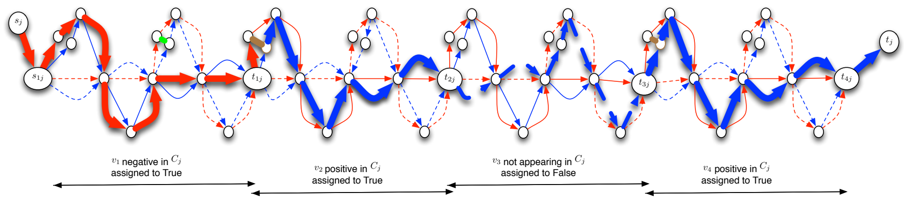


Fig. 2: Case j odd. Clause-graph G_j for a Clause $C_j = \bar{v}_1 \vee v_2 \vee v_4$ in a formula with 4 Variables. The bold path corresponds to an assignment of v_1, v_2 and v_4 to $True$, and of v_3 to $False$.

Main graph. To conclude, we have to be sure that the assignment of the variables is coherent between the clauses. For this purpose, let us combine the subgraphs G_j, $j \leq m$, as follows (see Fig 3). First, for any $1 \leq j < m$, let us identify t_j and s_{j+1}. Then, some vertices (depicted in grey in Fig 3) of G_{ij} are identified with vertices of $G_{i,j+1}$ in such a way that using a positive (resp., negative) path in G_{ij} forces the use of the same type of path in $G_{i,j+1}$. That is, the choice of the path used in G_{ij} is transferred to $G_{i,j+1}$ and therefore it corresponds to a truth assignment for Variable v_i.

Namely, for each $1 \leq j < m$ and for each $1 \leq i \leq n$, we identify the vertices $true_{i,j+1}$ and $false'_{ij}$ on the one hand, and the vertices $true'_{ij}$ and $false_{i,j+1}$ on the other hand to obtain the "grey" vertices. Finally, forbidden transitions $\mathcal{F}$ of G, include, besides all transitions in $\mathcal{F}_j$ for $j = 1, \ldots, m$, new transitions which are defined in order to forbid "crossing" a grey vertex, i.e., it is not possible to go from $G_{i,j}$ to $G_{i,j+1}$ via a grey vertex. The following claims present the key properties of an $\mathcal{F}$-valid path in G.

Claim 6 *Any $\mathcal{F}$-valid path P from s_1 to t_m in G consists of the concatenation of $\mathcal{F}_j$-valid paths from s_j to t_j in G_j from $j = 1$ to m.*

Claim 7 *Let P be an $\mathcal{F}$-valid $s_1 t_m$-path in G. Then, for any $1 \leq i \leq n$, either*

- *for any $1 \leq j \leq m$, the subpath of P between s_{ij} and t_{ij} passes through $true_{ij}$ and $true'_{ij}$ and does not pass through $false_{ij}$ nor $false'_{ij}$, or*
- *for any $1 \leq j \leq m$, the subpath of P between s_{ij} and t_{ij} passes through $false_{ij}$ and $false'_{ij}$ and does not pass through $true_{ij}$ nor $true'_{ij}$.*

Proof. By Claims 4 and 6, for any $1 \leq i \leq n$ and any $1 \leq j \leq m$, there is a subpath P_{ij} of P that goes from s_{ij} to t_{ij}. Moreover, the paths P_{ij} are pairwise vertex-disjoint.

For $1 \leq i \leq n$, by Claim 2, P_{i1} either passes through $true_{i1}$ and $true'_{i1}$, or through $false_{i1}$ and $false'_{i1}$. Let us assume that we are in the first case (the second case can be handled symmetrically). We prove by induction on $j \leq m$ that P_{ij} passes through $true_{ij}$ and $true'_{ij}$ and does not pass through $false_{ij}$ nor $false'_{ij}$.

Indeed, if P passes through $true_{ij} = false'_{i,j+1}$ and $true'_{ij} = false_{i,j+1}$, then $P_{i,j+1}$ cannot use $false_{i,j+1}$ nor $false'_{i,j+1}$ since P_{ij} and $P_{i,j+1}$ are vertex-disjoint. By Claim 2, $P_{i,j+1}$ passes through $true_{i,j+1}$ and $true'_{i,j+1}$.

□

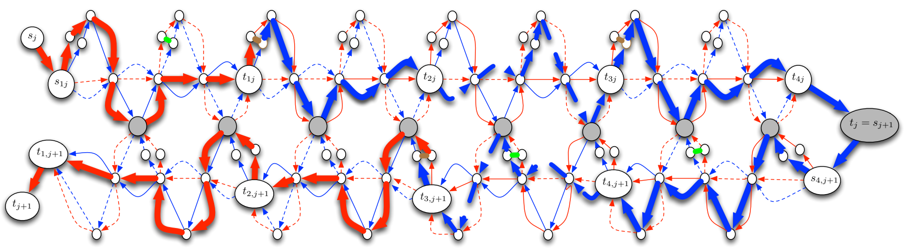


Fig. 3: Combining $C_j = \bar{v}_1 \vee v_2 \vee v_4$ and $C_{j+1} = v_2 \vee \bar{v}_3 \vee \bar{v}_4$ (Case j odd).

Note that $(G, \mathcal{F})$ can be constructed in polynomial-time. Moreover, G is clearly planar with maximum degree 8. Hence, the next claim allows to prove Lemma 1.

Claim 8 *Φ is satisfiable if and only if there is an $\mathcal{F}$-valid s_1-t_m-path in G.*

Proof. Let φ be a truth assignment which satisfies Φ. We can build an $\mathcal{F}$-valid s_1-t_m-path in G as follows. For each row $1 \leq j \leq m$, we build a path P_j from s_i to t_j by concatenating the paths P_{ij}, $1 \leq j \leq m$, which are built as follows. Among the variables that appear in C_j, let v_k be the variable with the smallest index, which satisfies the clause.

- For $1 \leq i < k$, if $\varphi(v_i) = true$, then $P_{ij} = \mathcal{R}T_{ij}$ if j is odd and $P_{ij} = \mathcal{B}T_{ij}$ if j is even, respectively. If $\varphi(v_i) = false$, then $P_{ij} = \mathcal{R}F_{ij}$ if j is odd, and $P_{ij} = \mathcal{B}F_{ij}$ if j is even.
- If $\varphi(v_k) = true$, then $P_{ij} = \mathcal{R}\mathcal{B}T_{ij}$ if j is odd, and $P_{ij} = \mathcal{B}\mathcal{R}T_{ij}$ if j is even. If $\varphi(v_k) = false$, then $P_{ij} = \mathcal{R}\mathcal{B}F_{ij}$ if j is odd, and $P_{ij} = \mathcal{B}\mathcal{R}F_{ij}$ if j is even.
- For $k < i \leq n$, if $\varphi(v_i) = true$, then $P_{ij} = \mathcal{B}T_{ij}$ if j is odd, and $P_ij = \mathcal{R}T_{ij}$ if j is even. If $\varphi(v_i) = false$, then $P_{ij} = \mathcal{B}F_{ij}$ if j is odd, and $P_{ij} = \mathcal{R}F_{ij}$ otherwise.

The path P obtained from the concatenation of paths P_j for $1 \leq j \leq m$ is an $\mathcal{F}$-valid path from s_1 to t_m.

Now let us suppose that there is an $\mathcal{F}$-valid path P from s_1 to t_m. According to Claim 7, for any $1 \leq i \leq n$, for any $1 \leq j \leq m$, P passes through $true_{ij}$ and $true'_{ij}$ or for any $1 \leq j \leq m$, P passes through $false_{ij}$ and $false'_{ij}$. Let us then consider the truth assignment φ of Φ such that for each $1 \leq i \leq n$:

- If P uses $true_{ij}$ and $true'_{ij}$ in all rows $1 \leq j \leq m$, then $\varphi(v_i) = true$.

– If P uses $false_{ij}$ and $false'_{ij}$ in all rows $1 \leq j \leq m$, then $\varphi(v_i) = false$.

Thanks to Claim 7, φ is a valid truth assignment. We need to prove that φ satisfies Φ. According to Claim 6, for each row $1 \leq j \leq m$, P contains an $\mathcal{F}_j$-valid path P_j from s_j to t_j. Each path P_j uses a green or a brown edge as stated by Claim 3. With respect to the possible ways to use a green or a brown edge which are stated in Claim 2, the use of a brown edge in P_j forces P_j (and hence P) to use, for a variable v_i that appears positively in C_j, the vertices $true_{ij}$ and $true'_{ij}$. Similarly, the use of a green edge in P_j forces P_j (and hence P) to use, for a variable v_i that appears negatively in C_j, the vertices $false_{ij}$ and $falce'_{ij}$. This means that for each clause C_j, for one of the variables that appear in C_j which we denote v_i, $\varphi(v_i) = true$ ($\varphi(v_i) = false$) if v_i appears positively (negatively) in C_j, respectively. Thus, the truth assignment φ satisfies Φ. □

Due to lack of space, we only give a sketch of the proof of the next lemma. The full proof can be found in the Appendix.

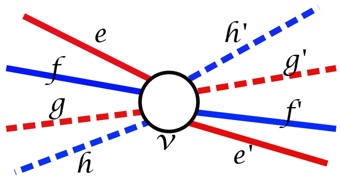
Lemma 2. *The PAFT problem is NP-complete in planar graphs with maximum degree* 4*.*

Proof (sketch). The graph G used in the reduction of the proof of Lemma 1 is planar and each vertex of G has either degree 8, degree 5 or degree at most 4. We transform G into a planar graph G' with maximum degree 4 and an associated set of forbidden transitions $\mathcal{F}'$ such that finding an $\mathcal{F}$-valid path in G is equivalent to finding an $\mathcal{F}'$-valid path in G':

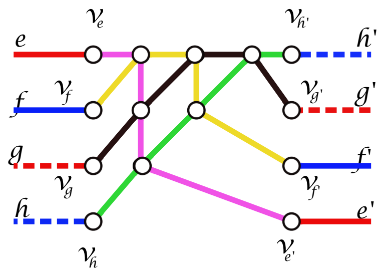
– We transform vertices of degree 5 to vertices of degree 3 as follows. For vertices s_{1j} (resp. t_{1j}) where j is odd we delete the 2 blue edges incident to s_{1j} (resp. t_{1j}). For vertices s_{nj} (resp. t_{1j}) where j is even, we delete the 2 red edges incident to s_{nj} (resp. t_{1j}).
– We replace each vertex v of degree 8 by a gadget g_v of maximum degree 4. Gadget g_v is designed such that it can be crossed at most once by a path of G' and only if the edges used to enter and leave g_v correspond to an allowed transition around v. Figure 5 gives an example of a vertex v in G and the corresponding gadget g_v in G'. □

Theorem 1. *The problem of finding a path avoiding forbidden transitions is NP-complete in grids.*

Proof. To prove the theorem we use the notion of planar grid embedding [13]. A planar grid embedding of a graph G is a mapping Q of G into a grid such that Q maps each vertex of G into a distinct vertex of the grid and each edge e of G into a path of the grid $Q(e)$ whose endpoints are mappings of vertices linked by e. For every pair $\{e, e'\}$ of edges of G, the corresponding paths $Q(e)$ and $Q(e')$ have no points in common, except, possibly, the endpoints. It has been proved in [14], that if $G = (V, E)$ is a planar graph such that $|V| = n$ and $\Delta \leq 4$, then a planar grid embedding of G in a grid of size at most $9n^2$ can be found in



A vertex of degree 8 and allowed transtions $A(v) = \{\{e, e'\}, \{f, f'\}, \{g, g'\}, \{h, h'\}\}$



Gadget g_v: the paths $P_{ee'}$, $P_{ff'}$,$P_{gg'}$, and $P_{hh'}$ are respectively the pink, yellow, black and green paths. Only transitions around vertices v_i and transitions of paths $P_{ee'}$, $P_{ff'}$,$P_{gg'}$, and $P_{hh'}$ are allowed

Fig. 4: Example of a vertex v and the corresponding gadget g_v

polynomial-time. Let us consider an instance of the problem of finding a path avoiding forbidden transitions in a planar graph $G = (V, E)$ of maximum degree at most 4 with a set of allowed transitions $\mathcal{A}$ ($\mathcal{A} = E \times E \setminus \mathcal{F}$). Let Q be a grid planar embedding of G into a grid K of size at most $O(|V|^2)$. Finding a PAFT between two nodes s and t in G with the set $\mathcal{A}$ is equivalent to finding a PAFT between the nodes $Q(s)$ and $Q(t)$ in K with the set of allowed transitions $\mathcal{A}'$ defined such that:

- For each $e \in E$, all the transitions in the path $Q(e)$ are allowed.
- For each $\{e, e'\} \in \mathcal{A}$, the pair of edges of $Q(e)$ and $Q(e')$, which share a vertex, is an allowed transition.

□

4 Parameterized complexity

On the positive side, by using dynamic programming on a tree-decomposition of the input graph, we prove that the problem is FPT when the parameter is the sum of the treewidth and the maximum degree.

A *tree-decomposition* of a graph [11] is a way to represent G by a family of subsets of its vertex-set organized in a tree-like manner and satisfying some connectivity property. The *treewidth* of G measures the proximity of G to a tree. More formally, a tree decomposition of $G = (V, E)$ is a pair $(T, \mathcal{X})$ where $\mathcal{X} = \{X_t | t \in V(T)\}$ is a family of subsets, called *bags*, of V, and T is a tree, such that: $\bigcup_{t \in V(T)} X_t = V$, for any edge $uv \in E$, there is a bag X_t (for some node $t \in V(T)$) containing both u and v, and for any vertex $v \in V$, the set $\{t \in V(T) | v \in X_t\}$ induces a subtree of T. The *width* of a tree-decomposition $(T, \mathcal{X})$ is $max_{t \in V(T)} |X_t| - 1$ and its *size* is order $|V(T)|$ of T. The treewidth of G, denoted by $tw(G)$, is the minimum width over all possible tree-decompositions of G.

Theorem 2 proves that when the treewidth of the graph is bounded, the PAFT can be solved in polynomial time. Complete proof of the theorem can be found in the Appendix.

Theorem 2. *The problem of finding a path avoiding forbidden transitions is FPT when parameterized by $k + \Delta$ where k is the treewidth and Δ is the maximum degree. In particular, there exists an algorithm that finds the shortest path avoiding forbidden transitions between two vertices in time $O((3\Delta(k+1))^{2k+2}n))$*

The Algorithm uses dynamic programming techniques and its key idea is similar to the one used to find a Hamiltonian cycle in graphs with bounded treewidth [3].

In more details, let $G = (V, E)$ be a graph with bounded treewidth k, $\mathcal{F}$ a set of forbidden transitions , and s and t two vertices of V. Let $(T, \mathcal{X})$ be a tree-decomposition of width k of G rooted in an arbitrary node. Let $G[A]$ be the subgraph of G induced by the set of vertices A. For each $u \in V(T)$, we denote by X_u,T_u and V_u the set of vertices of the bag corresponding to u, the subtree of T rooted at u, and the set of vertices of the bags corresponding to the nodes of T_u , respectively.

If there exists an $\mathcal{F}$-valid path P from s to t, then the intersection of this path with $G[V_u]$ for a node $u \in T$ consists of a set of paths avoiding forbidden transitions each having both endpoints in X_u. If $t \in V_u$, then one of the paths has only one endpoint in X_u. With respect to the parts of path P that are in $G[V_u]$, vertices in X_u can be partitioned into three subsets X_u^0, X_u^1, and X_u^2 which are the vertices of degree 0, 1 and 2 in $P \cap G[V_u]$, respectively. Furthermore, a matching M of X_u^1 decides which vertices are endpoints of the same subpath and a set of edges S defines which edges incident to X_u^1 are in P. For each node $u \in T$ and each subproblem $(X_u^0, X_u^1, X_u^2, M, S)$ where (X_u^0, X_u^1, X_u^2) is a partition of X_u, M is a matching of X_u^1 and S is a set of edges incident to the vertices of X_u^1, we need to check if there exists a set of paths avoiding forbidden transitions in V_u such that their endpoints are exactly X_u^1 according to the matching M, they contain the edges of S and the vertices of X_u^2 and they do not contain any vertex of X_u^0. For each node, we will need to solve at most $3^{k+1}(k+1)^{k+1}\Delta^{k+1}$ subproblems; there are at most 3^{k+1} possible partitions of the vertices of X_u into the 3 different sets, $(k+1)^{k+1}$ possible matchings for a set of $k+1$ elements and Δ possible edges for each element of X_u^1.

5 Conclusion

We have proven that the problem of finding a path avoiding forbidden transitions is NP-complete even in well-structured graphs as grids. We have also proved that PAFT can be solved in polynomial time when the treewidth is bounded. We believe that the PAFT is actually $W[1]$-hard when parameterized by the treewidth. Future work might focus on proving this conjecture and also on using structural properties of planar graphs to improve the running time of the algorithm for solving PAFT in planar graphs with bounded treewidth. Another interesting direction in the study of PAFT could be to consider the optimization problem where the objective is to find a path with minimum number of forbidden transitions and to investigate possible approximation solutions.

References

1. Ahmed, M., and Lubiw, A. Shortest paths avoiding forbidden subpaths. *Networks 61*, 4 (2013), 322–334.
2. Bernstein, G., Lee, Y., Gavler, A., and Martensson, J. Modeling WDM wavelength switching systems for use in GMPLS and automated path computation. *Optical Communications and Networking, IEEE 1*, 1 (June 2009), 187–195.
3. Bodlaender, H. Dynamic programming on graphs with bounded treewidth. In *Automata, Languages and Programming*, vol. 317 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1988, pp. 105–118.
4. Chen, Y., Hua, N., Wan, X., Zhang, H., and Zheng, X. Dynamic lightpath provisioning in optical wdm mesh networks with asymmetric nodes. *Photonic Network Com. 25*, 3 (2013), 166–177.
5. Gourvès, L., Lyra, A., Martinhon, C. A., and Monnot, J. Complexity of trails, paths and circuits in arc-colored digraphs. *Discrete Applied Mathematics 161*, 6 (2013), 819 – 828.
6. Gutiérrez, E., and Medaglia, A. L. Labeling algorithm for the shortest path problem with turn prohibitions with application to large-scale road networks. *Annals of Operations Research 157*, 1 (2008), 169–182.
7. Hashiguchi, T., Tajima, K., Takita, Y., and Naito, T. Node-disjoint paths search in wdm networks with asymmetric nodes. In *Optical Network Design and Modeling (ONDM), 2011 15th International Conference on* (Feb 2011), pp. 1–6.
8. Hsu, C.-C., Chen, D.-R., and Ding, H.-Y. An efficient algorithm for the shortest path problem with forbidden paths. In *Algorithms and Architectures for Parallel Processing*, A. Hua and S.-L. Chang, Eds., vol. 5574 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2009, pp. 638–650.
9. Jaumard, B., and Kien, D. Optimizing roadm configuration in wdm networks. In *Telecommunications Network Strategy and Planning Symposium (Networks), 2014 16th International* (Sept 2014), pp. 1–7.
10. Pugliese, L. D. P., and Guerriero, F. Shortest path problem with forbidden paths: The elementary version. *European Journal of Operational Research 227*, 2 (2013), 254 – 267.
11. Robertson, N., and Seymour, P. D. Graph minors. II. algorithmic aspects of tree-width. *J. Algorithms 7*, 3 (1986), 309–322.
12. Szeider, S. Finding paths in graphs avoiding forbidden transitions. *Discrete Applied Mathematics 126*, 2 - 3 (2003), 261 – 273.
13. Tamassia, R. On embedding a graph in the grid with the minimum number of bends. *SIAM J. Comput. 16*, 3 (June 1987), 421–444.
14. Valiant, L. G. Universality considerations in VLSI circuits. *IEEE Transactions on Computers 30*, 2 (1981), 135–140.
15. Villeneuve, D., and Desaulniers, G. The shortest path problem with forbidden paths. *European Journal of Operational Research 165*, 1 (2005), 97 – 107.

Appendix

We provide in this appendix the proofs of Lemma 2 and Theorem 2.

A Proof of Lemma 2

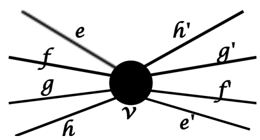
Let G be the graph obtained from the reduction in the proof of lemma 1 and let $\mathcal{E}$ be the planar embedding of G that is obtained by embedding the smaller gadgets as in Figures 1,2,3. the graph G has the following properties:

- G is planar.
- Each vertex of G has either degree 8 or degree ≤ 4. In fact, there are also vertices of degree 5 which we transform to vertices of degree 3 as follows. For vertices s_{1j} (resp. t_{1j}) where j is odd we delete the 2 blue edges incident to s_{1j} (resp. t_{1j}). For vertices s_{nj} (resp. t_{1j}) where j is even, we delete the 2 red edges incident to s_{nj} (resp. t_{1j}). This transformation does not affect the reduction or the proof.
- According to its forbidden transitions and to its disposition in the planar embedding $\mathcal{E}$, a vertex v of G of degree 8 has one of three following types:
  Type 1: The edges incident to v are $\omega(v) = \{e, e', f, f', g, g', h, h'\}$ and the allowed transitions around v are $A(v) = \{\{e, e'\}, \{f, f'\}, \{g, g'\}, \{h, h'\}\}$. The edges of v in the planar embedding $\mathcal{E}$. (v is a vertex of type x_{ij}, y_{ij} or z_{ij} in the graph G)
  Type 2: The edges incident to v are $\omega(v) = \{e, e', f, f', g, g', h, h'\}$ and the allowed transitions around v are $A(v) = \{\{e, e'\}, \{f, f'\}, \{g, g'\}, \{h, h'\}\}$. The edges of v in the planar embedding $\mathcal{E}$. (v is a vertex of type $true_{ij}$, $true'_{ij}$, $false_{ij}$, or $false'_{ij}$ in the graph G)
  Type 3: The edges incident to v are $\omega(v) = \{e, e', f, f', g, g', h, h'\}$ and the allowed transitions around v are $A(v) = \{\{e, e'\}, \{e, f'\}, \{f, f'\}, \{f, e'\}, \{g, g'\}, \{g, h'\}, \{h, h'\}, \{h, g'\}\}$. The edges of v in the planar embedding $\mathcal{E}$ are as depicted in Figure7a. (v is a vertex of type s_{ij} in the graph G)

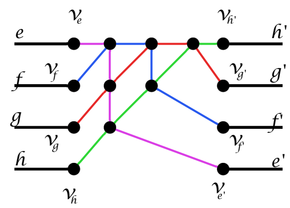
To prove the lemma, we are going to replace each vertex v of degree 8 in G with a gadget D_v. After replacing all vertices of degree 8, we will obtain a graph G' of maximum degree 4 and a new set of forbidden transitions $\mathcal{F}'$ such that finding an $\mathcal{F}-$valid path from s to t in G is equivalent to finding an $\mathcal{F}'$-valid path from s to t in G'. Let v be a vertex of degree 8 of G. D_v is constructed according to the type of v as follows:

Type 1 In this case, D_v is constructed as follows. For each $i \in \omega(v)$, a vertex v_i is created. For each $\{i, j\} \in A(v)$, vertices v_i and v_j are linked with a path P_{ij} of length four . The four paths P_{ij} are pairwise intersecting in distinct vertices as illustrated in Figure 5b. The allowed transitions in D_v are the transitions of the paths P_{ij}. Now to replace v with D_v in G, we do the following: each edge $i \in \omega(v)$ of G is linked to vertex v_i of D_v. The gadget D_v is planar, and edges $i \in \omega(v)$ are connected to it in the same "order" they were connected to v in the planar embedding of G as illustrated in Figure 5.

Note the gadget D_v cannot be crossed twice with the same path, otherwise the path is not simple. Moreover, D_v can be crossed if and only if the edges used to enter and leave form an allowed transition around v.



A vertex of degree 8 and allowed transtions $A(v) = \{\{e, e'\}, \{f, f'\}, \{g, g'\}, \{h, h'\}\}$ (edges are ordered as in the planar embedding $\mathcal{E}$ of G)



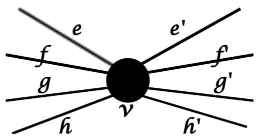
Gadget D_v: the paths $P_{ee'}$, $P_{ff'}$,$P_{gg'}$, and $P_{hh'}$ are respectively the pink, blue, red and green paths. Transitions around vertices v_i and transitions of paths $P_{ee'}$, $P_{ff'}$,$P_{gg'}$, and $P_{hh'}$ are allowed

Fig. 5: Type 1

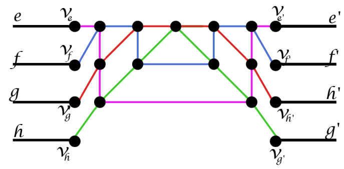
Type 2 In this case, D_v is constructed as follows. For each $i \in \omega(v)$, a vertex v_i is created. For each $\{i, j\} \in A(v)$, vertices v_i and v_j are linked with a path P_{ij} of length 7 . Each two of the four paths P_{ij} intersect in two different vertices as illustrated in Figure 6b. The allowed transitions in D_v are the transitions of the paths P_{ij}. Now to replace v with D_v in G, we do the following: each edge $i \in \omega(v)$ of G is linked to vertex v_i of D_v. The gadget D_v is planar, and edges $i \in \omega(v)$ are connected to it in the same "order" they were connected to v in the planar embedding $\mathcal{E}$ of G as illustrated in Figure 6. Note that the gadget D_v cannot be crossed twice with the same path, otherwise the path is not simple. Moreover, D_v can be crossed if and only if the edges used to enter and leave form an allowed transition around v.

Type 3 In this case, D_v is constructed as follows. For each $\{i, i'\} \in A(v)$, vertices v_i and v_j are linked with a path P_{ij} of length 7. Each two of the paths P_{ij} intersect twice in distinct vertices as illustrated in Figure 7b. Furthermore, we add two edges linking the paths $P_{ee'}$ and $P_{ff'}$, and $P_{gg'}$ and $P_{hh'}$, respectively. Now to replace v with D_v in G, we do the following: each edge $i \in \omega(v)$ of G is linked to vertex v_i of D_v. The gadget D_v is planar, and edges $i \in \omega(v)$ are connected to it in the same "order" they were connected to v in the planar embedding $\mathcal{E}$ of G as illustrated in Figure 7. Note that the gadget D_v cannot be crossed twice with the same path, otherwise the path is not simple. Moreover, D_v can be crossed if and only if the edges used to enter and leave form an allowed transition around v.

The graph G' is the one obtained from G after replacing vertices of degree 8 with the gadgets described above. The set of forbidden transitions $\mathcal{F}'$ consists

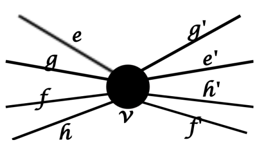


A vertex of degree 8 and allowed transtions $A(v) = \{\{e,e'\},\{e,f'\},\{f,f'\},\{f,e'\},\{g,g'\},\{g,h'\},\{h,h'\},\{h,g'\}\}$ (edges are ordered as in the planar embedding $\mathcal{E}$ of G)

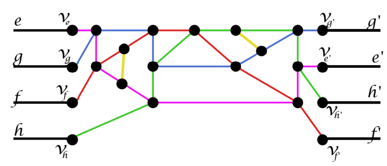


Gadget D_v: the paths $P_{ee'}$, $P_{ff'}$,$P_{gg'}$, and $P_{hh'}$ are respectively the pink, blue, red and green paths. Transitions around vertices v_i and transitions of paths $P_{ee'}$, $P_{ff'}$,$P_{gg'}$, and $P_{hh'}$ are allowed

Fig. 6: Type 2



A vertex of degree 8 and allowed transtions $A(v) = \{\{e,e'\},\{e,f'\},\{f,f'\},\{f,e'\},\{g,g'\},\{g,h'\},\{h,h'\},\{h,g'\}\}$ (edges are ordered as in the planar embedding $\mathcal{E}$ of G)



Gadget D_v: the paths $P_{ee'}$, $P_{ff'}$,$P_{gg'}$, and $P_{hh'}$ are respectively the pink, red, blue and green paths. transitions around vertices v_i, transitions of the paths $P_{ee'}$, $P_{ff'}$,$P_{gg'}$, and $P_{hh'}$, and transitions containing the yellow edge are allowed

Fig. 7: Type 3

of the transitions of the set $\mathcal{F}$ and the forbidden transitions of the gadgets D_v as described above. The maximum degree of G' is 4 and G' is planar.

Let us now suppose that there is an $\mathcal{F}$-valid path P from s to t in G. Let P' be the s-t-path of G' constructed as follows: P' uses all edges used by P. Furthermore, if P uses a degree 8 vertex of type 1 or 2 with a transition $\{e, e'\}$ then P' uses e, subpath $P_{ee'}$, and e'. If P uses a degree 8 vertex of type 3 with transition $\{e, e'\}$ (or $\{e, f'\}$), then P' uses e, e' and the subpath $P_{ee'}$ (e,f', and the subpath $P_{ef'}$ which is the concatenation of a subpath of path $P_{ee'}$, a yellow edge and a subpath of path $P_{ee'}$), respectively. The path P' is $\mathcal{F}'$-valid.

Now, let us suppose that there is an $\mathcal{F}'$-valid path P' from s to t in G'. If P' only uses edges from G, then it can be considered as an $\mathcal{F}$-valid path from s to t in G. If P' uses an edge that is not in G, then P' crosses one of the gadgets D_v. As we have specified above, the gadgets D_v can only be crossed in specified ways that ensure that the edges used to enter and leave the gadget form an allowed transition. We can then remove the edges of P' that do not belong to G to obtain an $\mathcal{F}$-valid path P in G. For any v of degree 8, the path P does not pass twice through v since gadget D_v in G' cannot be crossed twice by the same path.

B Proof of Theorem 2

To prove theorem 2, we first introduce the following definition and lemma.

Definition 1. *A rooted tree decomposition $((T, \mathcal{X}), r)$ of G is nice if for every node $u \in V(T)$:*

- *u has no children and $|X_u| = 1$ (u is called a leaf node), or*
- *u has one child v with $X_u \subset X_v$ and $|X_u| = |X_v| - 1$ (u is called a forget node),or*
- *u has one child v with $X_v \subset X_u$ and $|X_u| = |X_v|+1$ (u is called an introduce node), or*
- *u has two children v and w with $X_u = X_v = X_w$ (u is called a join node.).*

Lemma 3. *When given a tree decomposition of width w of G, in polynomial time we can construct a nice tree decomposition $(T, \mathcal{X})$ of G of width k, with $|V(T)| \in O(kn)$, where $n = |V(G)|$.*

We use the notion of nice tree decomposition and adapt the dynamic programming algorithm for finding a Hamiltonian cycle in a graph to prove Theorem 2.

Let $G = (V, E)$ be a graph with bounded treewidth k, $\mathcal{F} \subseteq E \times E$ a set of forbidden transitions (and $\mathcal{A} \subseteq E \times E$ the set of allowed transitions), and s and t two vertices of V. We would like to find the shortest path P from s to t avoiding the forbidden transitions $\mathcal{F}$.

Let $G_{e,f}$ such that e and f are edges incident to s and t, respectively, be the graph obtained from G, by deleting all edges incident to s and t except for e and f. Finding the shortest path avoiding forbidden transitions $\mathcal{F}$ from s to t in G is equivalent to finding the shortest path among all shortest paths avoiding forbidden transitions $\mathcal{F}$ from s to t in $G_{e,f}$, for each possible pair e, f. In the following we will present how to solve the CFT problem in $G_{e,f}$. To obtain the solution in G, we will need to repeat the algorithm at most Δ^2 times.

Let $(T, \mathcal{X})$ be a nice tree-decomposition of width k of $G_{e,f}$. We assume that s appears in one introduce bag and t in two bags, a leaf and its introduce parent. We root T at the node containing s. Let $G[A]$ be the subgraph of $G_{e,f}$ induced by the set of vertices A. . For each $u \in V(T)$ we denote by X_u,T_u and V_u the vertices of the bag corresponding to u, the subtree of T rooted at u, and the vertices of the bags corresponding to the nodes of T_u , respectively.

If there exists an $\mathcal{F}$-valid path P from s to t, then the intersection of this path with $G[V_u]$ for a node $u \in T$ is a set of paths (avoiding forbidden transitions) each having both endpoints in X_u. If $t \in V_u$, then one of the paths has only one endpoint in X_u.

With respect to the parts of path P that are in $G[V_u]$, vertices in X_u can be partitioned into three subsets: X_u^0, X_u^1, and X_u^2 which are the vertices of degree 0, 1 and 2 in $P \cap G[V_u]$, respectively. Furthermore, a matching M of X_u^1 decides which vertices are endpoints of the same subpath and a set of edges S defines which edges incident to X_u^1 are in P. For each node $u \in T$ and each subproblem $(X_u^0, X_u^1, X_u^2, M, S)$ where (X_u^0, X_u^1, X_u^2) is a partition of X_u, M is a matching of X_u^1 and S is a set of edges incident to the vertices of X_u^1, we need to see if there exists a set of paths avoiding forbidden transitions in V_u such that their endpoints are exactly X_u^1 according to the matching M, they contain the edges of S and the vertices of X_u^2 and they do not contain any vertex of X_u^0. For the case where $t \in V_u$, we will need to check the possible matchings of each subset of X_u^1 of size $|X_u^1| - 1$. For each node, we will need to solve at most $3^{k+1}(k+1)^{k+1}\Delta^{k+1}$ subproblems; there are at most 3^{k+1} possible partitions of the vertices of X_u into the 3 different sets, $(k+1)^{k+1}$ possible matchings for a set of $k+1$ elements and Δ possible edges for each element of X_u^1.

Let us see how to solve a problem $(X_u^0, X_u^1, X_u^2, M, S)$ at a node u supposing that all the problems at its descendants have been solved:

- If u is a leaf, then $X_u = \{a\}$. The only problem that has a solution is $(X_u^0 = \{a\}, X_u^1 = \emptyset, X_u^2 = \emptyset, M = \emptyset, S = \emptyset)$.
- If u is a forget node, let v be the child of u. We have $X_u = X_v \setminus a$. We can distinguish two cases:
  - If $a \neq t$, then the problem $(X_u^0, X_u^1, X_u^2, M, S)$ has a solution if and only if one of the problems $(X_u^0 \cup \{a\}, X_u^1, X_u^2, M, S)$ and $(X_u^0, X_u^1, X_u^2 \cup \{a\}, M, S)$ at node v has a solution.
  - If $a = t$, then the problem $(X_u^0, X_u^1, X_u^2, M, S)$ has a solution if and only if the problem $(X_u^0, X_u^1 \cup \{a\}, X_u^2, M, S)$ at v has a solution.
- If u is an introduce node, let v be the child of u. We have $X_u = X_v \cup a$ (all neighbors of a in V_u are in X_u). Note that $a \neq t$ since t appears in a forget node and its introduce parent. In this case we proceed as follows.
  - If $a \in X_u^0$, then solving $(X_u^0, X_u^1, X_u^2, M, S)$ at u is equivalent to solving $(X_u^0 \setminus \{a\}, X_u^1, X_u^2, M, S)$ at v.

- if $a \in X_u^1$, let ab be the edge incident to a in S. Since all neighbors of a in V_u are in X_u, then $b \in X_u \cap X_v$. Let us consider the following cases:
  - ∗ If $b = t$, then the only problem that has a solution at u is $(X_u \setminus \{a,t\}, \{a,t\}, \emptyset, \{(a,t)\}, \{(a,t)\})$. To solve it, we need to check at v the solution of the problem $(X_u \setminus \{a\}, \emptyset, \emptyset, \emptyset, \emptyset)$.
  - ∗ If $b \in X_u^1$ $(b \neq t)$, (the problem has a solution only if $(a,b) \in M$ and the edge incident to b in S is ab) then check at v the solution of the problem $(X_u^0 \cup \{b\}, X_u^1 \setminus \{a,b\}, X_u^2, M', S')$ where $M' = M \setminus (a,b)$ and $S' = S \setminus ab$.
  - ∗ If $b \in X_u^2$ $(b \neq t)$, then check at v the solution of the problem $(X_u^0, X_u^1 \setminus \{a\} \cup \{b\}, X_u^2 \setminus \{b\}, M', S')$ where $M' = M \setminus (a,h) \cup (b,h)$ and S' contains the set S minus the edge ab plus an edge incident to b that forms an allowed transition with edge ba (there are at most Δ such problems).
- If $a \in X_u^2$, then for every two neighbors b and c of a in X_u such that (ba, ac) is an allowed transition do the following.
  - ∗ If $b \in X_u^1$ and $c \in X_u^1$, then check the solution at v of the problem $(X_u^0 \cup \{b,c\}, X_u^1 \setminus \{b,c\}, X_u^2 \setminus \{a\}, M', S')$ where $M' = M \setminus (b,c)$ and remove ab and bc from S to obtain S'.
  - ∗ If $b \in X_u^2$ and $c \in X_u^2$, then check the solution at v of the problem $(X_u^0, X_u^1 \cup \{b,c\}, X_u^2 \setminus \{a,b,c\}, M', S')$ where $M' = M \cup \{bh, ch'\} \setminus hh'$ (bc should not be in the matching) and to obtain S', add to S two edges incident to b and c and forming allowed transitions with ab and ac, respectively (there are $\frac{k+1}{2}$ possible choices for hh' and Δ^2 possible choices for the two edges to add to S).
  - ∗ If $b \in X_u^1$ and $c \in X_u^2$, then check the solution at v of the problem $(X_u^0 \cup \{b\}, X_u^1 \setminus \{b\} \cup \{c\}, X_u^2 \setminus \{a,c\}, M', S')$ where $M' = M \setminus bh \cup ch$ and to obtain S' remove ab from S and add an edge incident to c that forms an allowed transition with ca. (There are Δ possibilities).

  Note that the number of pairs of neighbors of a to consider are of order of k^2.

– If u is a join node, let v and w be its children. For any two subproblems at v and w we check if the union of the two solutions is a solution for $(X_u^0, X_u^1, X_u^2, M, S)$ of node u. (At most $(3^{k+1}k + 1^(k+1)\Delta^{k+1})^2$ possibilities).

At the node containing s, we only need to solve subproblems where s and t are of degree 1 and all other vertices have either degree 2 or 0.

To find the shortest path, one has to choose, whenever having a choice between different solutions for a subproblem at a node, the solution with the minimum number of edges.