École Polytechnique

BACHELOR THESIS IN COMPUTER SCIENCE

Fast computations of connected toroidal Schnyder woods

Author:

Milana Komisarova, École Polytechnique

Advisor:

Luca Castelli Aleardi, LIX

Academic year 2024/2025

Abstract

Graphs embedded on surfaces are very important in mathematics and computer science, especially in applications such as graph encoding and geometric representations. For planar graphs, Schnyder woods have been explored for many years, and many properties have been found, while on toroidal graphs there are still many unanswered questions.
This thesis is focused on Schnyder woods on toroidal triangulations. We find an algorithm for reversing oriented tri-coloured cycles in a Schnyder wood, answering an open question from Gonçalves and Lévêque [7]. We prove that by performing the reversals, a Schnyder wood with one connected component per each color can always be obtained.
These results could be relevant for applications in graph drawing, compact encoding of toroidal graphs, and combinatorial optimization on higher-genus surfaces.

Contents

1 Introduction

1.1 Motivation

Graphs are an incredibly important and useful tool for solving many problems in different fields, such as mathematics and computer science. Planar graphs (graphs admitting a crossing-free embedding in the plane) have been explored for many years, leading to many theorems and structural characterizations. Planar and surface graphs are very useful are widely used nowadays in many different fields and application domains (e.g. computer graphics, 3D modelling) since they represent the combinatorial structure underlying 3D surface meshes (one of the most used discrete representations of 3D shapes).

Graphs embedded on higher genus surfaces (e.g. the torus) haven't been explored that much from the combinatorial point of view. Therefore this thesis is exploring some theorems and combinatorial structures (called Schnyder woods) about graphs on the torus, and if we're being more precise: in particular we focus on Schnyder woods for toroidal (simple) triangulations. There are many exciting properties that are already explored and this work continues to delve deeper, looking at non-crossing and half-crossing Schnyder woods on a triangulated toroidal graph with multiple mono- chromatic cycles. This work proves that performing the reversal of oriented cycles of directed edges, we can obtain a Schnyder wood with only one component in each color, which answers a question left open by Goncalves and Leveque [7].

This could be relevant for future applications of Schnyder woods (e.g., for graph encoding on the torus).

1.2 State of the Art

This work heavily relies on many previous works of people such as Luca Castelli Aleardi[3], Enno Brehm[2], Benjamin Leveque [7] and others, and already discovered properties of Schnyder woods, such as the Schnyder rule, many properties of monochromatic cycles, crossing conditions, connectedness conditions. It is also utilising the algorithm from the paper of Luca Castelli Aleardi, Éric Fusy, Jyh-Chwen Ko, and Razvan S. Puscasu called "Computation of toroidal Schnyder woods made simple and fast: from theory to practice."[3] to obtain the necessary starting condition for the algorithms that are explained. The main theorem this work aims to prove is: "Given a toroidal simple triangulation G and a Schnyder wood $S(G)$ (not necessarily connected), there exists a linear-time algorithm computing a Schnyder wood S' which is connected (in each color). Moreover, if $S(G)$ is half-crossing (resp. crossing) then S' is half-crossing (resp. crossing)." This theorem answers a conjecture left open in Goncalves and Leveque [7].

2 Preliminaries

2.1 Graphs embedded on surfaces

A graph G on a surface is G = (V,E), where V denotes the set of its vertices and E denotes the set of its edges. There are no edge-crossing allowed. The graph also has faces - a region bounded by a specific set of its edges. In this manuscript, we are focusing on triangulated graphs - that is, all its faces are triangles, which is equivalent to having 3 edges on its contour. We are working with graphs embedded on a torus, that is, toroidal graphs. [7]

2.2 Cutting a toroidal graph into a cylinder

In order to visualize the graph easier, it is really important to understand how a toroidal graph can be cut into a cylinder, and afterwards, a plane. Firstly, we must cut along the boundary of the torus going into the genus-1 hole, as shown in the picture. After cutting and stretching it out, we obtain a cylinder. We note that all vertices of the graph on the cut will be doubled - they will now exist both at the top and and the bottom ring of the cylinder, but they are still the same vertices. Afterwards, to get an even clearer representation of our graph, we cut the cylinder along its height and fold it out to obtain a rectangle. We note that here the same property holds - vertices on both sides of the rectangle are the same.

2.3 Schnyder woods (in the plane) and their basic properties (spanning trees)

We define a Schnyder wood an a planar triangulated graph as a partition of all its inner edges into three sets, we call them T_0, T_1, T_2, where each set has its own color - all edges of T_0 are red, all edges of T_1 are blue and all edges of T_2 are black. All of these edges are oriented and colored in such a way that each inner vertex has exactly one outgoing edge of each color, and each outer vertex will respectively have each at least one ingoing edge of only one color. We respectively name the outer vertices as the color of the incoming edges, that is, v_0, v_1, v_2. [7] A very important rule of Schnyder woods is that the Schnyder local condition, as seen in Figure 1, must be respected at every inner vertex of the graph, that is, red incoming edges must be only between black outgoing on the left and blue outgoing on the right (looking from the vertex), black incoming edges must be only between blue outgoing on the left and red outgoing on the right, and blue incoming edges must be only between red outgoing on the left and black outgoing on the right.

Figure 1: Schnyder local condition

This is based on [1].

An important property of the Schnyder woods is the spanning property, proved by Schnyder in 1990. It states that the three sets T_0, T_1, T_2 are spanning trees of the inner vertices of the graph, and each has their root at respectively v_0, v_1, v_2. It has also been proven that T_i does not contain any cycles and is connected, t.i., there are no disjoint components of the same color.

2.4 Toroidal Schnyder woods: definitions

We now define a toroidal Schnyder wood, that is, Schnyder woods on the torus. To get a toroidal Schnyder wood, we make our graph a 3-orientation (t.i., all of the edges are oriented and colored in such a way that each inner vertex has exactly one outgoing edge of each color) with Schnyder rule valid at every vertex.

2.5 Basic properties of cycles in toroidal Schnyder woods

Some properties of cycles in toroidal Schnyder woods include

- Toroidal Schnyder woods must contain a mono-chromatic cycle of each color. [7]
- All mono-chromatic cycles are non-contractibles, t.i. they cannot be reduced to a single point in the plane, due to Schnyder wood conditions.[7]
- All mono-chromatic cycles of the same color are homotopic and disjoint and oriented in one direction.[7]
- All mono-chromatic cycles of different colors are either disjoint and homotopic, t.i. parallel, or crossing. [7]
- All monochromatic cycles of color red and blue are oriented downwards, whilst black cycles are oriented upwards.[7]

2.6 Non-crossing, half-crossing and crossing Schnyder woods

Definition 1. *Toroidal Schnyder woods are*

- ***Non-crossing****, if all mono-chromatic cycles are pairwise disjoint and homotopic. [7] [1]*

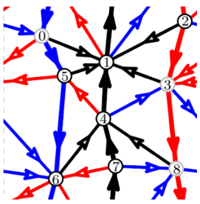

Figure 2: Non-crossing Schnyder wood example

- ***Half-crossing****, if at least one monochromatic cycle of exactly one color intersects at least one monochromatic cycles of both other colors, and all cycles of the other two colors are pairwise disjoint and homotopic. [7] [1]*

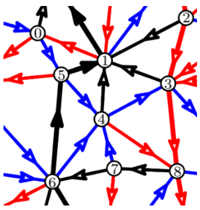

Figure 3: Half-crossing Schnyder wood example

- ***Crossing****, if every monochromatic cycle intersects at least one monochromatic cycle of each color.[7] [1]*

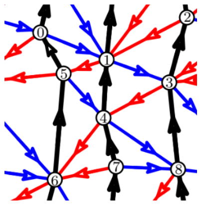

Figure 4: Crossing Schnyder wood example

2.7 Connected Schnyder woods

Definition 2. *A Schnyder wood is called* ***connected*** *if all monochromatic edges cannot be partitioned into two or more disjoint sets.[7]*

2.8 Reversal of multicolored directed cycles

It is possible to reverse a three-color oriented cycle in a Schnyder wood. It is already proved by Enno Brehm that reversing an oriented cycle of directed edges in a Schnyder wood you get another (different) valid Schnyder wood. To do that, one first must find a 3-oriented cycle, that is, a cycle that has a red path (one or multiple red edges that are a connected path in the same direction), a blue path and a black path. Afterwards, the colors and the directions of the outside and inside edges just get changed depending on whether the cycle is clockwise or counterclockwise, to still comply with the Schnyder rule. For example, if we have a clockwise oriented 3-coloured cycle, we change the color of the boundary as follows - the color the red path to black, the color of the black path to blue, and the color of the blue path to red, whilst also changing all directions to the opposite, t.i. creating a counterclockwise cycle. Inside the cycle, all colours also get respectively changed to satisfy the Schnyder rule conditions. Further explanations for this can be found in Enno's Brehm's work .[2]

3 Computation of toroidal Schnyder woods: short discussion of algorithms

There are two important algorithms from the paper of Luca Castelli Aleardi, Éric Fusy, Jyh-Chwen Ko, and Razvan S. Puscasu, that the later algorithms will be based on. Algorithm 1 and 3 from the paper "Computation of toroidal Schnyder woods made simple and fast: from theory to practice." [3] are describing how based on canonical ordering and using rivers to transform the cylindrical graph back to a toroidal embedding, it is possible to always obtain a select type of Schnyder wood.

That is, for non-crossing Schnyder woods, it is always possible to obtain a non-crossing Schnyder wood with either exactly one red or one blue monochromatic cycle, with multiple other color cycles, t.i. multiple blue and multiple black or multiple red and multiple black respectively, and that can be done in linear time. We are going to be using this exact type of Schnyder wood in the first algorithm described.

For half-crossing Schnyder woods, the algorithm shows how it is always possible to obtain a half-crossing Schnyder wood with also either exactly one red or one blue monochromatic cycle that is crossing all the other cycles, with multiple other color cycles, t.i. multiple blue and multiple black or multiple red and multiple black respectively that are parallel to each other, and that can be done in linear time. We are using this type of Schnyder woods in the second algorithm.

4 Our main contribution: making a toroidal Schnyder wood connected

In this section, we are exploring algorithms for toroidal Schnyder woods. I call a 'red' cycle a cycle that is monochromatic and red, a 'blue' cycle a cycle that is monochromatic and blue and a 'black' cycle a cycle that is monochromatic and black.

We also assume that our given Schnyder wood is cut into a rectangle and drawn on the plane to facilitate the understanding of the algorithm.

4.1 An algorithm for making a non-crossing Schnyder wood connected

After using the algorithms from the paper of Luca Castelli Aleardi et al. work [3], we can definitely obtain a non-crossing Schnyder wood with multiple red and multiple black cycles, but definitely only one blue cycle, that is, all of these cycles are homotopic and disjoint and there are no other monochromatic cycles. Here without the loss of generality we assume that red is the one that has multiple cycles, while blue has only one, it can also be the other way around, and in that case, the proof is symmetric, since both blue and red cycles are oriented downwards, whilst the black cycles are oriented upwards.

Now we proceed with the following algorithm to create a non-crossing Schnyder wood with only one monochromatic cycle of each color from the given Schnyder wood, which would make it connected, as if there are multiple monochromatic cycles, they are disjoint and homotopic, hence not connected. The algorithm's main idea is reversing three-coloured cycles to destroy the existing cycles, whilst not creating any new ones.

4.1.1 Step 1 - removing the red cycles

We first consider the rightmost black cycle that has a red cycle on the right of it with no other cycles inbetween them.

We now look at our red cycle and choose a vertex u. There exists an outgoing black edge on the left on the cycle due to Schnyder conditions, since the red cycle is oriented downwards. We then choose the vertex that this edge ends on, and proceed in a similar way, since every single vertex must have exactly one outgoing black edge due to Schnyder rules. We obtain a monochromatic black path made from one or many black edges. We can see that it must cross the chosen black cycle at one of the vertices. If it doesn't, that means that it ends in a black cycle that is before the chosen red cycle, which is contradictory to how we have chosen our cycles. Now we have a vertex z on the black cycle that this path has ended on. Now we look at this vertex z. There exists an outgoing blue edge on the right due to Schnyder conditions, since the black cycle is oriented upwards. We then choose the vertex that this edge ends on, and proceed in a similar way, since every single vertex must have exactly one outgoing blue edge due to Schnyder rules. We obtain a monochromatic blue path made from one or many blue edges. We can see that it must cross the chosen red cycle at one of the vertices. If it doesn't, that means that it ends in a blue cycle that is before the chosen red cycle, which is contradictory to how we have chosen our cycles. Therefore it must touch the red cycle at one of the vertices v. Since we have v and u on the same red cycle, there is a red path between them. Now that we have drawn our three paths from u to z, from z to v and from v to z, we have created a tri-color cycle oriented clockwise.

Now we proceed with the reversal of this cycle as described before. We see that the blue path becomes red, the red path becomes black and the black path becomes blue. We see that the previously existing red cycle has been destroyed, since a part of it is now black. We note that there have been no new red cycles created, since if there would have been, then the newly made red path would have been a part of it, however, it is impossible, because the vertex that it starts from is on the destroyed red cycle, so it must include it as well. That is impossible, because then it must end at the same vertex that it ended before, and that means have the same parts as before without the destroyed region. It is obviously impossible due to Schnyder rule condition - we have a black upwards oriented edge inbetween the red downwards oriented edges, which then means the incoming red edge to make a cycle must come from the right side of the vertex v, which is impossible, since the new part of the cycle is outgoing to the right side. So there are no new red cycles.

There also cannot be any new black cycles, because it would also have to contain the new segment, but in order for it to become a cycle, we must have edges incoming in and outgoing from this path from the same side - from the left or the right of the destroyed red cycle. We can quickly check that this is impossible due to Schnyder rules, so there are no new black cycles as well. Analogically this can be proven for the blue cycles. Therefore we have created no new cycles whilst destroying one of the red ones.

We can proceed the same way for all red cycles, destroying all of them one by one and applying the same algorithm, and there will not be any new cycles by the same reasoning, therefore we can destroy all but one red cycles to obtain a non-crossing Schnyder wood with one red cycle, one blue cycle and many black cycles.

4.1.2 Step 2 - removing the black cycles

Afterwards, we proceed analogically with destroying the black cycles. We can find a red path from one black cycle to another black cycle, then a blue path back to the starting black cycle, and get a tri-coloured cycle again, which after reversal will result in a destruction of the chosen black cycle. If there are no black cycles left without another colour cycle inbetween them, then we just do the cycle reversal between a black and a blue cycle exactly the way described above.

That way we can obtain a non-crossing Schnyder wood with only one red, one black and one blue cycles.

4.2 Illustration of the reversal for non-crossing Schnyder woods

Here is a visualization of the steps. We see the starting point of step 1 in the figure 5, and how the toroidal triangulation looks after reversal in figure 6.

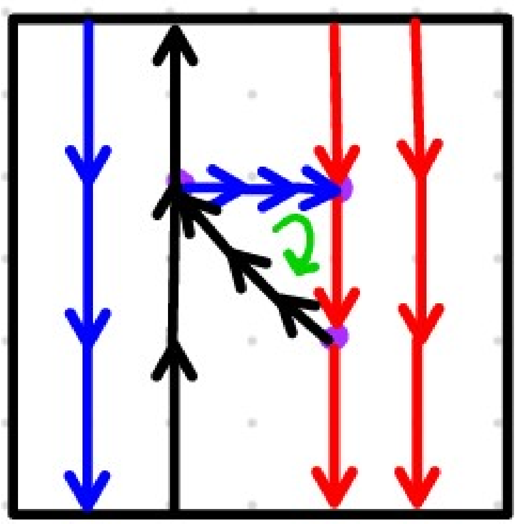

Figure 5: Starting point of step 1

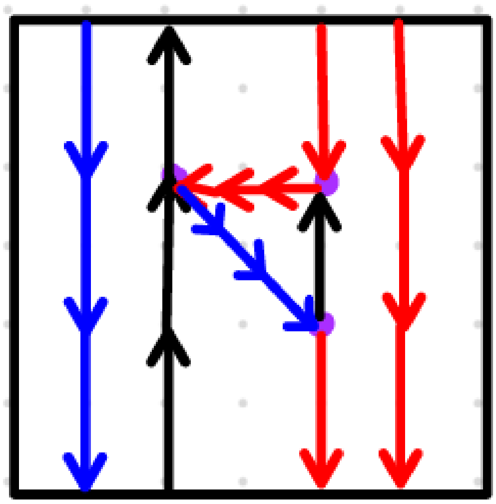

Figure 6: After reversal of step 1

Then, we see the starting point of step 2 in the figure 7, and how the toroidal triangulation looks after reversal in figure 8.

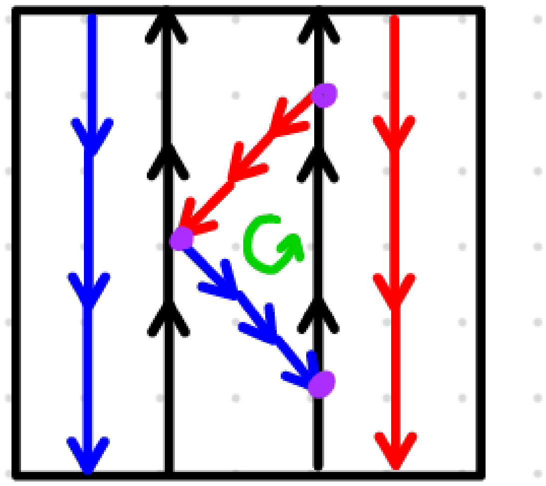

Figure 7: Starting point of step 2

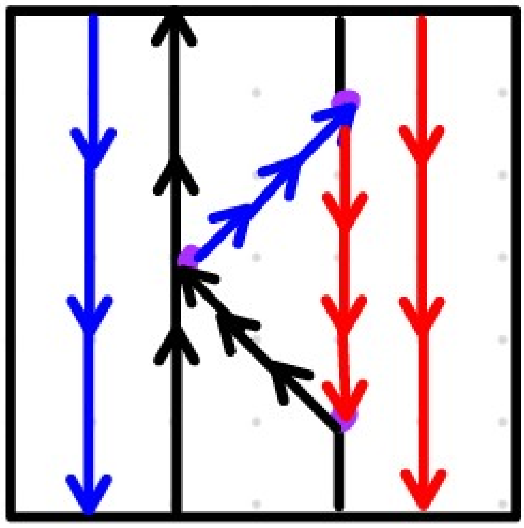

Figure 8: After reversal of step 2

4.3 Proof of correctness

We note that the following algorithm always works. First, we can definitely find a Schnyder wood corresponding to the conditions mentioned by using algorithms from the paper [3]. We note that it is always possible to choose a rightmost black cycle due to the graph having multiple by construction, and it is possible to find such a cycle with a red cycle on the right with no other cycles in between since there is one one blue cycle, so red and black must be next to each other somewhere in the graph. We can proceed with the algorithm of choosing edges due to Schnyder woods conditions - every vertex must have exactly one black, one blue, one red outgoing edge. The reversal can be performed without making the Schnyder wood invalid [2]. We have looked at all possible cases and also clearly proven that no new cycles are going to be created, therefore this algorithm works perfectly.

4.4 An algorithm for making a half-crossing Schnyder wood connected

After using the algorithms from Luca's Castelli's Aleardi's et al. work [3], we can obtain a half-crossing Schnyder wood with (without the loss of generality) one red cycle that is crossing both the black and the blue cycles, which there are multiple of, and the blue and the black cycles are not crossing each other.

We first look at the case when we have a black cycle on the right of a blue cycle with no other cycles inbetween. Similarly, as in the previous algorithm, we draw a tri-coloured cycle between them, we start from the blue cycle and find a black path towards the black one, afterwards return with a red path, and connect them with the part of the blue cycle. We can now distinguish two cases - the found tri-color cycle can either contain a part of the existing red cycle or not.

4.4.1 Case 1 - doesn't contain a part of the existing red cycle

If it doesn't, that means that the reversal of the cycle can be easily performed without impacting the existing red cycle, so preserving the half-crossing condition, and as proved before, this will destroy the blue cycle, but not create any new any color cycles.

4.4.2 Case 2 - does contain a part of the existing red cycle

If it does contain a part of the existing red cycle, it is important to see why there will be a new one created and the half-crossing condition will still be preserved. The reversal of the found cycle should still be performed as before, which will destroy the blue cycle. We note that the red cycle will end at one of the reversed cycle's path, and part of the cycle will become a blue edge. We note that from this vertex where the cycle got destroyed, there still must be an outgoing red edge due to Schnyder wood conditions. This red edge will be inside the reversed cycle due to Schnyder rules, so it will have to cross the previous blue cycle at some point, and it will inevitably reach the other part of the red cycle due to each vertex having an outgoing red edge by definition. To reach it, it must cross all blue and black cycles that are in its way or reach the red cycle. Therefore there will be a new red cycle that will cross all blue and black cycles, whilst the old one will be destroyed, and there will be no other new blue or black cycles.

Similarly, we proceed to destroy other blue cycles, and after having only one left, we proceed with the same algorithm as above with deleting all black cycles. Afterwards, we obtain a Schnyder wood with only one red, one blue and one black cycles that is half-crossing, and therefore now connected.

4.5 Illustration of the reversal for half-crossing Schnyder woods

Here is a visualization of the steps. We see the starting point of case 1 in the figure 9, and how the toroidal triangulation looks after reversal in figure 10.

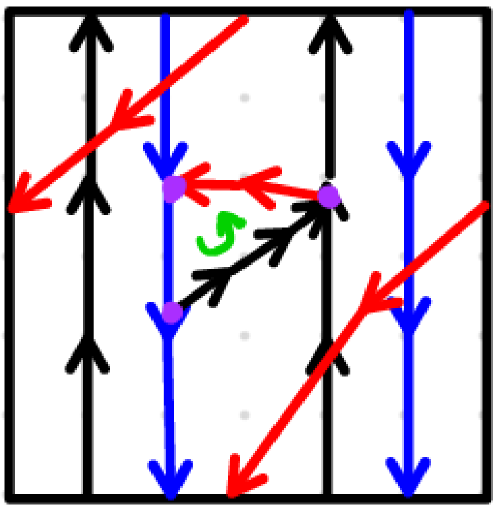

Figure 9: Starting point of case 1

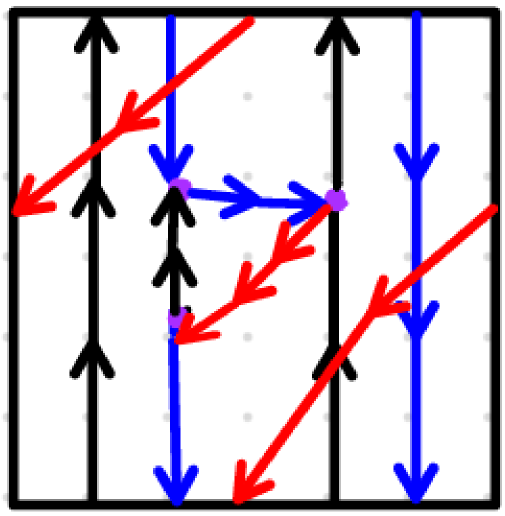

Figure 10: After reversal of case 1

Then, we see the starting point of case 2 in the figure 11, and how the toroidal triangulation looks after reversal in figure 12. We can also see the new red cycle that contains the edges from the inside of the reversed cycle.

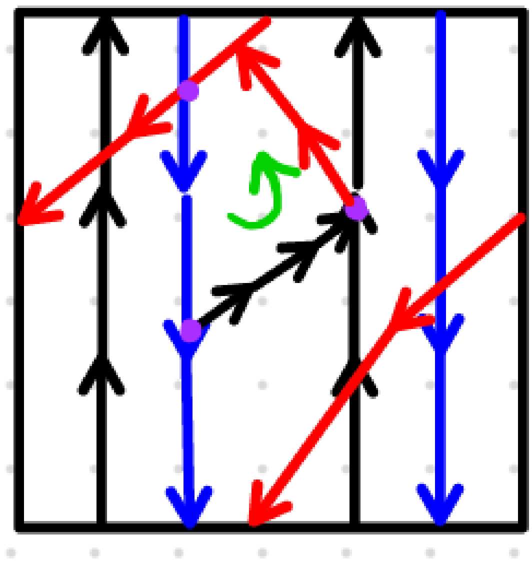

Figure 11: Starting point of case 2

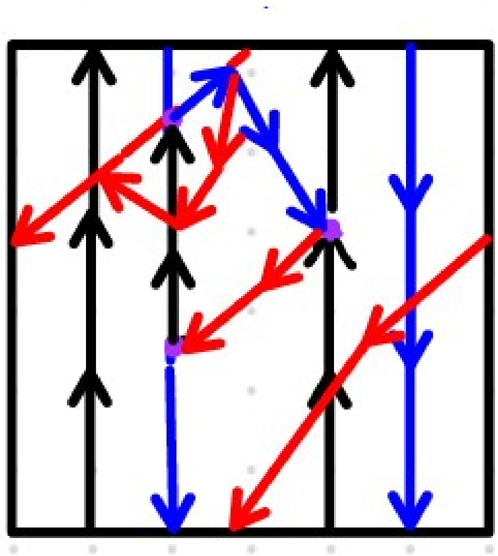

Figure 12: After reversal of case 2

4.6 Proof of correctness

Similarly as the first algorithm, we can definitely find a Schnyder wood corresponding to the conditions mentioned by using algorithms from the paper of Luca's Castelli's Aleardi's et al.[3],. We note everything that is done in the algorithm is possible by construction or Schnyder woods conditions and rules. The reversal can be performed without making the Schnyder wood invalid [2]. We have looked at all possible cases and also clearly proven that no new cycles are going to be created, therefore this algorithm works perfectly.

5 Concluding remarks: how to deal with crossing Schnyder woods

In this manuscript, we have dealt with non-crossing and half-crossing Schnyder woods. We note that all operations we have used are linear. While performing the algorithm, we first detect the multicoloured cycle, which is finding and following a path, hence linear. Then, we perform the reversal of the cycle, which is linear as well. Therefore it is highly suggested that using similar algorithms, it is possible to make a crossing Schnyder wood connected as well. It is definitely possible to use the facts that all monochromatic cycles of the same color are always parallel and non contractible. Using Schnyder conditions, it is expected that similar steps should lead to successfully obtaining a toroidal triangulation with connected crossing Schnyder woods as well.

6 Bibliography

[6, 2, 8, 5, 4, 7, 3]

7 References

[1] Luca Castelli Aleardi. Toroidal drawing iii: Toroidal schnyder woods, 202X. Lecture presentation.

[2] Enno Brehm. *3-orientations and Schnyder 3-tree-decompositions*. PhD thesis, Freie Universit"at Berlin, 2000.

[3] Luca Castelli-Aleardi, Éric Fusy, Jyh-Chwen Ko, and Razvan S. Puscasu. Computation of toroidal schnyder woods made simple and fast: from theory to practice. In *to appear in the Proc. of the 41th Int. Symposium on Computational Geometry (SoCG 2025)*, 2025.

[4] Reinhard Diestel. *Graph Theory*. Springer, 2000.

[5] Stefan Felsner. *Geometric Graphs and Arrangements*. Springer, 2004.

[6] Eric Fusy. *Combinatoire des cartes planaires et applications algorithmiques*. PhD thesis, Ecole Polytechnique, 2007.

[7] Daniel Gonçalves and Benjamin L'evêque. Toroidal maps: Schnyder woods, orthogonal surfaces and straight-line representations. *Discrete & Computational Geometry*, 51(1):67–131, 2014.

[8] Benjamin L'evêque. *Generalization of Schnyder woods to orientable surfaces and applications*. Hdr thesis, Université Grenoble Alpes, 2016.