Title

The Stable Andrews—Curtis Conjecture and Generic 2-Polyhedra

ISBN

9798276023830

Author

Fagan, Lucas

Publication Date

2025-11-25

Peer reviewed|Thesis/dissertation

University of California
Santa Barbara

The Stable Andrews–Curtis Conjecture and Generic 2-Polyhedra

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy
in
Mathematics
by

Lucas Fagan

Committee in charge:

Professor Zhenghan Wang, Chair
Professor Jon McCammond
Professor Fedya Manin
Professor Stephen Bigelow

December 2025

The dissertation of Lucas Fagan is approved.

Professor Jon McCammond

Professor Fedya Manin

Professor Stephen Bigelow

Professor Zhenghan Wang, Chair

November 2025

The Stable Andrews–Curtis Conjecture and Generic 2-Polyhedra

by

Lucas Fagan

To Rob and the Big Sweetie

Acknowledgements

I have really enjoyed my time in graduate school at UCSB—it's been fantastic from start to finish—and there are many people to whom I owe my gratitude.

First and foremost, I would like to thank my advisor, Zhenghan Wang. It's truly been one of the greatest joys of my academic career to be his student. I have learned so much from him, and I will always be grateful the opportunity to work and chat with him. As a scientist, mentor, and human being, I aim to be like him. He is the best.

I would also like to thank the rest of my committee: Jon McCammond, Fedya Manin, and Stephen Bigelow. In particular, I would like to thank Jon McCammond for his ideas and fun discussions on fake surfaces.

Thank you also to Yang Qiu for many great discussions and collaborations, and for always answering my questions.

And, thank you to Medina Price for being extraordinarily helpful with every stage of the PhD. I couldn't have done it without her guidance.

Next, I would like to thank Addy for being by my side for the whole journey, and my family for their constant, unconditional love and support throughout this process. I've been blessed by a strong support system in my personal life, which has made my whole journey so much easier.

I'd also like to thank all my friends: I've met some great people both through the PhD and on the tennis court, and they are a big reason I've had such a fun time during my PhD.

I also wouldn't be here without the support of so many teachers over the years, in math and other subjects: Dr. Purdy, Mr. Brescher, Dr. B, Mr. Lovelock, Dr. Rose, Dr. Leslie, and many more. I've met so many incredible people who have supported me one way or another, mostly in ways they will never know. To all these people, I am extremely grateful.

Finally, thank you to the town of Santa Barbara—a true paradise on Earth, and I am humbled to have had the opportunity to live in such a place.

Lucas Fagan

lucasfagan@ucsb.edu - 908.451.9384

EDUCATION

PhD Candidate in Mathematics, University of California, Santa Barbara Nov 2025
- Thesis: The Stable Andrews-Curtis Conjecture and Generic Two-Polyhedra
- Advisor: Zhenghan Wang

Awards
- Microsoft Graduate Fellowship (2024, 2025)

MA in Mathematics, UCSB 2023

BS in Mathematics, Duke University 2021
- Thesis: Schur Polynomials and Crystal Graphs
- Advisor: Spencer Leslie

Awards
- Graduation with distinction
- PRUV Fellow

BS in Computer Science, Duke University 2021

RESEARCH INTERESTS

Computational topology, reinforcement learning, computer-assisted proofs, machine learning, (stable) Andrews-Curtis conjecture, Zeeman conjecture, fake surfaces, quantum topology, low-dimensional topology

EXPERIENCE

Visiting Researcher in AI & Mathematics, Caltech July 2025–Present
- Apply long-horizon, sparse rewards reinforcement learning techniques to address complex research-level math problems on Sergei Gukov's team

Graduate Researcher, UC Santa Barbara 2021–Present
- Investigate the Andrews-Curtis conjecture and Zeeman conjectures via computational topology, classical search algorithms, and reinforcement learning algorithms
- Show potential counterexamples to be consistent with the conjectures, and construct more challenging potential counterexamples

Teaching assistant, UC Santa Barbara

- Led discussions and designed problems to facilitate learning in Linear algebra with Applications (Fall 2021, Fall 2022), Differential Equations (Summer 2022, Spring 2023), Calculus with Applications I (Winter 2022), Calculus with Applications II (Spring 2022, Winter 2023), Transition to Proofs (Fall 2024), Intro to Topology (Summer 2024)

PUBLICATIONS/PREPRINTS

- A. Shehper et al., "What Makes Math Problems Hard for Reinforcement Learning: A Case Study," arXiv:2408.15332 (to appear at NeurIPS 2025)
- L. Fagan et al., "Stable Andrews-Curtis Conjecture via Fake Surfaces and Zeeman Conjecture," arXiv:2412.12293
- L. Fagan et al., "Classification of Cellular Fake Surfaces," arXiv:2406.09439 (to appear in *Journal of Experimental Mathematics*)
- L. Fagan, J. McCammond, "Fake surfaces for Potential Counterexamples to the Stable Andrews-Curtis conjecture," (In Preparation—Manuscript available upon request)

TALKS

- **"Contractible Fake Surfaces and the Andrews–Curtis and Zeeman Conjectures."** Special Session: *Advances in Quantum Topology, Tensor Categories, and Diagrammatic Algebras*, October 2025.
- **"T and U moves to Group Presentations,"** *UCSB*, February 2025
- **"The Stable Andrews-Curtis Conjecture, Unknot Diagrams, and Potential Counterexamples,"** *UCSB*, November 2024
- **"Fake Surfaces for AK(n),"** *UCSB*, June 2023
- **"Contractible Fake Surfaces and the Andrews-Curtis and Zeeman Conjectures,"** *UCSB*, April 2023
- **"The Stable Andrews-Curtis Conjecture,"** *UCSB*, February 2023
- **"Fake Surfaces and the Zeeman Conjecture,"** *UCSB*, October 2023

TECHNICAL SKILLS

- Programming: Python, Java
- ML: PyTorch, JAX
- Computational Mathematics: SageMath

Abstract

The Stable Andrews–Curtis Conjecture and Generic 2-Polyhedra

by

Lucas Fagan

The Andrews–Curtis conjecture, proposed in 1965, asks whether any balanced presentation of the trivial group can be transformed to the trivial presentation using a specific set of moves. This thesis investigates both the standard conjecture and its stable variant from algorithmic and topological perspectives.

We develop a state-of-the-art algorithm for trivializing group presentations and apply it to resolve several open problems. We prove that all but two presentations in the Akbulut–Kirby series can be reduced in length, and we resolve various potential counterexamples in the Miller–Schupp series, including three infinite subfamilies.

The stable Andrews–Curtis conjecture allows additional stabilization moves and is equivalent to a topological statement: any contractible 2-dimensional polyhedron can be 3-deformed to a point. We investigate this formulation through the systematic study of fake surfaces, which are generic 2-polyhedra. We present a complete classification of acyclic cellular fake surfaces up to complexity 4, and classify complexity 5 surfaces without small disks. This classification reveals 2, 17, 238, and 4618 distinct surfaces of complexities 1 through 4, respectively.

Using this classification, we establish an inductive scheme for proving new cases of the stable Andrews–Curtis and Zeeman conjectures and verify it holds up to complexity 5. As consequences, we prove the contractibility conjecture for all acyclic fake surfaces of complexity 4 and verify the embedded disk conjecture up to complexity 5.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The Andrews–Curtis conjecture, proposed by James J. Andrews and Morton L. Curtis in 1965 [2], is a group-theoretic conjecture concerning whether any balanced presentation of the trivial group $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_n \rangle$ can be transformed into the trivial presentation $\langle x_1, \ldots, x_n \mid x_1, \ldots, x_n \rangle$ through a sequence of specific moves called AC-moves. The conjecture remains open but is widely believed to be false.

In Chapter 2, we approach this problem from a computational perspective. We build a state-of-the-art tool to trivialize group presentations and use it to investigate various potential counterexamples that have remained open in the literature. In particular, we prove that all but two presentations in the Akbulut-Kirby series [1] of potential counterexamples $\mathrm{AK}(n) = \langle x, y \mid xyx = yxy, x^n = y^{n+1} \rangle$ can be reduced in length, and we resolve various potential counterexamples in the Miller–Schupp series [7] $\mathrm{MS}(n, w) = \langle x^{-1}y^n x = y^{n+1}, x = w \rangle$, including three infinite subfamilies.

The Andrews–Curtis conjecture has a variant, called the stable Andrews–Curtis conjecture, which allows for a stabilization move and its inverse in addition to the standard AC-moves. It is unknown whether this variant is weaker than the standard Andrews–Curtis conjecture. The stable conjecture is especially important due to its deep connections to topology [53]. It can be stated topologically: any contractible 2-dimensional polyhedron can be 3-deformed to a point, where 3-deformed means collapsed through a sequence of elementary moves in at most three dimensions. This formulation is closely related to the Zeeman conjecture [56], which states that for any 2-polyhedron P, we can collapse $P \times I$ to a point.

In Chapter 3, we discuss the stable Andrews–Curtis conjecture and present connections to knot diagrams, which have been thought to provide potential counterexamples to the equivalence of the stable and non-stable variants of the conjecture. Using the algebraic formulation of the conjecture, we prove that this approach cannot succeed in certain cases.

In Chapter 4, we turn to the topological formulation of the stable Andrews–Curtis conjecture. The conjecture deals with arbitrary contractible 2-polyhedra, but it suffices to study generic 2-polyhedra, which are called fake surfaces [30]. For this reason, understanding fake surfaces is important for understanding the stable Andrews–Curtis conjecture. We introduce fake surfaces as interesting objects in their own right, describe several of their properties, and initiate an inductive scheme aimed at proving the stable Andrews–Curtis conjecture. We verify this scheme for fake surfaces up to complexity 5 and discuss its potential for proving that proposed counterexamples do indeed satisfy the conjecture.

In Chapter 5, we present a classification of low-complexity fake surfaces, which forms the foundation of the inductive scheme above. We also establish several properties of such surfaces, prove the contractibility conjecture for acyclic fake surfaces of complexity 4, extending the work of Ikeda [21] and Robertson [40], and verify the embedded disk conjecture up to complexity 5.

Finally, in Chapter 6, we outline several directions for further research motivated by the results of this dissertation.

Chapter 2

The Andrews-Curtis Conjecture

In this chapter, we begin by introducing the Andrews–Curtis conjecture in Section 2.1, followed by an overview of key ideas and previous work in Section 2.2. In Section 2.3, we present our state-of-the-art algorithm for trivializing presentations and discuss the results it produces as well as its limitations.

2.1 Introduction

A central object of study in combinatorial group theory is a group presentation, which defines a group as a quotient of a free group on a set of generators by the normal subgroup generated by a set of relators. A particularly interesting class of presentations are *balanced presentations*, where the number of generators is equal to the number of relators [28]. The Andrews–Curtis conjecture, proposed in 1965 by James J. Andrews and Morton L. Curtis, asks whether all balanced presentations of the trivial group can be transformed to the trivial presentation using certain moves [2].

We first define the allowed transformations, or "moves," that can be applied to the set of relators in a presentation.

Definition 2.1 (Andrews–Curtis Moves)**.** Let $\mathcal{P} = \langle x_1, \ldots, x_n \mid r_1, \ldots, r_n \rangle$ be a balanced group presentation. The following three moves are called **Andrews–Curtis moves** or **AC-moves**:

(AC1) Replace a relator r_i with its inverse, r_i^{-1}.

(AC2) Replace a relator r_i with the product $r_i r_j$ for some $j \neq i$.

(AC3) Replace a relator r_i with its conjugate $g^{-1} r_i g$ for any generator $g \in \{x_1^{\pm 1}, \ldots, x_n^{\pm 1}\}$.

Two presentations are said to be *Andrews–Curtis equivalent* or *AC-equivalent* if one can be obtained from the other by a finite sequence of these moves.

A presentation is said to be *Andrews–Curtis trivial* or *AC-trivial* if it is Andrews–Curtis equivalent to $\langle x_1, \ldots, x_n \mid x_1, \ldots, x_n \rangle$.

The conjecture states that for a fixed number of generators n, all balanced presentations of the trivial group are AC-equivalent:

Conjecture 2.1 (The Andrews–Curtis Conjecture)
Any balanced presentation of the trivial group, $\mathcal{P} = \langle x_1, \ldots, x_n \mid r_1, \ldots, r_n \rangle$, is Andrews–Curtis equivalent to the trivial presentation $\langle x_1, \ldots, x_n \mid x_1, \ldots, x_n \rangle$ [2].

The conjecture has proven to be difficult to resolve and has remained a significant open problem in the field. The general consensus is that the conjecture is false, as many potential counterexamples have been found that have eluded all attempts at trivialization.

In this chapter, we explore our work on computational methods that have been developed to tackle this problem. We will explore algorithms designed to search the space of presentations, heuristics for guiding these searches, and the results of applying these methods to known potential counterexamples.

Remark (Substitution Move)
AC-moves allow us to use a relator to make substitutions in other relators. This is useful

in simplifying trivialization paths, as a substitution often combines multiple AC-moves and is a natural way to modify group presentations.

To be precise, let $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{i-1}, w^{-1}w', r_{i+1}, \ldots, r_n \rangle$ be a balanced presentation of the trivial group with some words w, w' in the generators. Then we may replace any occurrence of w in a relator r_j, $j \neq i$, with w' through a combination of AC-moves.

Indeed, to see how to realize this through AC-moves, consider the example substituting w' for w in the relator w_1ww_2: first, conjugate (by repeated application of AC3) to obtain w_2w_1w; then multiply by $w^{-1}w'$ to obtain w_2w_1w' (using AC2); and finally, conjugate again to obtain the desired result $w_1w'w_2$ (by repeated application of AC3). For a relator of the form $w_1ww_2 \cdots w_{l-1}ww_l$, we may similarly substitute any number of occurrences of w with w'. The number of AC-moves packaged into the substitution move is linear in the lengths of the words involved.

Remark (Automorphisms)

It is natural to consider the additional move of applying an automorphism to the underlying free group. This extended move set is sometimes referred to as Q^* moves in the literature. It is unknown whether adding this move results in an equivalent move set or not. However, something weaker holds: if a balanced presentation of the trivial group P can be transformed into the trivial presentation using AC-moves and automorphisms, then P is AC-trivial. This is discussed at greater length in [38], [6], and [34].

2.2 Background

2.2.1 The AC Graph

The difficulty in finding a theoretical proof or disproof has naturally led to computational approaches. The general strategy is to search for a "path" of valid AC-moves from a given balanced presentation of the trivial group to the trivial presentation. This is, in essence, a search problem on an infinite graph where the vertices are presentations and the edges are the AC-moves. Thus, there are numerous search algorithms from computer science that can be applied to prove the AC-triviality of various presentations.

However, this search is computationally intensive. A significant challenge is that the length of the relators can grow substantially during the transformation process, even if the target is the simple trivial presentation.

Most such searches focus on the simplest case of two-generator presentations. This is because, for presentations with n generators, there are $n+n(n-1)+2n^2 = 3n^2$ AC-moves, and so with more generators, the branching factor of a search increases quadratically. Using the language of the aforementioned AC graph, the number of adjacent vertices grows as $O(n^2)$.

Alternative Graphs

Using a different set of moves that are combinations of AC-moves admits a different graph in which trivialization paths may be shorter. For example, a natural choice is to unify the inversion of (AC1) and the multiplication (AC2) into the moves $r_i \to r_i r_j^{\pm 1}$ for $i \neq j$. This keeps the number of moves the same but may reduce path lengths. These moves are called AC′-moves in [42]. Here, it is clear that we can recover the original (AC1), as

$$(r_i, r_j) \to (r_i r_j^{-1}, r_j) \to (r_i r_j^{-1}, r_j r_i r_j^{-1}) = (r_i r_j^{-1}, r_i) \to (r_i r_j^{-1} r_i^{-1}, r_i) = (r_j^{-1}, r_i).$$

It is also not necessary to consider moves that generate each of the original AC-moves, as long as the moves can be expressed as AC-moves. This is because we only seek trivializations, and it may be fruitful to sacrifice certain transformations in order to facilitate others. For example, one can instead view the graph as having edges that are substitution moves as mentioned in the previous section. In this case, the author is not aware of any proof that one can execute each of the AC-moves through substitutions. In particular, one can recover the multiplication move (AC2) by allowing "trivial substitutions," i.e., where $w = 1$ (viewing r_i as $r_i = 1$ and r_j as $r_j = 1$ and substituting), but it is still not clear whether one can recover a move (AC3) followed by (AC2), i.e., $r_i \to r_i g r_j g^{-1}$ for certain values of g.

2.2.2 Potential Counterexamples

The most famous family of potential counterexamples to the AC conjecture comes from Selman Akbulut and Robion Kirby from 4-manifold topology [1]. This one parameter family of presentations is defined as follows:

$$\mathrm{AK}(n) = \langle x, y \mid xyx = yxy, x^n = y^{n+1} \rangle,$$

where $n \geq 1$ is a positive integer. It was inspired by the $n = 4$ case, which comes from a handle decomposition of the Akbulut–Kirby 4-sphere.

It is easy to see that AK(1) is AC-trivial: from $\langle x, y \mid xyx = yxy, x = y^2 \rangle$, simply substitute $x = y^2$ into the first relation, giving $\langle x, y \mid y^5 = y^4, x = y^2 \rangle$, which is the same as $\langle x, y \mid y, x = y^2 \rangle$. Then, we invert the first relator, and multiply the second relator by y^{-1} twice, and invert the first relator again, leaving us with $\langle x, y \mid x, y \rangle$.

The second presentation in this series, AK(2), is more difficult to trivialize, but is also AC-trivial. A trivialization path was first found by Alexei Myasnikov using genetic algorithms in [34].

The remainder of the family, AK(n) for $n \geq 3$, remains a collection of potential counterexamples to the conjecture.

Remark

In [38], it is shown that for any n and any automorphism of the underlying free group φ, we have that $\varphi(\mathrm{AK}(n))$ is AC-equivalent to AK(n). Thus, this family is one case of unsolved presentations where the extended move set with automorphisms aligns with the normal AC-moves.

The presentation AK(3), which has total relator length 13, is an especially important potential counterexample: it is the unique minimal-length potential counterexample on two generators, up to AC-equivalence. Indeed, it was established in [35] that all two-generator presentations with total relator length at most 12 satisfy the conjecture, and in [18], Havas and Ramsey showed that all presentations on two generators of total relator length 13 either satisfy the conjecture or are AC-equivalent to AK(3).

A second well-known two-parameter family containing many potential counterexamples is the Miller–Schupp family [7]:

$$\mathrm{MS}(n, w) = \langle x, y \mid x^{-1}y^n x = y^{n+1}, x = w \rangle,$$

where $n \geq 1$ is a positive integer and w is a word in x and y with zero exponent sum on x.

Myasnikov, Myasnikov, and Shpilrain showed that the AK(n) family is contained within the MS(n, w) family [36]:

Proposition 2.2 ([36])

For all $n \geq 2$, AK(n) is AC-equivalent to $\mathrm{MS}(n, y^{-1}x^{-1}yxy)$.

2.2.3 Computational Approaches

Classical Searches

Two basic search algorithms from computer science are depth-first search and breadth-first search.

Depth-first search explores a branch of a search tree as far as possible before backtracking. Breadth-first search (BFS), on the other hand, explores all the neighbor nodes at the present depth before moving on to the nodes at the next depth level.

Both of these techniques have problems that lead to them struggling to efficiently trivialize presentations. Indeed, a pure depth-first search will likely become trapped exploring a very long, fruitless path in the infinite AC graph.

Breadth-first search, on the other hand, is very memory-intensive. BFS needs to store all the nodes at the current level to explore the next, and given the large branching factor of the AC graph, this will quickly exhaust available memory before many moves can be applied. It is possible to modify this search to only include presentations under a fixed length, which makes the search more feasible as the search space becomes finite. It also has the additional benefit of potentially identifying a lower bound on the longest presentation in a trivialization sequence. For example, [38] found that along any potential trivialization of AK(3), there must be a presentation of length greater than 20.

A slightly more sophisticated alternative to both of these algorithms is A^* search, which is an informed search algorithm that finds the shortest path between a starting node and a goal node in a weighted graph. It works by maintaining a priority queue of paths to be explored and at each step, it extends the path that has the lowest value of $f(p) = g(p) + h(p)$, where $g(p)$ is the cost of the path from the start node to the current node p, and $h(p)$ is a heuristic function that estimates the cost of the cheapest path from p to the goal state.

In our case, each AC-move can be considered to have a uniform cost, so it is natural to use an edge weight of 1 for each step of the path. This means $g(p)$ is simply the number of AC-moves taken to reach the presentation. One natural choice for a heuristic function $h(p)$ is to use the total length of the relators in a presentation. The goal is to reach the trivial presentation where the total relator length is minimized, so this heuristic directs the search toward shorter presentations. We can also use $g(p) = \epsilon$ as the cost function. In this case, A^* search will always prioritize shorter presentations, regardless of how far they are from the starting state. The distance from the starting state is only used to break ties between two presentations of equal length.

Genetic Algorithms

Genetic algorithms offer a non-deterministic approach to searching the vast AC graph. As mentioned above, this method was notably used by Alexei Myasnikov to find a trivialization path for AK(2) and verify that all two-generator presentations of total relator length at most 12 are AC-trivial. Genetic algorithms are inspired by the process of natural selection and operate on a population of candidate solutions, which in this context are sequences of AC-moves. Each sequence is evaluated by a "fitness function," which typically prioritizes sequences that lead to shorter relators.

The algorithm proceeds in generations. In each generation, the fittest individuals are selected to "reproduce" by creating new sequences through "crossover" (combining parts of two parent sequences) and "mutation" (randomly altering a part of a sequence). This process aims to iteratively improve the population, hopefully evolving a sequence of moves that trivializes the presentation.

Machine Learning Approaches

It is also natural to apply machine learning techniques, particularly reinforcement learning, to the Andrews–Curtis conjecture. Reinforcement learning frames the problem as an agent (the algorithm) interacting with an environment (the space of presentations). The agent takes actions (applying AC-moves) to receive rewards, which are designed to guide it toward the goal of trivializing the presentation. A key challenge in this context is the sparsity of rewards; a successful sequence of moves might be extremely long, making it difficult for the agent to learn which actions are beneficial in the long run.

Reinforcement learning has demonstrated remarkable success in solving complex search problems in other domains, most famously in game-playing with AlphaGo and AlphaZero [43]. In these contexts, the agent often receives more frequent feedback or a clearer signal of progress. For instance, in a game of Go, while the ultimate reward is winning or losing, intermediate board states can be evaluated to estimate the probability of a win, providing a denser reward signal. This frequent feedback allows the learning algorithm to more easily associate actions with positive or negative long-term outcomes. The search for AC-trivializations, however, presents a classic sparse reward problem. The only definitive positive reward comes from reaching the trivial presentation, a rare event that might require thousands of moves. Intermediate steps, such as those that increase relator length, might appear to be counterproductive, making it exceptionally difficult for an RL agent to learn the long-term value of its actions without sophisticated reward-shaping strategies.

Another machine learning approach involves using supervised learning to guide more traditional search algorithms. For example, a classifier can be trained to predict whether a particular sequence of moves is likely to lead to a simplification, helping to prune the search space and make automated reasoning more efficient. The idea is to leverage neural networks to detect patterns and guide the search, which might uncover structure that is not obvious to human mathematicians.

Recent work has shown promise in applying machine learning to theorem proving [54, 3, 39] and mathematical problem-solving [45]. Applications of machine learning to the Andrews–Curtis conjecture are explored in joint work [42].

Automated Theorem Proving

Besides these AI-driven methods, other computational strategies have been employed. One significant approach is to reframe the problem in terms of automated theorem proving. This involves translating the AC transformations into a formal logical system, allowing the use of powerful automated theorem provers to search for a proof of equivalence between a given presentation and the trivial one. This method has been competitive with specialized search algorithms and has even provided new trivializations.

2.2.4 Bridson-Lishak presentations

A significant development that has profoundly reshaped the perspective on computational searches for Andrews–Curtis trivializations came from the work of Martin Bridson and Boris Lishak [5] [24]. They constructed families of balanced presentations of the trivial group for which any sequence of Andrews-Curtis moves that transforms them into the trivial presentation must be extraordinarily long. Specifically, they demonstrated the existence of presentations where the number of required AC-moves is bounded below by a superexponential function of the presentation's total relator length.

These presentations are constructed in such a way that trivializing them is provably complex. The core idea is to embed the difficulty of solving the word problem in a carefully chosen group into the process of AC-trivialization. By doing this, Bridson and Lishak created balanced presentations that are known to be AC-trivial, yet whose trivialization paths are far beyond the reach of any conceivable brute-force or simple heuristic-based search.

The existence of the Bridson-Lishak presentations has a strong impact on the view of the failures of computational searches to trivialize potential counterexamples to the Andrews-Curtis conjecture. It may seem, at first, that the failure to trivialize a potential counterexample like AK(3) after extensive searching could be interpreted as evidence that it might indeed be a true counterexample. However, the Bridson-Lishak results provide an alternative explanation: the presentation might be trivial, but the path to triviality is simply too complex for current search algorithms to find. It highlights that the absence of a short trivialization path does not imply the absence of a trivialization path altogether. This reinforces the need for more sophisticated search techniques that can navigate the incredibly complex and vast landscape of the AC graph, potentially by identifying and traversing incredibly long and non-obvious paths to simplification. In particular, it emphasizes that "supermoves" [42], i.e., moves that combine many AC-moves into one, may be necessary to trivialize many AC-trivial presentations.

2.3 Computational Results

2.3.1 Greedy Search Algorithm

We built a state-of-the-art solver to trivialize group presentations with AC-moves. It uses the ideas of A^* search from Section 2.2.3, where the moves are substitution moves, $g(p) = \epsilon$, and $h(p)$ is the total length of the relators. In other words, the search always prefers to explore presentations with shorter total length over longer ones, and ties are broken by distance from the starting presentation. For this reason, we call this algorithm *greedy search*, as in [42]. We describe the algorithm in more detail and provide some pseudocode below.

Remark

Trivialization paths discovered by greedy search are not necessarily optimal in terms of path length, since there might be a shorter path that goes through longer presentations. However, trivialization paths discovered by greedy search are optimal in terms of the length increase during the path. In particular, if n is the total length of the longest presentation in a greedy search trivialization of a presentation, then it is impossible to trivialize that presentation by only using presentations of length at most $n-1$. This also implies that if greedy search views a presentation of total length n before failing to trivialize a presentation, then the presentation cannot be trivialized without the total length increasing to at least n.

First, to minimize redundancy, every relator is reduced by freely canceling adjacent inverse pairs, including cyclic cancellations. Each reduced word is then placed in canonical form by considering all cyclic permutations and inversions, and selecting the lexicographically minimal representative. The lexicographically smaller of the two canonical relators is then set to be r_1, and the other r_2. This ensures that equivalent relators are always identified with a unique canonical word. For example, this step views $(x^{-1}y^{-1}x, x^{-1}y)$ as identical to $(x^{-1}y, y)$ or (y, xy^{-1}).

From a given presentation, the algorithm generates neighbors by applying all nontrivial substitutions. In particular, it identifies positions where a suffix of a cyclic permutation of one relator cancels with a prefix of a cyclic permutation of the other (or its inverse), and concatenates the words together to form a new candidate relator. Each resulting pair that is below a configured maximum length is reduced, canonicalized, and placed into a priority queue if it has not already been encountered. The priority, as mentioned above, is determined by the total word length, so that the search explores simpler configurations first, with ties broken by distance from the starting presentation.

The procedure terminates if a pair of relators both reduce to words of length one (i.e., a presentation whose canonical form is equivalent to that of (x, y)). If no such pair is found within the search bounds (on word length or number of nodes visited), the algorithm reports the minimal presentations discovered. In this way, the solver provides a systematic method for probing possible Andrews–Curtis reductions while efficiently navigating the enormous search space of relator pairs.

The pseudocode is given in Algorithms 1, 2, and 3 below.

Algorithm 1 Relator Canonicalization

```
function CanonicalRelator(r)
    r_min ← minimal lexicographic rotation of r (Booth's algorithm)
    inv_min ← minimal lexicographic rotation of inverse(r)
    if r_min > inv_min then return inv_min
    elsereturn r_min
    end if
end function
function CanonicalPair(r1, r2)
    cr1 ← CanonicalRelator(r1)
    cr2 ← CanonicalRelator(r2)
    if (|cr1| > |cr2|) or (|cr1| = |cr2| and cr1 > cr2) then
        Swap(cr1, cr2)
    end if
    return (cr1, cr2)
end function
```

Algorithm 2 Neighbor Generation

```
function GetNeighbors(r1, r2)
    results ← {}
    for all candidates c ∈ {r2, inverse(r2)} do
        for all rotations of r1 and c do
            if last(rot1) is inverse of first(rot2) then
                neigh ← concat(rot1, rot2)
                Add (neigh, r2) and (r1, neigh) to results
            end if
        end for
    end for
    return results
end function
```

Algorithm 3 ACRelatorSolver

```
procedure Initialize(r1, r2, max_nodes, max_len)
    Convert input to arrays
    Reduce relators and compute initial canonical pair
    Initialize priority queue and visited set
end procedure
function Solve
    while queue not empty and nodes < max_nodes do
        Pop (r1, r2) with minimal priority
        if |r1| = 1 and |r2| = 1 then
            Return path to trivial relators
        end if
        for all neighbors of (r1, r2) do
            Reduce and canonicalize neighbors
            if length below threshold and unseen then
                Add to queue and mark visited
            end if
        end for
    end while
    Return failure if no trivial relators found
end function
```

2.3.2 Greedy Search Performance on $\mathbf{MS}(n, w)$ and $\mathbf{AK}(n)$

$\mathbf{MS}(n, w)$ family

To test the performance of this algorithm on the infinite family $\mathrm{MS}(n, w)$, we consider the 1190 presentations that satisfy $n \leq 7$ with $\mathrm{length}(w) \leq 7$. In general, we find that, for a fixed w, varying n up to this bound gives us a sufficient signal for performance in that one-parameter family.

Greedy search was also able to solve all presentations of length less than 14 in the dataset. We note that while Proposition 2.2 shows that the AC-equivalence class of AK(3) appears in the dataset, none of these instances have length 13. Indeed, $\mathrm{MS}(3, y^{-1}x^{-1}yxy)$ has length 15.

At length 14, there are six presentations that greedy search could not solve. Four of these, namely,

$$\langle x, y \mid x^{-1}y^2x = y^3, x = x^{-2}y^{-1}x^2y^{\pm 1}\rangle$$

and

$$\langle x, y \mid x^{-1}y^3x = y^4, x = y^{\pm 1}x^2y^{\pm 1}\rangle,$$

are AC-equivalent to AK(3), while the other two,

$$\langle x, y \mid x^{-1}y^2x = y^3, x = yx^2yx^{-2}\rangle$$

and

$$\langle x, y \mid x^{-1}y^2x = y^3, x = yx^2y^{-1}x^{-2}\rangle,$$

could not be related to AK(3) or to the trivial presentation through any path involving only presentations with both relators having length less than 24. Thus, these two presentations of length 14 are new potential counterexamples. Indeed, they are the minimal-length potential counterexamples outside the AC-equivalence class of AK(3).

Overall, greedy search solved more than half of the 1190 presentations.

Notably, greedy search solved all $n = 1$ presentations in the aforementioned dataset. This motivated us to prove the AC-triviality of $\mathrm{MS}(1, w)$ for all w:

Theorem 2.3

$\mathrm{MS}(1, w)$ is AC-trivial for all w.

Proof. We have

$$\mathrm{MS}(1, w) = \langle x, y \mid x^{-1}yx = y^2, x = w\rangle,$$

where w has exponent sum 0 in x. We can rewrite the first relator in the following four ways:

(i) $yx = xy^2$

(ii) $x^{-1}y = y^2x^{-1}$

(iii) $x^{-1}y^{-1} = y^{-2}x^{-1}$

(iv) $y^{-1}x = xy^{-2}$

We can substitute these equations into the second relator to move around all occurrences of x and x^{-1}: relations (i) and (iv) move x to the left, while relations (ii) and (iii) move x^{-1} to the right. We continue until the second relator becomes $x^{a-1}y^bx^{-a}$ for some $a, b \in \mathbb{Z}$. Through conjugation, this simplifies to $x = y^b$, which, when substituted into the first relator, leads to a trivialization of the presentation. □

The following result follows from this theorem.

Theorem 2.4

$\mathrm{MS}(n, w_\star)$ for $w_\star = y^{-1}xyx^{-1}$ is AC-trivial for all n.

Proof. In the presentation,

$$\mathrm{MS}(n, w_\star) = \langle x, y \mid x^{-1}y^n x = y^{n+1}, x = y^{-1}xyx^{-1}\rangle$$

we can rewrite the second relation as $y^{-1}xy = x^2$, and apply the automorphism $x \leftrightarrow y$ to get

$$\mathrm{MS}(n, w_\star) = \langle x, y \mid x^{-1}yx = y^2, y^{-1}x^n y = x^{n+1}\rangle.$$

This presentation is the same as $\mathrm{MS}(1, y^{-1}x^n yx^{-n})$, which is AC-trivial by Theorem 2.3.

□

AC-trivializations of $\mathrm{MS}(n, w_\star)$ for $n = 3, 4, 5, 6, 7, 8$ were recently obtained using automated theorem proving in [25]. Here, we have obtained AC-trivializations of this family for all n. However, it is still useful to compare the computational performance of greedy search with automated theorem proving.

In [25], the author writes:

> The case of $\mathrm{MS}(7, w_\star)$ poses considerable challenge for any computational approach. We were not able to find AC-simplification using automated reasoning with IG encoding (unlike the cases with $n \leq 6$).
>
> We first were able to confirm AC-trivialization of $\mathrm{MS}(7, w_\star)$ using automated reasoning with EN encoding. It took Prover9 42681s to complete the search.

The aforementioned greedy search trivializes the presentation $\mathrm{MS}(7, w_\star)$ in 5.2s, which is over 8,200× faster than the 11.9 hours mentioned above.[1] The trivialization required visiting fewer than 20,000 presentations, and the trivialization path had length 493.

In [26], the same author also describes a trivialization of $\mathrm{MS}(8, w_\star)$:

[1]The authors of the original paper conducted their experiments using Prover9 on Intel i5-7200U (2.71 GHz, Windows 10) and Intel i7-6700 (3.40 GHz, Rocky Linux). Our experiments were performed on an Apple M2–based MacBook Pro (8-core, 8 GB RAM) running macOS. Because these systems differ substantially in architecture (x86_64 vs ARM64), OS environments, and potentially Prover9 builds, clock frequencies and raw hardware specifications are not directly comparable. This speedup should be interpreted as an empirical observation for this instance rather than a strict comparison of hardware performance.

It took Prover9 189707s and 9Gb of memory to find a proof. The length of the proof is 6270 steps.

Our aforementioned greedy search trivializes this presentation in 50s, visiting fewer than 150,000 presentations, with a path length of 1023. This is a 3,800× speedup compared to the almost 53 hours described above.[2]

Greedy Search Performance on AK(n)

While the greedy search algorithm trivializes AK(1) and AK(2) instantly, it is unable to trivialize any other presentation in the series. However, for AK(5), greedy search quickly finds a way to reduce the length from the starting length of 17 to 16. It can similarly reduce the length of AK(6) from 19 to 17. Examining these paths, we found the following result:

Theorem 2.5

For every $n \geq 2$, AK(n) is AC-equivalent to the presentation

$$\langle x, y \mid x^{-1}yx = xyx^{-1}y \ , \ xy^{n-1}x = yxy \rangle,$$

of length $n + 11$. This gives a reduction in length of AK(n) for all $n \geq 5$.

To prove this theorem, we will use Proposition 2.2 and prove two additional theorems (Theorems 2.6 and 2.7). From these results, Theorem 2.5 follows immediately.

Theorem 2.6

For each fixed $n > 0$, all members of the one-parameter family of presentations MS(n, w_k) are AC-equivalent. Here, $w_k = y^{-k}x^{-1}yxy$ is parameterized by $k \in \mathbb{Z}$.

Proof. Starting with the presentation,

$$\mathrm{MS}(n, w_k) = \langle x, y \mid x^{-1}y^n x = y^{n+1}, x = y^{-k}x^{-1}yxy \rangle,$$

[2]See the disclaimer on hardware and runtime comparability in the previous footnote

we will show that for any fixed n and k, $\mathrm{MS}(n, w_k)$ is AC-equivalent to $\mathrm{MS}(n, w_{k+1})$.

First, note that the first relation in $\mathrm{MS}(n, w_k)$ can be rearranged in the following three ways:

(i) Multiplying by $y^{k-n}x$ from the left and by y^{-1} from the right gives

$$y^k x y^{-1} = y^{k-n} x y^n.$$

(ii) Multiplying by y^{-1} from the left and by $x^{-1}y^{-1}$ from the right, and inverting the relation gives

$$y^{-(n-1)} x y = y x y^{-n}.$$

(iii) Multiplying by $y^{-1}x$ from the left and by y^{k-n} from the right, and inverting the relation gives

$$y^{n-k} x^{-1} y^{-(n-1)} = y^{-(k+1)} x^{-1} y.$$

Now we can rearrange the second relation to get $y^k x y^{-1} = x^{-1} y x$. Substituting (i) gives $y^{k-n} x y^n = x^{-1} y x$, which we can rewrite as $y x y^{-n} = x y^{k-n} x$. Now substituting (ii) gives $y^{-(n-1)} x y = x y^{k-n} x$, which we may rewrite as $y^{n-k} x^{-1} y^{-(n-1)} = x y^{-1} x^{-1}$. Finally, substituting (iii) gives $y^{-(k+1)} x^{-1} y = x y^{-1} x^{-1}$, which is equivalent to $x = y^{-(k+1)} x^{-1} y x y = w_{k+1}$. □

Theorem 2.7

For each $n > 0$ and $k \in \mathbb{Z}$, $\mathrm{MS}(n, w_k)$ is AC-equivalent to the presentation

$$P(n,k) = \langle x, y \mid y^{n-k-1} x^{-1} y x = x y x^{-1} y^{n-k}\ ,\ x = y^{-k} x^{-1} y x y \rangle,$$

of total length $|k| + |n-k| + |n-k-1| + 11$. For each fixed n, this length is minimized when $k = n-1$, which gives the presentation

$$P(n, n-1) = \langle x, y \mid x^{-1}yx = xyx^{-1}y \ , \ x = y^{-(n-1)}x^{-1}yxy \rangle,$$

of length $n + 11$.

Proof. Starting with the presentation,

$$\mathrm{MS}(n, w_k) = \langle x, y \mid x^{-1}y^n x = y^{n+1}, x = y^{-k}x^{-1}yxy \rangle,$$

the second relation may be equivalently expressed as

(i) $y^k x y^{-1} = x^{-1}yx$

(ii) $y^{-1}xy^k = xyx^{-1}$

Multiplying the first relation by $y^{-1}x$ from the left and by y^{-1} from the right, we get $y^{n-1}xy^{-1} = y^{-1}xy^n$, which we may rewrite as $y^{n-k-1}\left(y^k x y^{-1}\right) = \left(y^{-1}xy^k\right)y^{n-k}$. Substituting from (i) on the left-hand side and from (ii) on the right-hand side gives the required result. $\square$

Combining the previous three results shows that for each fixed n, $\mathrm{AK}(n)$ is AC-equivalent to all members of the infinite families $\mathrm{MS}(n, w_k)$ and $P(n, k)$. From this, Theorem 2.5 follows immediately by setting $k = n - 1$.

Remark

While $P(n, n-1)$ is always the shortest presentation in the P family, other $P(n, k)$ may also give reductions in length of $\mathrm{AK}(n)$. To determine which presentations in the P-family are shorter than $\mathrm{AK}(n)$, we compare the length of $\mathrm{AK}(n)$ (i.e., $2n + 7$) with the length of $P(n, k)$ given by:

$$\begin{cases} 2n + 10 - k, & \text{if } k \leq n - 1, \\ 3k + 12 - 2n, & \text{if } k \geq n. \end{cases}$$

This is less than $2n + 7$ when $4 \leq k \leq n - 1$ or $n \leq k < \frac{1}{3}(4n - 5)$.

We also note that the AC-triviality of AK(2), combined with Theorem 2.6, gives the following result.

Corollary 2.8

Each member of the infinite families $\mathrm{MS}(2, w_k)$ and $P(2, k)$ is AC-trivial.

2.3.3 Limitations of Greedy Search

Greedy search is based on the principle that it is better to explore shorter presentations rather than longer ones. While this leads to good performance overall, it also means that greedy search particularly struggles on presentations whose length must be increased substantially before it can be decreased, especially when there are many shorter-length presentations to check first. The fundamental limitation of greedy search is that all presentations it encounters must be stored in memory to avoid backtracking and to avoid revisiting presentations it has already seen. Eventually, the computer runs out of memory. Often this happens before the length has been able to increase by a large amount.

For example, if we run greedy search on AK(3) for 15M nodes, then it only ever reaches presentations of length 27, a length increase of 14. Table 2.1 shows the number of presentations that are visited before reaching the first presentation of a given length. This implies that in order to trivialize AK(3), the total length must reach at least 27, extending the lower bound of 20 given in [38]:

Proposition 2.9 (Length Increase for AK(3))

Either AK(3) is a counterexample to the Andrews–Curtis conjecture, or every trivialization sequence

$$\mathrm{AK}(3) = P_0 \;\to\; P_1 \;\to\; \cdots \;\to\; P_n \;\to\; \langle x, y \mid x, y\rangle$$

contains a presentation P_i with total relator length $|P_i| \geq 27$.

Table 2.1: Number of Presentations Visited Before Reaching a Presentation of a Given Length

Length	Number of Presentations Visited
15	1
16	26
17	40
18	1,000
19	2,128
20	19,604
21	38,612
22	136,720
23	246,856
24	855,209
25	1,532,733
26	5,431,781
27	10,118,637

One way to help greedy search trivialize presentations with large length increases is to use automorphisms of the underlying free group. Let P be a presentation and for an automorphism φ of the underlying free group, let $\varphi(P)$ refer to the resulting presentation. Then it is clear that if $P = P_0 \to P_1 \to \ldots \to P_n \to \langle x, y | x, y \rangle$ is a trivialization of P by substitution moves, $\varphi(P) = \varphi(P_0) \to \varphi(P_1) \to \ldots \to \varphi(P_n) \to \langle x, y | \varphi(x), \varphi(y) \rangle$ is also a path of substitution moves, and moreover $\langle x, y | \varphi(x), \varphi(y) \rangle$ is AC-trivial, as discussed in Section 2.1. However, the paths may require quite different length increases, and thus be of varying difficulty for greedy search. To give one example, the presentation $\mathrm{MS}(4, y^{-1}xyx^{-1}y^{-2})$ is easily trivialized by greedy search, requiring only 474 nodes and around a tenth of a second on a laptop. However, once we apply the automorphism $x \mapsto y^{-1}xy^2, y \mapsto xyx$, greedy search fails to trivialize the presentation with 1M nodes.

Notes

Greedy search with substitution moves was developed jointly with Michele Tarquini, and is based on greedy search from joint work with Ali Shehper, Anibal M. Medina-Mardones, Bartłomiej Lewandowski, Angus Gruen, Yang Qiu, Piotr Kucharski, Zhenghan Wang, and Sergei Gukov [42]. Many of the computational results presented here also originate from that joint work.

Chapter 3

The Stable Andrews–Curtis Conjecture

In this chapter, we discuss a version of the Andrews–Curtis conjecture, called the stable Andrews–Curtis conjecture, that allows for two additional moves, stabilization and its inverse. In Section 3.1 we introduce the additional moves, and in Section 3.2 we discuss the topological formulation of the conjecture and its connection to the Zeeman conjecture. Finally, in Section 3.3, we discuss the connection between the stable Andrews–Curtis conjecture and knot diagrams, which were thought to provide potential counterexamples distinguishing the stable and non-stable variants. Using the algebraic formulation of the conjecture, we show that such counterexamples cannot arise in certain cases.

3.1 Introduction

Definition 3.1 (Stable Andrews–Curtis Moves)**.** Let $\mathcal{P} = \langle x_1, \ldots, x_n \mid r_1, \ldots, r_n \rangle$ be a balanced group presentation. The *stable Andrews–Curtis moves* consist of the standard Andrews–Curtis moves (AC1)–(AC3) from the previous chapter supplemented by the following two moves:

(AC4) Add a new generator x_{n+1} and a new, trivial relator $r_{n+1} = x_{n+1}$. The new presentation is $\mathcal{P}' = \langle x_1, \ldots, x_{n+1} \mid r_1, \ldots, r_n, x_{n+1} \rangle$.

(AC5) The inverse of the stabilization move. If a presentation contains a relator r_i that is a single generator x_j, and this generator does not appear in any other relator, then both r_i and x_j can be removed.

Two presentations are said to be *stably Andrews–Curtis equivalent* if one can be obtained from the other by a finite sequence of moves (AC1)–(AC5).

A presentation is said to be *stably Andrews–Curtis trivial* if it is stably Andrews–Curtis equivalent to the trivial presentation.

Using this extended move set, we can state the stable version of the conjecture:

Conjecture 3.1 (Stable Andrews–Curtis Conjecture)
Every balanced presentation of the trivial group is stably Andrews–Curtis equivalent to the trivial presentation $\langle \mid \rangle$.

Substitution and Removal Move

We also present a move which is a combination of (AC1)–(AC5) and is useful in trivializing presentations:

Lemma 3.2 (Substitution and Removal Move)
Let $P = \langle x_1, \ldots, x_n, y \mid r_1, \ldots, r_n, y^{-1}w \rangle$ be a presentation of the trivial group, where w is a word in $x_1, \ldots, x_n$. Then $P' = \langle x_1, \ldots, x_n \mid r_1', \ldots, r_n' \rangle$ is stably AC-equivalent to P, where r_i' is r_i with all occurrences of y replaced by w.

Proof. First, we substitute w for y in $r_1, \ldots, r_n$, giving $\widetilde{P} = \langle x_1, \ldots, x_n, y \mid r_1', \ldots, r_n', y^{-1}w \rangle$. Since $r_1', \ldots, r_n'$ do not contain y, we can define the surjective homomorphism $\phi : \widetilde{P} \longrightarrow P'$ by $\phi(x_i) = x_i$ and $\phi(y) = w$. Since $\widetilde{P}$ is trivial, P' is trivial. Therefore, the normal closure of $r_1', \ldots, r_n'$ gives the free group generated by $x_1, \ldots, x_n$, which means w can be written as a product of conjugates of the r_i', namely, $w = w_1(r_{i_1}')^{\pm 1}w_1^{-1}w_2(r_{i_2}')^{\pm 1}w_2^{-1}\cdots w_m(r_{i_m}')^{\pm 1}w_m^{-1}$, for words w_i in the x_i and $i_k \in \{1, \ldots, n\}$. Note that m may be much larger than n. Now using (AC1)–(AC3) we can transform $y^{-1}w$ to $w_1(r_{i_1}')^{\pm 1}w_1^{-1}\cdots w_m(r_{i_m}')^{\pm 1}w_m^{-1}w^{-1}y$ which equals y, and then we can remove y by (AC5). $\square$

Remark

At first glance, this move seems quite similar to the substitution move from the previous chapter. However, in the case of substitution and removal, it is nontrivial in general to write down the sequence of (AC1)–(AC5) moves to realize such a move. Also note that a single application of the substitution and removal move may require arbitrarily many (AC1)–(AC5) moves, and we are not aware of any general bounds. When considering trivializing presentations through explicit sequences of moves, this move has the advantage of simultaneously applying many of the moves (AC1)–(AC5).

The Power of the Stabilization Move

For the two families of potential counterexamples mentioned in the previous chapter—the presentations AK(n) and MS(n, w)—all such presentations that are potential counterexamples to the Andrews–Curtis conjecture are also potential counterexamples to the stable Andrews–Curtis conjecture. Indeed, in the context of balanced presentations of the trivial group, we are not aware of any evidence that AC-triviality is stronger than stable AC-triviality.

However, these moves can be equivalently applied to any group presentation, and outside the context of balanced presentations of the trivial group, it is known that (AC1)–(AC5) generates a larger equivalence class than just (AC1)–(AC3). Indeed, in [4], Barmak notes that the first examples of such presentations are likely due to Zieschang in [57]: $\langle x, y \mid x^3y^5\rangle$ and $\langle x, y \mid x^3y^3x^3y^2\rangle$.

It is easy to see that these presentations are stably AC-equivalent. We can do a substitution and removal in reverse to get

$$\langle x, y \mid x^3y^3x^3y^2\rangle \to \langle x, y, z \mid zyz, z = x^3y^2\rangle,$$

and then simply do a substitution and removal for $y = z^{-2}$, giving

$$\langle x, y \mid x^3y^3x^3y^2\rangle \to \langle x, y, z \mid zyz, z = x^3y^2\rangle \to \langle x, z \mid z = x^3z^{-4}\rangle = \langle x, z \mid z^5 = x^3\rangle$$

(note we can swap the sign on the power of z by doing one additional substitution and removal to match the first original presentation on the nose). As Barmak explains in [4], these presentations are not AC-equivalent merely by inspection, as the move (AC2) is impossible with a single relator.

This idea was generalized by McCool and Pietrowski in [32], where they showed the same for the presentations $\langle x, y \mid x^k y^{pt+1}\rangle$ and $\langle x, y \mid (x^k y^t)^p y\rangle$ for $k, p, t \geq 2$. These are stably AC-equivalent by the same trick, namely, setting $z = x^k y^t$ through an inverse substitution and removal and then doing a substitution and removal for y.

There are also examples with more than one relator: Metzler shows in [33] that the presentations $\langle x, y \mid x^5, y^5, [x, y]\rangle$ and $\langle x, y \mid x^5, y^5, [x^2, y]\rangle$ are stably AC-equivalent but not AC-equivalent.

Finally, there exist examples of balanced presentations which are stably AC-equivalent but not AC-equivalent. Indeed, in [4], Barmak introduces two presentations of $\mathbb{Z} \times \mathbb{Z}$ which are shown to not be AC-equivalent:

$$\langle x, y \mid [x, y], 1\rangle \quad \text{and} \quad \left\langle x, y \,\middle|\, \left[x, \left[x, y^{-1}\right]\right]^2 y \left[y^{-1}, x\right] y^{-1}, \left[x, \left[\left[y^{-1}, x\right], x\right]\right] \right\rangle.$$

In [15], these two presentations are shown to be stably AC-equivalent.

3.2 Topological Version

The stable Andrews–Curtis conjecture has an equivalent topological formulation, first pointed out by a referee of the original paper by Andrews and Curtis [2]. It is based on Whitehead's simple homotopy theory [51] [50].

3.2.1 Simple Homotopy Theory

We recall some notions from simple homotopy theory in the setting of polyhedra.

Definition 3.2 (Elementary simplicial collapse)**.** Let K be a simplicial complex. Suppose that $\sigma^n, \delta^{n-1} \in K$ are open simplices such that:

1. σ is *principal*, i.e., it is not a proper face of any other simplex in K;
2. δ is a *free face* of σ, i.e., δ is not a proper face of any simplex in K other than σ.

Then the transition from K to $K \setminus (\sigma \cup \delta)$ is called an *elementary simplicial collapse.*

If a simplicial complex K can be transformed to a subcomplex L through a sequence of elementary simplicial collapses, we say K *collapses to* L and write $K \searrow L$.

Definition 3.3 (Elementary polyhedral collapse)**.** Let P be a polyhedron. Suppose that $P = Q \cup B^n$, where B^n is an n-cell and $P \cap B^n = B^{n-1}$ is an $(n-1)$-dimensional face of B^n. Then the transition from P to Q is called an *elementary polyhedral collapse.*

If a polyhedron K can be transformed to a subpolyhedron L through a sequence of elementary polyhedral collapses, we say K *collapses to* L and write $K \searrow L$.

We note that a simplicial collapse is a special case of a polyhedral collapse, and conversely, it is possible to choose a triangulation of the ball B^n such that the collapse onto B^{n-1} can be realized as a simplicial collapse.

Elementary expansions are defined as the inverse operations of elementary collapses.

Definition 3.4 (Collapsible)**.** If a polyhedron K collapses to a point, we say K is *collapsible.*

Certainly, collapsibility implies contractibility, but the converse is not true. Two common examples are the dunce cap, where a 2-cell is attached to a circle x with attaching map xxx^{-1}, and the house with two rooms. In each of these cases, there is no free face to start collapsing. This is the obstruction to collapsibility.

Definition 3.5 (Simple homotopy equivalence)**.** Let X and Y be polyhedra. A continuous map $f : X \to Y$ is called a *simple homotopy equivalence* if there exists a finite sequence of polyhedra

$$X = X_0,\, X_1,\, X_2,\, \ldots,\, X_m = Y$$

such that for each $i = 0, \ldots, m-1$, either X_{i+1} is obtained from X_i by an elementary polyhedral collapse or elementary polyhedral expansion.

If such a finite sequence exists, we say that X and Y are *simple-homotopy equivalent*, and we write

$$X \simeq_s Y.$$

Every simple homotopy equivalence is a homotopy equivalence, but not conversely. The obstruction $\tau(f)$ to a homotopy equivalence $f : X \to Y$ being a simple homotopy equivalence is called the *Whitehead torsion*, which takes values in an abelian group $\mathrm{Wh}(\pi)$ called the *Whitehead group*, which depends on $\pi = \pi_1(X) = \pi_1(Y)$.

Definition 3.6 (m-deformation equivalence)**.** Let $m \geq 0$ be an integer. Two polyhedra X and Y are said to be *m-deformation equivalent* if there exists a finite sequence

$$X = X_0,\, X_1,\, \ldots,\, X_k = Y$$

such that each step $X_i \leftrightarrow X_{i+1}$ is either an elementary expansion or an elementary collapse involving only cells of dimension at most m.

In particular, if such a sequence exists with $m = 3$, we say that X and Y are *3-deformation equivalent*.

We note that, in dimension n for $n > 2$, all simple homotopy equivalences can be realized as $(n+1)$-deformation equivalences:

Theorem 3.3 ([49])
If $m > 2$, then for any two simple homotopy equivalent polyhedra X, Y of dimension $\leq m$, there is a sequence of elementary collapses and expansions of dimension at most $m+1$ that takes X to Y.

In particular, this theorem implies that simple homotopy equivalent 2-polyhedra are 4-deformation equivalent.

3.2.2 Topological Version of stable Andrews–Curtis Conjecture

The topological version of the stable Andrews–Curtis conjecture asks whether the result from Theorem 3.3 can be generalized to dimension 2 in the case of contractible polyhedra, i.e., whether contractible, simple homotopy equivalent 2-polyhedra are 3-deformation equivalent.

The connection between the stable Andrews–Curtis conjecture and topology is that, in short, 3-deformation equivalence classes of 2-complexes are a topological analog to stable AC-equivalence classes of group presentations. On the one hand, there is a natural way to build a 2-complex K_P from any presentation $P = \langle x_1, \ldots, x_n \mid r_1, \ldots, r_m \rangle$: we take a single vertex, add n edges corresponding to each generator, and m 2-cells glued according to the relations. Conversely, for any 2-complex K, we can pick a maximal tree T of the 1-skeleton and collapse it, which leaves a 2-complex K/T with a single vertex. From here, we can read off a presentation of its fundamental group. We note that K/T is 3-deformation equivalent to K, and that different choices of T lead to presentations that are stably AC-equivalent.

That these different move sets correspond is nontrivial, and a proof of the equivalence can be found in [53] and [20]. Using this equivalence, we can state the topological version of the Andrews–Curtis conjecture.

Conjecture 3.4 (Stable Andrews–Curtis Conjecture, Topological Formulation)
Any (finite) contractible 2-dimensional polyhedron K can be 3-deformed to a point.

3.2.3 Connections to Zeeman Conjecture

The topological version of the stable Andrews–Curtis conjecture is closely related to another important topological conjecture called the Zeeman conjecture.

Conjecture 3.5 (Zeeman Conjecture)
If K is a contractible 2-dimensional polyhedron, then $K \times I$ is collapsible.

There exists some evidence for and against the conjecture, due to M. M. Cohen.

Theorem 3.6 (M. Cohen, [9])
If K is a contractible 2-polyhedron, then $K \times I^6$ is collapsible. In general, if K is a contractible n-polyhedron for $n \geq 3$, then $K \times I^{2n}$ is collapsible.

Theorem 3.7 (M. Cohen, [10], Theorem 3)
For all $n \geq 3$ there exists a contractible n-polyhedron K such that $K \times I$ is not collapsible.

Importantly, we observe that the Zeeman conjecture implies the stable Andrews–Curtis conjecture, since we can take a contractible 2-polyhedron K, cross it with an interval, and then apply the conjecture to collapse it to a point.

Proposition 3.8
The Zeeman conjecture implies the stable Andrews–Curtis conjecture.

Proof. Any 2-polyhedron K expands to $K \times I$. If K is contractible, then by the Zeeman conjecture, $K \times I$ collapses to a point. □

We also note that the Zeeman conjecture is related to the famous Poincaré conjecture, which was proven by Perelman.

Theorem 3.9 (Poincaré Conjecture)
If S is a 3-dimensional PL manifold homotopy equivalent to S^3, then it is PL-homeomorphic to S^3.

If we remove an open ball, we can equivalently reformulate the conjecture as: *any compact contractible 3-dimensional PL manifold is PL-homeomorphic to the 3-ball.*

In particular, the Zeeman conjecture also implies the Poincaré conjecture, first observed by Zeeman himself.

Proposition 3.10 ([55], Theorem 2)
The Zeeman conjecture implies the Poincaré conjecture.

Proof. Let M be a compact contractible 3-dimensional PL-manifold. There exists a spine X of M of dimension at most 2. Since M collapses to X, we have that $M \times I$ collapses to $X \times I$, which collapses to a point by the Zeeman conjecture. Thus, $M \times I$ is homeomorphic to a regular neighborhood of a point, i.e., a 4-ball. We note that M, as $M \times \{0\}$, lies in the S^3 boundary of $M \times I$, and the boundary of M is S^2. Thus, by the Alexander Theorem, which says that every 2-sphere in S^3 bounds a 3-ball, M is a 3-ball. □

The connection between these three conjectures is even stronger than mentioned here once the Zeeman conjecture is restricted to fake surfaces. Fake surfaces are introduced in Chapter 4, and the connections between the Zeeman conjecture restricted to fake surfaces, the stable Andrews–Curtis conjecture, and the Poincaré conjecture are discussed in Section 4.4.

3.3 Unknot Diagrams and Stably AC-Trivial Presentations

In [36], Myasnikov, Myasnikov, and Shpilrain discuss a way to produce stably AC-trivial presentations from diagrams of the unknot. These presentations are not known to be AC-trivial, and thus serve as potential counterexamples to the AC conjecture and to the equivalence of the conjectures. In this section, we discuss results related to both uses: using unknot diagrams to prove the stable AC-triviality of presentations that elude computational trivialization attempts, and the possibility of producing new potential counterexamples to the Andrews–Curtis conjecture.

3.3.1 Stably AC-Trivial Presentations From Wirtinger Presentations

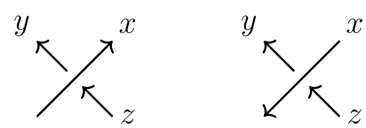
We recall the construction of the Wirtinger presentation of a knot diagram. Given any oriented knot diagram with n crossings, we assign generators $\{x_i\}_{i=1}^n$ to each of the arcs, and each crossing gives us a relator $\{r_i\}_{i=1}^n$. In our notation, the left crossing in Figure 3.1 corresponds to the relator $y = x^{-1}zx$, and the right one gives $y = xzx^{-1}$. These generators and relators form a balanced presentation of the fundamental group of the knot complement. Moreover, there exists an ordering of the relators and a choice of ± 1 as exponents such that the relators satisfy the equation

$$\prod_{i=1}^{n} r_i^{\pm 1} = 1.$$

Thus, any relator r_k can be eliminated by expressing it as a product of the other relators, giving

$$\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{k-1}, r_{k+1}, \ldots, r_n \rangle$$

without changing the underlying group.



$$y = x^{-1}zx \qquad y = xzx^{-1}$$

Figure 3.1: Types of Crossings

When the knot diagram is of the unknot, the fundamental group of the knot complement is $\mathbb{Z}$, which is generated by each generator $x_1, \ldots, x_n$. If we add a relator w that is a word in $x_1, \ldots, x_n$ with exponent sum ± 1, we get a balanced presentation $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{k-1}, r_{k+1}, \ldots, r_n, w \rangle$ of the trivial group. The result below establishes that every such balanced presentation is stably AC-trivial.

Proposition 3.11 (Myasnikov, Myasnikov, and Shpilrain, [36])
Let $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_n \rangle$ be the Wirtinger presentation of an unknot diagram. Then for any $k = 1, \ldots, n$ and word w in the x_i with exponent sum ± 1, the balanced presentation of the trivial group $\langle x_1, \ldots, x_n \,|\, r_1, \ldots, r_{k-1}, r_{k+1}, \ldots, r_n, w \rangle$ is stably Andrews–Curtis-trivial.

The proof of this proposition relies on the well-known realization of Reidemeister moves for knot diagrams as stable AC-moves for the associated balanced presentation of the trivial group, first studied in [48].

Proof. As a knot diagram of the unknot is reduced to the trivial diagram, the associated balanced presentation is shown to be stably AC-trivial.

Two diagrams of the same knot can be related by a finite sequence of the three Reidemeister moves shown in Figure 3.2. As a diagram of the unknot is reduced to the trivial diagram, the associated balanced presentation is reduced to the trivial presentation.

Here we distinguish **R2a** and **R2b** because they give slightly different computations within the presentations. Since we want to consider the behavior of balanced presentations $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{k-1}, r_{k+1}, \ldots, r_n, w \rangle$ of the trivial group under Reidemeister

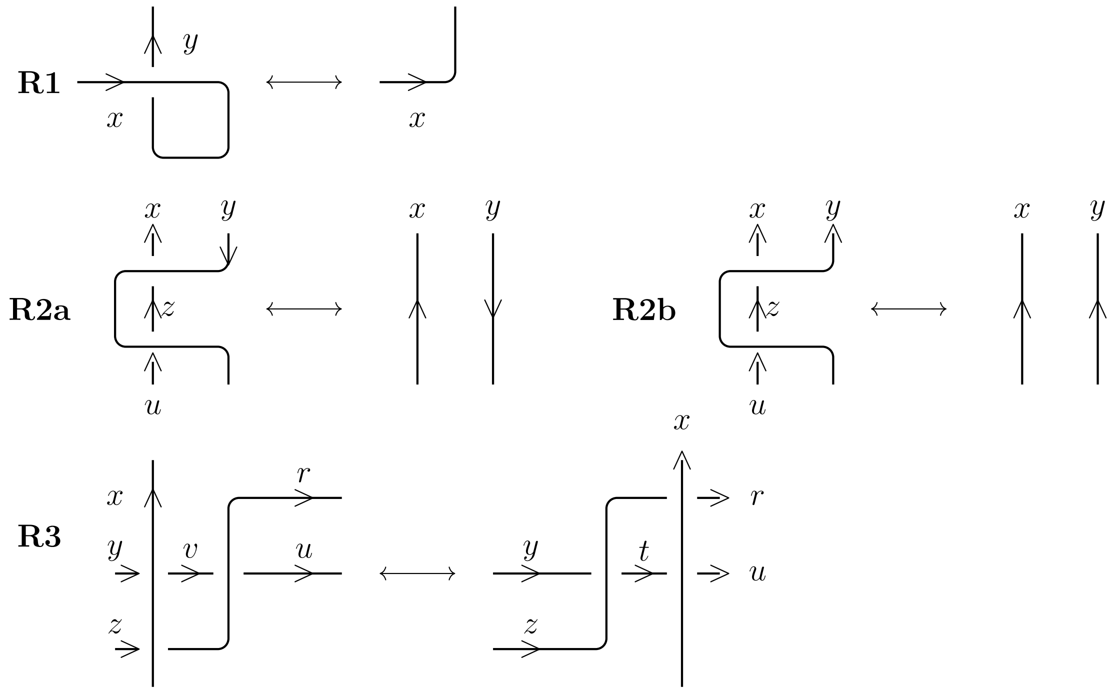


Figure 3.2: The 3 Reidemeister Moves

moves, we can assume that the relator eliminated, r_k, is not one given by the crossings in **R1**, **R2a**, **R2b**, or **R3**. We note, however, that since any relator in a Wirtinger presentation can be written as the product of the other relators, the relator we eliminated can be recovered using (AC1)–(AC3).

R1:

$$\langle x_1, \ldots, x_n, x, y \mid r_1, \ldots, r_n, yx^{-1}, w\rangle \longleftrightarrow \langle x_1, \ldots, x_n, x \mid r_1', \ldots, r_n', w'\rangle$$

Here r_i' and w' are obtained from replacing y in r_i and w by x. The equivalence comes from the substitution and removal in Lemma 3.2.

R2a:

$$
\begin{aligned}
&\langle x_1,\dots,x_n,x,y,z,u \mid r_1,\dots,r_{n+1}, xyz^{-1}y^{-1}, zy^{-1}u^{-1}y, w\rangle \\
&\longleftrightarrow \langle x_1,\dots,x_n,x,y,u \mid r'_1,\dots,r'_{n+1}, ux^{-1}, w'\rangle \\
&= \langle x_1,\dots,x_n,x,y,u \mid r_1,\dots,r_{n+1}, ux^{-1}, w'\rangle \\
&\longleftrightarrow \langle x_1,\dots,x_n,x,y \mid \tilde{r}_1,\dots,\tilde{r}_{n+1}, \tilde{w}\rangle
\end{aligned}
$$

Here, r'_i and w' are obtained from replacing z in r_i and w by $y^{-1}xy$; $\tilde{r}_i$ is obtained from replacing u in r_i by x; and $\tilde{w}$ is obtained from replacing u in w' by x. The second equality comes from the fact that the r_i do not contain z.

R2b is similar to **R2a**.

R3:

$$
\begin{aligned}
&\langle x_1,\dots,x_n,x,y,z,u,v,r \mid r_1,\dots,r_{n+2}, vxy^{-1}x^{-1}, urv^{-1}r^{-1}, rxz^{-1}x^{-1}, w\rangle \\
&\longleftrightarrow \langle x_1,\dots,x_n,x,y,z,u,r \mid r_1,\dots,r_{n+2}, urxy^{-1}x^{-1}r^{-1}, rxz^{-1}x^{-1}, w'\rangle \\
&\longleftrightarrow \langle x_1,\dots,x_n,x,y,z,u,r \mid r_1,\dots,r_{n+2}, uxzy^{-1}z^{-1}x^{-1}, rxz^{-1}x^{-1}, w'\rangle \\
&\longleftrightarrow \langle x_1,\dots,x_n,x,y,z,u,r,t \mid r_1,\dots,r_{n+2}, uxzy^{-1}z^{-1}x^{-1}, rxz^{-1}x^{-1}, tzy^{-1}z^{-1}, w'\rangle \\
&\longleftrightarrow \langle x_1,\dots,x_n,x,y,z,u,r,t \mid r_1,\dots,r_{n+2}, rxz^{-1}x^{-1}, tzy^{-1}z^{-1}, uxt^{-1}x^{-1}, w'\rangle
\end{aligned}
$$

Here, w' is obtained from replacing v in w by xyx^{-1}.

After reducing our diagram to the trivial diagram of the unknot, we are left with $\langle x \mid \bar{w}\rangle$, where $\bar{w}$ is w after applying all the Reidemeister moves. This is because the trivial diagram of the unknot gives the presentation $\langle x \mid \,\rangle$ of the infinite cyclic group, so $\bar{w}$ is the only relator remaining. Moreover, $\bar{w} = x^{\pm 1}$, since at each stage w has exponent sum ± 1, so $\langle x \mid \bar{w}\rangle$ can be reduced to the trivial presentation $\langle \mid \rangle$. This proves Proposition 3.11. □

3.3.2 Balanced Presentations with Fewer Generators

Starting with a balanced presentation coming from Proposition 3.11, one may recursively apply stable AC-moves to reduce the number of generators and relators. The resultant

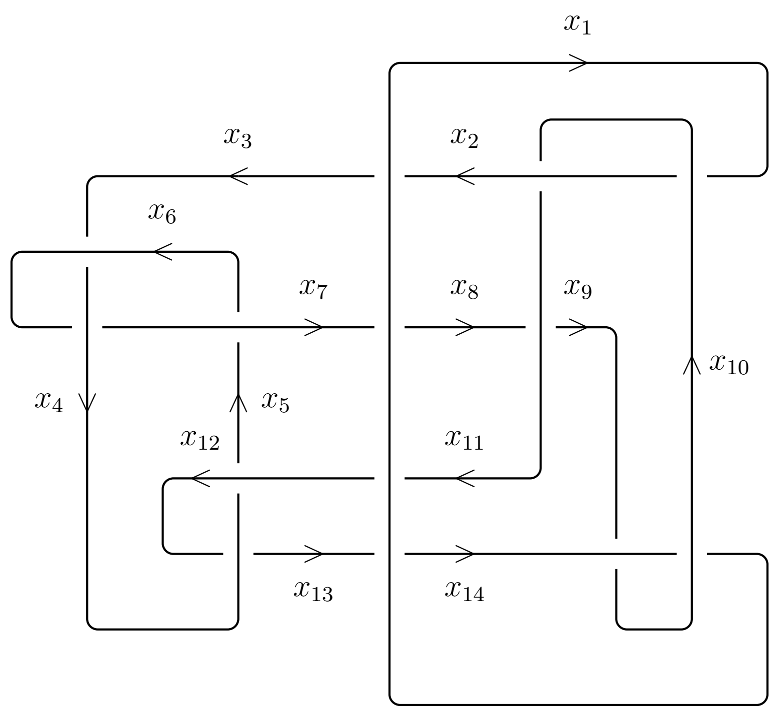


Figure 3.3: A Diagram of the Unknot

presentations are all stably AC-trivial by construction, but they could, a priori, serve as potential counterexamples to the standard Andrews–Curtis conjecture [36]. However, we will show in the next section that a large subset of these are, in fact, AC-trivial.

As an example of this procedure, consider the unknot diagram shown in Figure 3.3. The associated Wirtinger presentation is as follows,

$$
\begin{aligned}
W = \langle x_1, x_2, \ldots, x_{13}, x_{14} \mid\, & x_1 = x_{10}x_{14}x_{10}^{-1}, x_2 = x_{10}^{-1}x_1x_{10}, \\
& x_3 = x_1^{-1}x_2x_1, x_4 = x_6^{-1}x_3x_6, \\
& x_5 = x_{12}x_4x_{12}^{-1}, x_6 = x_7^{-1}x_5x_7, \\
& x_7 = x_4^{-1}x_6x_4, x_8 = x_1x_7x_1^{-1}, \\
& x_9 = x_{11}^{-1}x_8x_{11}, x_{10} = x_{14}x_9x_{14}^{-1} \\
& x_{11} = x_2^{-1}x_{10}x_2, x_{12} = x_1^{-1}x_{11}x_1 \\
& x_{13} = x_4x_{12}x_4^{-1}, x_{14} = x_1x_{13}x_1^{-1} \,\rangle\,.
\end{aligned}
$$

Using stable AC-moves, we can reduce the number of generators in W to obtain several 3-generator families of presentations. Consider, for example, deleting the relator r_6, adding a word w, and eliminating generators in the following order through Lemma 3.2: x_1, x_2, x_3, x_4, x_5, x_7, x_8, x_9, x_{11}, x_{12}, and x_{13}. We are left with a presentation of three generators x_6, x_{10}, and x_{14}. Renaming $x = x_{10}$, $y = x_{14}$, and $z = x_6$, the resultant presentation equals

$$\begin{aligned}\langle x, y, z \mid &[y^{-1}, x^{-1}]yx^{-1}z^{-1}xy^{-1}x^{-1}[y^{-1}, x]zx[y^{-1}, x^{-1}]yx^{-1}zxy^{-1}x^{-1},\\ &[y^{-1}, x]z^{-1}x[y^{-1}, x^{-1}]yx^{-1}zxy^{-1}[x^{-1}, y^{-1}]xyx^{-1}z^{-1}xy^{-1}[x^{-1}, y^{-1}]x^{-1}zxy^{-1}x^{-1},\\ &w\rangle.\end{aligned}$$

After choosing an appropriate w, we may eliminate one more generator, thus obtaining two-generator presentations through this method.

Remark

Note, however, that if we were to obtain a 2-generator family without picking a word w (and merely through recursive use of stable AC-moves), we would obtain an infinite family of the form $\langle x, y \mid r_1, w\rangle$. Members of this family are all AC-trivial through the following argument. If $\langle x, y \mid r_1, w\rangle$ is a presentation coming from an unknot diagram, then $\langle x, y \mid r_1\rangle$ is a presentation of $\mathbb{Z}$ with $r_1 = xy^{\pm 1}$, and any presentation of the form $\langle x, y \mid xy^{\pm 1}, w\rangle$ is AC-trivializable by first substituting for y and then removing it.

3.3.3 Knot Diagrams give AC-trivial Presentations

In this section, we will prove that every presentation of the form

$$\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{k-1}, r_{k+1}, \ldots, r_n, w\rangle$$

as in Proposition 3.11 is in fact AC-trivial. Moreover, we will show that presentations with fewer generators that come from recursively applying substitution moves from Lemma 3.2 are also AC-trivial.

The proofs of AC-triviality rely on the following lemma, which establishes that presentations with different choices of w are all AC-equivalent.

Lemma 3.12

Let $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{n-1}\rangle$ be a presentation of $\mathbb{Z}$. Then there exists one word x in the

x_i such that for any word w that makes the group trivial, the balanced presentation of the trivial group $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{n-1}, w\rangle$ is AC-equivalent to $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{n-1}, x\rangle$.

Proof. Since $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{n-1}\rangle$ is a presentation of $\mathbb{Z}$, there exists one word x that generates $\mathbb{Z}$. Then any x_i can be expressed as the product of x^{n_i} with a product of conjugates of $r_1, \ldots, r_{n-1}$ for some $n_i \in \mathbb{Z}$.

Suppose $w = x_{i_1}^{\epsilon_1} x_{i_2}^{\epsilon_2} \cdots x_{i_m}^{\epsilon_m}$ for $i_k \in \{1, \ldots, n\}$ and $\epsilon_k \in \mathbb{Z}$. For each x_i in w, multiply w by the corresponding product of conjugates of $r_1, \ldots, r_{n-1}$ to replace x_i by x^{n_i}. Then w is changed to x^p for some p. Since x generates $\mathbb{Z}$, we have $p = \pm 1$. $\square$

Note that the lemma holds for *any* presentation $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{n-1}\rangle$ of $\mathbb{Z}$, and not just the ones coming from knot diagrams of the unknot. We will now use this lemma to prove our result for presentations of Proposition 3.11.

Theorem 3.13

Let $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_n\rangle$ be the Wirtinger presentation of an unknot diagram. Then for any $k = 1, \ldots, n$ and word w in the x_i with exponent sum ± 1, the balanced presentation of the trivial group $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{k-1}, r_{k+1}, \ldots, r_n, w\rangle$ is Andrews–Curtis-trivial.

Proof. The relation

$$\prod_{i=1}^{n} r_i^{\pm 1} = 1$$

for a Wirtinger presentation of a knot diagram allows us to write any one relator r_i as a product of other relators and their inverses. Hence, the presentations

$$\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{k-1}, r_{k+1}, \ldots, r_n, w\rangle$$

with different choices of $k \in \{1, \ldots, n\}$ and fixed w are all AC-equivalent. Without loss of generality, we set $k = n$. As the presentation comes from a Wirtinger presentation, we can write it as

$$\left\langle x_1, \ldots, x_n \mid x_1 = z_2 x_2 z_2^{-1}, x_2 = z_3 x_3 z_3^{-1}, \ldots, x_{n-1} = z_n x_n z_n^{-1}, w \right\rangle,$$

where $z_i \in \{x_1^{\pm 1}, \ldots, x_n^{\pm 1}\}$.

Using Lemma 3.12, we choose $w = x_n$. With the second-last relator $x_{n-1} = z_{n-1} x_n z_{n-1}^{-1}$, substituting $x_n = 1$ simplifies it to $x_{n-1} = 1$. Substituting $x_{n-1} = 1$ into the preceding relator, we proceed iteratively, simplifying each relation until the first relation reads $x_1 = 1$. Thus, the presentation reduces to $\langle x_1, \ldots, x_n \mid x_1, \ldots, x_n \rangle$, rendering it AC-trivial. □

Remark

Starting with a balanced presentation of Theorem 3.13 and reducing the number of generators only through recursive application of substitution and removal (Lemma 3.2), we obtain a large class of AC-trivial presentations with fewer generators. Indeed, each relator in any such presentation is still a simple conjugation relation ($x_i = z x_j z^{-1}$, where z is some word in the generators). Thus, we can use the same argument as in the proof of Theorem 3.13 to trivialize the relators iteratively.

We also note that for any AC-trivial presentation, we can apply any moves (AC1)–(AC4) without changing the AC-triviality of the presentation.

For example, consider the 3-generator family presented in the previous subsection, which was obtained through repeated application of Lemma 3.2. The first relator is x equals z conjugated by $yxyx^{-1}z^{-1}xy^{-1}x^{-1}y^{-1}xyx^{-1}$, and the second relator is y equals x conjugated by $xyx^{-1}z^{-1}xy^{-1}x^{-1}yxyx^{-1}zxy^{-1}x^{-1}y^{-1}$. Substituting $w = z$ simplifies the first relator to x, which when substituted into the second relator, simplifies it to y.

We note, however, that the method described in the above remark is only one way to reduce the number of generators. It is also possible to consider more general sequences of stable AC-moves, which may produce presentations where the relators are not all in simple conjugation form. It would be interesting to prove or disprove that any such presentation is also AC-trivial.

Conjecture 3.14

Let $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{k-1}, r_{k+1}, \ldots, r_n, w \rangle$ be the AC-trivial presentation coming from an unknot diagram as in Theorem 3.13. Then any stably AC-equivalent presentation $\langle x_1, \ldots, x_m \mid s_1, \ldots, s_{m-1}, w' \rangle$ is also AC-trivial.

Moreover, Wirtinger presentations of unknot diagrams are specific cases of presentations of $\mathbb{Z}$ where each generator generates the group. It is also interesting to consider whether any such presentation of $\mathbb{Z}$ leads to AC-trivial presentations.

Conjecture 3.15

Let $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{n-1} \rangle$ be any presentation of $\mathbb{Z}$ where each x_i generates $\mathbb{Z}$, i.e., $x_i = x_j^{\pm 1}$ for any i, j. Then for any word w in the x_i with exponent sum ± 1, the presentation $\langle x_1, \ldots, x_n \mid r_1, \ldots, r_{n-1}, w \rangle$ is AC-trivial.

Notes

The results about unknot diagrams are from joint work with Ali Shehper, Anibal M. Medina-Mardones, Bartłomiej Lewandowski, Angus Gruen, Yang Qiu, Piotr Kucharski, Zhenghan Wang, and Sergei Gukov [42].

Chapter 4

Fake Surfaces

In this chapter, we discuss two-dimensional generic polyhedra, also known as fake surfaces. In Section 4.1, we outline the importance of fake surfaces in mathematics. Sections 4.2 and 4.3 provide definitions and basic properties of fake surfaces. In Section 4.4, we describe their connection to the stable Andrews–Curtis and Zeeman conjectures. Section 4.5 introduces an induction scheme aimed at proving new cases of these conjectures. In Section 4.6, we examine the fake surfaces associated with the AK(n) family of potential counterexamples and give an explicit construction in the case of AK(3). Finally, Section 4.7 develops computational techniques building on Sections 4.5 and 4.6, with the aim of establishing additional cases of the stable Andrews–Curtis and Zeeman conjectures.

4.1 Introduction

Generic polyhedra are interesting objects to study in their own right. Outside of mathematics, two-dimensional generic polyhedra, also called fake surfaces, appear in nature, for instance in soap films [44]. Within mathematics, they have been used to study 3-manifolds (e.g., [8] and [31]) and corresponding quantum invariants [47]. When decorated with gleams, they are also used to study 4-manifolds in Turaev's shadow theory [46, 11]. In particular, within 4-manifold topology, they are used in the classification of 4-manifolds with shadow complexity 0 and 1 [29, 22], in the context of Mazur manifolds [19, 23], and in capped grope theory [16].

We study fake surfaces as interesting topological objects in their own right that generalize surfaces, with applications to the stable Andrews–Curtis and Zeeman conjectures [30].

4.2 Definitions

Definition 4.1 (Polyhedron)**.** A *polyhedron* P is a subset of $\mathbb{R}^n$ for some n such that for all $x \in P$, there exists a neighborhood C_x of x such that $C_x = xL$, where L is a compact subset of $\mathbb{R}^n$. We say that P is *closed* if it has no boundary, i.e., no closed free faces.[1]

Let Δ^n denote the standard n-simplex with $n+1$ vertices, and let Δ^n_k denote its k-skeleton for $0 \le k \le n$.

Definition 4.2. Let $\Pi^n_k = \mathrm{Cone}(\Delta^{k+1}_{k-1}) \times I^{n-k}$ for $0 \le k \le n$, that is, the thickened cone on the codimension 2 skeleton of the $(k+1)$-simplex Δ^{k+1}. When $k = 0$, Π^n_0 is simply the n-cube I^n.

Definition 4.3 (Generic polyhedron)**.** A *generic n-polyhedron* P is a polyhedron where each point $x \in P$ has a neighborhood of the form Π^n_k for some $0 \le k \le n$.

We call points with neighborhoods Π^n_0 *manifold points*, and points with neighborhoods Π^n_k for $1 \le k \le n$ *singularities of type k*.

We define an *n-singular manifold M* to be a generic n-polyhedron up to PL isomorphism.

In this terminology, an n-singular manifold has n types of singularities in addition to the manifold points.

[1]Here we use "closed" as defined in [21].

Example

A generic one-polyhedron is a polyhedron where each point has a neighborhood that is an interval or the cone on three points. In other words, generic one-polyhedra are trivalent graphs. The points of the second type—the vertices—are the singular points.

Definition 4.4 (Fake surface)**.** A *fake surface* is a compact 2-singular manifold. The *complexity* of a fake surface is defined as the number of type 2 singularities.

The two types of singular points in a fake surface are shown in Figure 4.1.

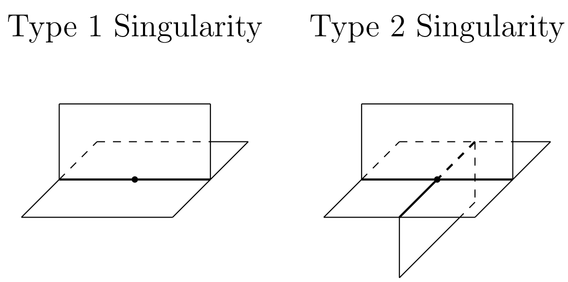


Figure 4.1: Two Types of Singularities in a Fake Surface

Every fake surface has an intrinsic stratification: the two-dimensional stratum consists of manifold points, the one-dimensional stratum consists of singularities of type 1, and the zero-dimensional stratum consists of singularities of type 2. The number of type 2 singularities of a fake surface is finite.

Note that the type 1 singularities form a 4-regular multigraph, which is not necessarily connected. Moreover, a fake surface without any singularities is a surface in the usual sense.

Two examples of fake surfaces are given in Figure 4.2. Their 1-skeleta are outlined in black, and given explicitly in Figure 4.3.

Definition 4.5 (Cellular and Planar Fake Surface)**.** A fake surface is called *cellular* if the intrinsic stratification is a CW complex, i.e., all connected open components of the intrinsic strata are disks.

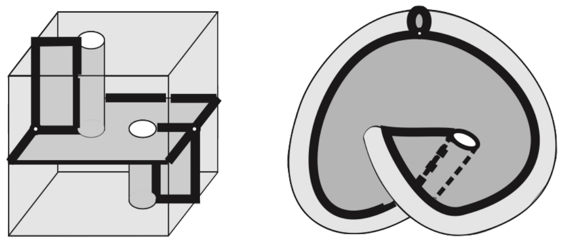


Figure 4.2: Two Examples of Fake Surfaces

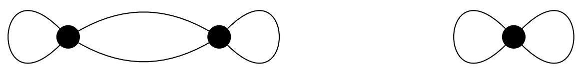


Figure 4.3: 1-Skeleta for Example Fake Surfaces

A fake surface is called *planar* if every edge has at least one type 2 singularity, and every open 2-cell is planar, i.e. of genus 0.

Note that a planar fake surface is not necessarily cellular, as there could be annuli in the 2-skeleton.

We recall two useful facts about fake surfaces:

Fact

A contractible fake surface is planar.

Fact

Every cellular fake surface has a connected intrinsic 1-skeleton.

Henceforth, fake surfaces are assumed to be cellular.

4.3 Fake Surface Spines and Non-Spines

4.3.1 Fake Surface Spines

Definition 4.6 (Spine)**.** A subpolyhedron P of a closed connected 3-manifold M is called a *spine* of M if $M \setminus \operatorname{Int} B^3$ collapses to P. A subpolyhedron P of a 3-manifold with boundary is called a *spine* of M if M collapses to P.

In general, all 3-manifolds contain a spine of dimension ≤ 2. Indeed, as long as we are in dimension 3, there will be a free face to continue collapsing.

A spine carries a lot of information about the corresponding 3-manifold, but does not uniquely determine it. For example, both the annulus $S^1 \times I$ and the Möbius band have their central circles as spines.

However, limiting the scope to spines that are fake surfaces allows us to resolve this issue.

Theorem 4.1 (B. Casler, [8])
Every closed 3-manifold has a fake surface as a spine.

Theorem 4.2 (S. Matveev, [30])
If two closed 3-manifolds have homeomorphic fake surface spines, the manifolds themselves are homeomorphic.

Therefore, we can present 3-manifolds by their fake surface spines. For example, the two fake surfaces in Figure 4.2 are spines of the 3-ball.

In general, every n-manifold has a generic $(n-1)$-dimensional polyhedron as a spine. However, in higher dimensions, *every* generic $(n-1)$-dimensional polyhedron is a spine of some n-manifold, much like in group theory where every presentation determines a group. The case of fake surfaces and 3-manifolds is especially interesting because not all fake surfaces appear as spines of 3-manifolds. Indeed, for low complexity, most fake surfaces are not spines of 3-manifolds (see Table 5.3 and Conjecture 5.2). In other words, even though we can present 3-manifolds using fake surfaces, some fake surfaces do not present 3-manifolds. To detect which fake surfaces are spines of 3-manifolds, we introduce the notion of a T-bundle of a 2-cell.

4.3.2 T-bundles of Fake Surfaces

Along the boundary of each 2-cell, a T-bundle can be constructed, as shown in [30]. The T-bundle can be trivial, consisting of a disk attached along the middle circle of the annulus $S^1 \times I$ (which we denote $A \cup D$), or nontrivial, consisting of a disk attached to the middle circle of a Möbius band (which we denote $M \cup D$).

Definition 4.7 (Trivial and Nontrivial T-Bundles)**.** The boundary curve of a 2-cell of a fake surface has a *trivial* or *nontrivial* T-bundle if its characteristic map $f : D^2 \to P$ can be extended to a local embedding $f^{A \cup D} : A \cup D \to P$ or to a local embedding $f^{M \cup D} : M \cup D \to P$.

To determine whether the T-bundle of a 2-cell is trivial, we examine the behavior along the boundary of the 2-cell. We demonstrate this in the case of an embedded disk, which is most amenable to visualization. The relevant region around an embedded disk appears as in Figure 4.4 without loss of generality. (The neighborhood also contains 2-cells along the vertical lines that are not shown in the figure.) There are two possibilities for the missing region "above" the embedded disk: either a disk is attached along the blue line or along the red line. The former case yields a trivial T-bundle, while the latter yields a nontrivial T-bundle.

4.3.3 Non-Spine Fake Surfaces

Definition 4.8. A fake surface is called *unthickenable* or a *non-spine* if it does not embed in a 3-manifold.

The existence of a nontrivial T-bundle is the obstruction to a fake surface being a spine. Thus, non-spine fake surfaces are those that have at least one nontrivial T-bundle.

Theorem 4.3 ([30], Theorem 1.1.20)
A fake surface P is a spine if and only if the boundary curves of all its 2-cells have trivial T-bundles.

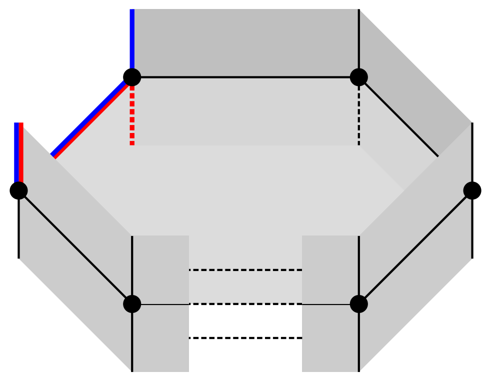


Figure 4.4: Computing T-bundles of an Embedded Disk in a Fake Surface

4.4 Fake Surfaces for Zeeman and Stable Andrews–Curtis Conjectures

The stable Andrews–Curtis conjecture involves general 2-complexes.[2] However, it is sufficient to study just the case of fake surfaces, as any 2-complex can 3-deform into a fake surface. In other words, the stable Andrews–Curtis conjecture is equivalent to the stable Andrews–Curtis conjecture restricted to fake surfaces. Moreover, restricting the Zeeman conjecture to fake surfaces makes it equivalent to the stable Andrews–Curtis conjecture in the non-spine case and to the Poincaré conjecture in the spine case. In this

[2] In general, a CW complex can be turned into a polyhedron after choosing a suitable subdivision, but there is no canonical choice of such a subdivision. Although we work within the category of polyhedra, it is often convenient to leave a space in its CW complex form without specifying any subdivision. Thus, when we refer to a "complex," we mean the underlying polyhedral space, without committing to a particular subdivision.

way, understanding fake surfaces is important for understanding these conjectures.

4.4.1 2-Complexes 3-Deform to Fake Surfaces

Given a 2-complex K, there are several algorithms in the literature to transform K into a fake surface P_K via 3-deformations. Many are based on the so-called banana-pineapple trick from Remark 2, Page 26 of [56] for spines of 3-manifolds: "First, choose a spine in the interior; expand each edge like a banana and collapse from one side; then expand each vertex like a pineapple and collapse from one face." See also [52]. We estimate the complexity of the resulting fake surface following the banana-pineapple trick in the case of the standard 2-complex realization of a group presentation, but emphasize that this is by no means the only or best way to 3-deform an arbitrary 2-complex into a fake surface.

Proposition 4.4

Let $G = \langle x_1, \ldots, x_k \mid r_1, \ldots, r_\ell \rangle$ be a group presentation. Let n_{x_i} be the total number of occurrences of x_i and x_i^{-1} across all the relators. Let $L = \sum_i n_{x_i}$ and without loss of generality assume $n_{x_1} \geq n_{x_i}$ for all i. Then applying the banana-pineapple trick to the standard 2-complex realization of G gives a fake surface of complexity $2L - 4k - n_{x_1} + 2$.

Proof. Consider the edge corresponding to x_i in the standard 2-complex realization of G. Thickening it to a banana and collapsing through a longitudinal piece of the peel creates $n_{x_i} - 2$ new edges, as each of the n_{x_i} 2-cells attached to that edge creates a new edge except for the two that border the collapsed longitudinal piece of the peel. Now thickening the vertex and collapsing the resulting pineapple through a piece of the peel creates a new vertex for each of the $\sum_{i=1}^{k} 2(n_{x_i} - 2)$ edges (where we multiply by 2 because each edge appears twice in a neighborhood of the vertex), except those that border the piece of the peel from which we collapse. In order to make the complexity as small as possible, we choose to collapse from the piece of the peel in the middle of the $n_{x_1} - 2$ edges coming from one side of the original x_1 edge, thus giving $\sum_{i=1}^{k} 2(n_{x_i} - 2) - (n_{x_1} - 2)$

vertices. Simplifying gives $2L - 4k - n_{x_1} + 2$. □

Furthermore, in [30], it is shown that every 2-complex is indeed 3-deformable to a non-spine fake surface. This can be accomplished by applying a U move, as discussed in Section 4.7.1.

4.4.2 Connection with Zeeman Conjecture

Restricting the Zeeman conjecture to spines or non-spine fake surfaces allows for a stronger connection to the Poincaré and stable Andrews–Curtis conjecture.

Proposition 4.5 ([17], Theorem 3)
The Zeeman conjecture for fake surface spines is equivalent to the Poincaré conjecture.

The forward implication was shown in Section 3.2.3; we simply modify the proof by taking a fake surface spine as opposed to a generic spine.

The reverse implication follows from the fact that if M is a homotopy ball and K is a fake surface spine, then $K \times I$ collapses to something homeomorphic to M. By the Poincaré conjecture, such a ball is unique and is certainly collapsible.

On the other hand, the Zeeman conjecture restricted to non-spine fake surfaces is equivalent to the stable Andrews–Curtis conjecture.

Proposition 4.6 ([30], Corollary 1.3.57)
The Zeeman conjecture for non-spine fake surfaces is equivalent to the stable Andrews–Curtis conjecture.

Proof. The forward implication was shown in Section 3.2.3; we simply modify the proof by first 3-deforming to a non-spine fake surface.

The reverse implication follows from the following result ([30], Theorem 1.3.54): let P_1 be a fake surface that satisfies the Zeeman conjecture. Then for any non-spine fake surface P_2 that is 3-deformation equivalent to P_1, the Zeeman conjecture also holds.

Now, if a non-spine fake surface P satisfies the stable Andrews–Curtis conjecture, it 3-deforms to a point and thus is 3-deformation equivalent to, e.g., the Bing House. Since the Bing House satisfies the Zeeman conjecture, P also satisfies the Zeeman conjecture. $\square$

These results show that to establish whether a given fake surface is a counterexample to the stable Andrews–Curtis conjecture, it suffices to attempt to 3-deform it into a fake surface spine.

4.5 Proving New Cases

In the remainder of this chapter, we discuss using fake surfaces to prove new cases of the Zeeman conjecture and the stable Andrews–Curtis conjecture.

For a given contractible fake surface, there are at least three ways to show it satisfies the stable Andrews–Curtis conjecture. The first, covered in Chapters 2 and 3, is to collapse a maximal tree of the 1-skeleton, read off a presentation, and use stable Andrews–Curtis moves from the algebraic formulation of the conjecture to reach the trivial presentation.

The remaining two ways are to construct a 3-deformation to a point or to construct a 3-deformation to a spine fake surface. We again emphasize that, at least for low complexity, spines are rare among all contractible fake surfaces, as discussed in Section 5.4 and Table 5.3.

In this section, we address the construction of a 3-deformation to a point. In particular, the results of this section, when combined with the classification of fake surfaces up to complexity 5 presented in Chapter 5, prove the following theorem:

Theorem 4.7

Let K be a contractible fake surface of complexity at most 5. Then K 3-deforms to a point. In other words, K satisfies the stable Andrews–Curtis conjecture.

We also highlight a few corollaries of this theorem. In each case, we let K be a contractible fake surface of complexity at most 5. The first corollary follows from the algebraic formulation of the stable Andrews–Curtis conjecture, discussed in Chapter 3.

Corollary 4.8

Let P be the presentation of the trivial group coming from collapsing a maximal tree of the 1-skeleton of K to a basepoint. Then the stable Andrews–Curtis conjecture holds for P.

The second corollary follows from the equivalence between the Zeeman conjecture restricted to contractible fake surfaces and the union of the stable Andrews–Curtis and Poincaré conjectures, as discussed in Section 4.4.

Corollary 4.9

The Zeeman conjecture holds for K.

The third corollary follows from the fact that K can be embedded in $\mathbb{R}^5$ by general position. Since K is contractible, the boundary of its regular neighborhood $N(K)$ is PL-homotopic to a 4-sphere. Since K satisfies the stable Andrews–Curtis conjecture, $N(K)$ is a tubular neighborhood of a point and thus a 5-ball.

Corollary 4.10

The closed regular neighborhood $N(K)$ of K embedded in $\mathbb{R}^5$ is the standard 5-ball.

To prove Theorem 4.7, we proceed by induction on complexity. Ikeda [21] showed that all contractible fake surfaces of complexity 1 satisfy the stable Andrews–Curtis and Zeeman conjectures, providing the base case. Thus, it suffices to prove that all contractible fake surfaces of complexity between 2 and 5 can be reduced in complexity.

The proof is outlined as follows: in Section 4.5.1, we give a set of conditions under which a contractible fake surface can be reduced in complexity. The classification in

Chapter 5 shows that all contractible fake surfaces up to complexity 5 meet one of these criteria. However, the fake surfaces classified in Chapter 5 are assumed to have connected 1-skeleta. In some cases, reducing the complexity of a fake surface results in a fake surface with 2 connected components in the 1-skeleton. This case is addressed in Section 4.5.2.

4.5.1 Reducing the Complexity of a Fake Surface

In this section, we prove the following lemma:

Lemma 4.11 (Complexity Reduction Lemma)
A fake surface F can be 3-deformed to another fake surface of lower complexity if F satisfies one of the following criteria:

1) F contains an embedded disk with at most three vertices and a trivial T-bundle.

2) F contains an embedded disk with at most two vertices and a nontrivial T-bundle.

3) F contains two embedded disks with three vertices that share exactly one vertex.

4) F contains one embedded disk with 3 vertices and another embedded disk with 3 or 4 vertices, and the two embedded disks share exactly one edge.

5) F contains an embedded disk with 4 vertices and trivial T-bundle and another disk with 4 vertices, and either the two disks share exactly two non-adjacent edges or the embedded disk intersects the other disk in one edge twice. If the second disk has a trivial T-bundle, then the condition that two edges are non-adjacent can be removed.

6) F contains an embedded disk with 4 vertices and trivial T-bundle, another disk with 4 vertices that intersects the embedded disk in two consecutive edges, and another disk with 4 or 5 vertices as in Figures 4.16 and 4.17.

7) F contains an embedded disk with 5 vertices and trivial T-bundle and another embedded disk with 4 vertices and trivial T-bundle, and the two disks share exactly two non-adjacent edges.

8) F contains an embedded disk with 5 vertices and trivial T-bundle and three disks with 4 vertices as in either Figures 4.19 or 4.20.

Remark (Trivial and Nontrivial Embedded Disk Moves)
In the proof of Cases 1 and 2, we describe two moves to 3-deform a closed fake surface with an embedded disk depending on if the embedded disk has a trivial or nontrivial T-bundle. In particular, we obtain a new surface by collapsing the union of the old surface and a 3-ball along a disk. We call these the *trivial embedded disk move* and *nontrivial embedded disk move*, respectively. When we apply these moves to an embedded disk, we say that we do so "along" an edge(s), which refers to the edge(s) that are removed (see Sections 4.5.3 and 4.5.4). After being introduced in Cases 1 and 2, these moves are then used throughout the proof of Lemma 4.11.

When the embedded disk has at most 3 vertices on the boundary and trivial T-bundle or at most 2 vertices and nontrivial T-bundle, applying the corresponding embedded disk move results in a surface of lower complexity. The moves in this situation are discussed in [30], and this is what is needed to show Cases 1 and 2. For completeness, we provide full details for the most general cases in our proofs below. We also describe the trivial embedded disk move algorithmically in the case of an embedded disk with 4 vertices in Section 4.5.3 and the nontrivial embedded disk move in the case of 3 vertices in Section 4.5.4, which we use in various other cases in the proof. The embedded disk moves for arbitrary number of vertices is again explored in Section 4.7.2.

Proof of Lemma 4.11. Before beginning the casework, we introduce a few conventions.

For a fake surface F, we define $\mathcal{S}_n(F)$ to be the set of type n singularities of F. In particular, $|\mathcal{S}_2(F)|$ is the number of vertices of F.

We use the notation $[x_1,\ldots,x_n]$ to represent the attaching map of a disk, where a negative denotes inverse. Similarly, (x_1, x_2, x_3) denotes a part of an attaching map of a disk. In Case 3, we use adjacent disks to give explicit descriptions of T-bundles in accordance with Figure 4.4.

Case 1

Suppose F_1 is a closed fake surface with an embedded disk D and there are n vertices on the boundary ∂D of the disk D. We discuss the general trivial embedded disk move and then show that when $n \leq 3$, the complexity decreases.

When the T-bundle over ∂D is trivial, the neighborhood of D in F_1 is as shown in Figure 4.5, where we represent the edges of 1-skeleton by green lines, vertices in $\mathcal{S}_2(F_1)$ by black points, D by the blue disk. Denote by A the annulus above D and with one boundary component attached to ∂D and the other boundary component indicated by the red circle in Figure 4.5. Set $J = D \cup A$. Then J is a disk embedded in F_1, and $J \times I$ is a 3-ball topologically. We denote by E the 3-polyhedron obtained by attaching $J \times I$ to F_1 such that $J \times 0 = J$.

As shown in Figure 4.6, $J \times I$ consists of n 3-subpolyhedrons whose 1-faces are indicated by red and blue lines. They are all 3-balls topologically. Now we collapse these 3-balls along certain 2-faces to get a new closed fake surface F_2. First, we collapse the 3-ball indicated by the left diagram in Figure 4.6 along the free 2-face enclosed by red lines. Next we collapse the other 3-balls indicated by the right diagram in Figure 4.6 by pushing upwards the 2-faces enclosed by red lines.

When $n \geq 2$, we get a new closed fake surface locally as shown in Figure 4.7. The original vertices A_1 and A_2 are removed and new vertices $A'_3, \ldots, A'_n$ are introduced, so the new surface F_2 contains $|\mathcal{S}_2(F_1)| + n - 4$ vertices.

In particular, for $n = 2$, the number of vertices decreases by 2: $|\mathcal{S}_2(F_2)| = |\mathcal{S}_2(F_1)| - 2$. Green edges are separated into two disjoint parts locally as shown in Figure 4.7. Thus

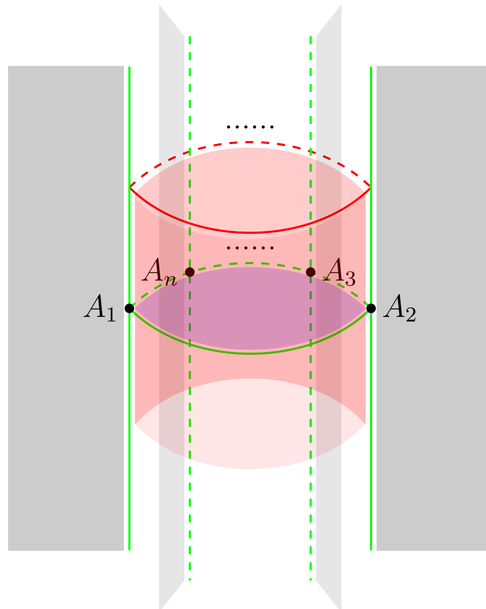


Figure 4.5: Neighborhood of D in F_1 for trivial bundles

F_2 may have one more connected component in its 1-skeleton than F_1. For $n = 3$, the number of vertices decreases by 1. Green edges are still connected together, so F_2 has the same number of connected components of 1-skeleton as F_1. In this case, the move is simply the T^{-1} move from [30] (see Section 4.7.1).

When $n = 1$, the collapses above introduce a free 1-face as indicated by the red line in Figure 4.8. Collapsing the disk containing this free 1-face may increase the number of connected components of the 1-skeleton. If the complexity is greater than 1, the embedded disk D must contain a vertex other than A_1. Thus $|\mathcal{S}_2(F_2)| \leq \max\{|\mathcal{S}_2(F_1)| - 2, 0\}$.

Case 2

Again suppose F_1 is a closed fake surface with an embedded disk D and there are n vertices on the boundary ∂D of the disk D. We discuss the general nontrivial embedded disk move and then show that when $n \leq 3$, the complexity decreases.

When the T-bundle over ∂D is nontrivial, the neighborhood of D in F_1 is as shown in Figure 4.9. Denote by B the upper band of the Möbius band, where one long edge

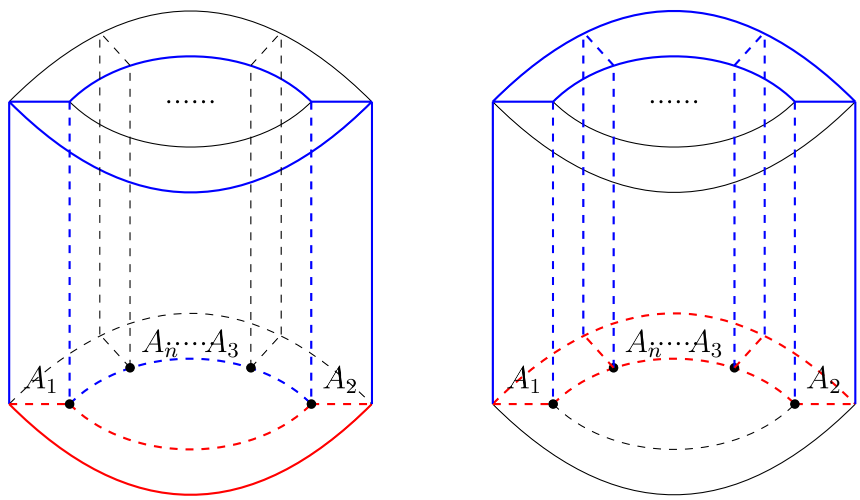
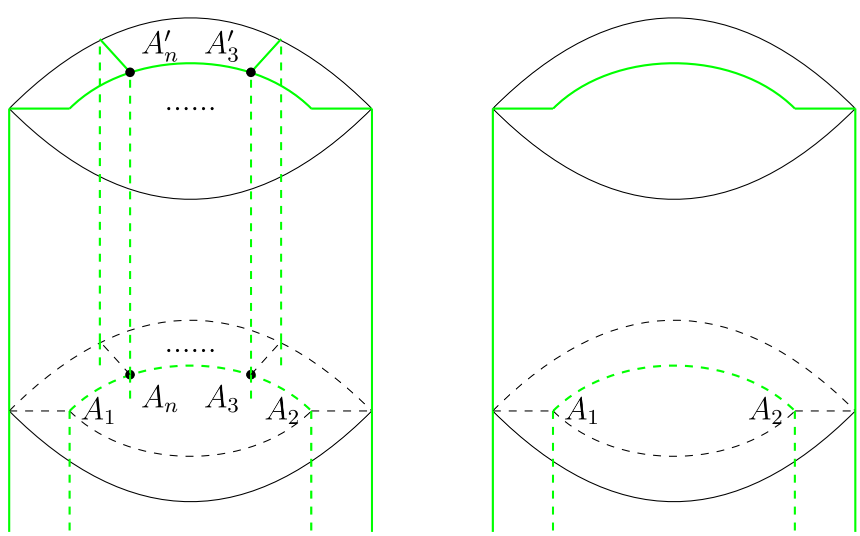


Figure 4.6: $J \times I$ for trivial bundles



$n \geq 3$ $n = 2$

Figure 4.7: F_2 for trivial bundles

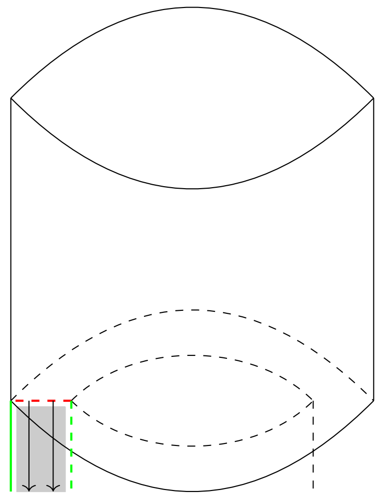


Figure 4.8: F_2 with free 1-face for $n = 1$

is attached to ∂D and the other long edge is indicated by the red line in the bottom diagram in Figure 4.10. Set $J = D \cup B$. Then J is an embedded disk in F_1 as shown in Figure 4.10. Attach 3-ball $J \times I$ to F_1 along $J \times 0 = J$ to get E.

As shown in Figure 4.11, $J \times I$ consists of n 3-balls whose 1-faces are indicated by red and blue lines. Similarly to the case of trivial bundles, we collapse E to get F_2 as follows. First, we collapse the 3-ball indicated by the left diagram in Figure 4.11 along the free 2-face enclosed by red lines. Next, we collapse the other 3-balls indicated by the right diagram in Figure 4.11 by pushing upwards the 2-faces enclosed by red lines.

When $n \geq 2$, we get a new closed fake surface indicated by the left diagram in Figure 4.12. The original vertex A_2 is removed and new vertices $A'_3, \ldots, A'_n$ are introduced. The original vertex A_1 remains, unlike in the case of trivial bundles, so the new surface F_2 contains $|\mathcal{S}_2(F_1)| + n - 3$ vertices.

In particular, for $n = 2$, the number of vertices decreases by 1. Green edges are still connected together, so F_2 has the same number of connected components of 1-skeleton as F_1. In this case, the move is simply the U^{-1} move from [30] (see Section 4.7.1). Note that for $n = 3$, the number of vertices is unchanged.

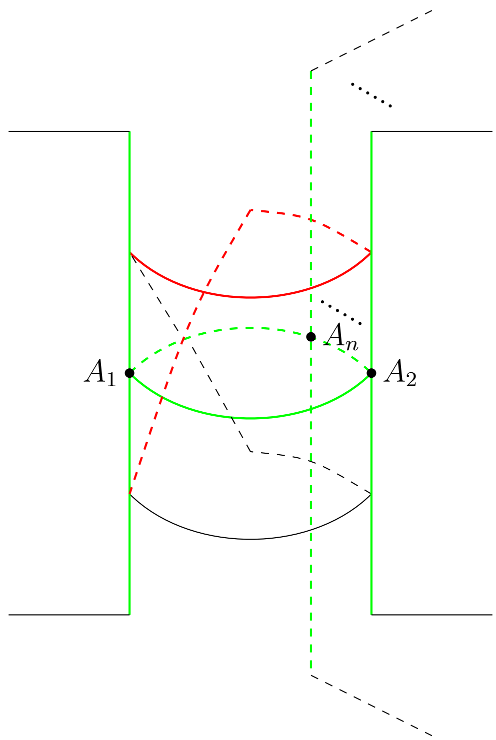


Figure 4.9: Neighborhood of D in F_1 for nontrivial bundles

When $n = 1$, we get a closed surface F_2 indicated by the right diagram in Figure 4.12. The original green circle $\partial D \subset \mathcal{S}_2(F_1)$ is changed to a dashed circle which represents a circle in a region. The other two remaining green edges are connected. Thus F_2 has the same number of connected components of 1-skeleton as F_1. The unique original vertex A_1 is removed, and so F_2 contains one fewer vertex than F_1.

Case 3

It suffices to consider the case where both embedded disks have nontrivial T-bundles, since otherwise we can reduce the complexity by Case 1. In Figure 4.13, let the T-bundle of the red disk $[7, -12, 4]$ be given by

$$(3, 7, 15), (-15, -12, 13), (-13, 4, 1), (-1, 7, 16), (-16, -12, 14), (-14, 4, -3),$$

and the T-bundle of the blue disk $[1, 2, 3]$ be given by

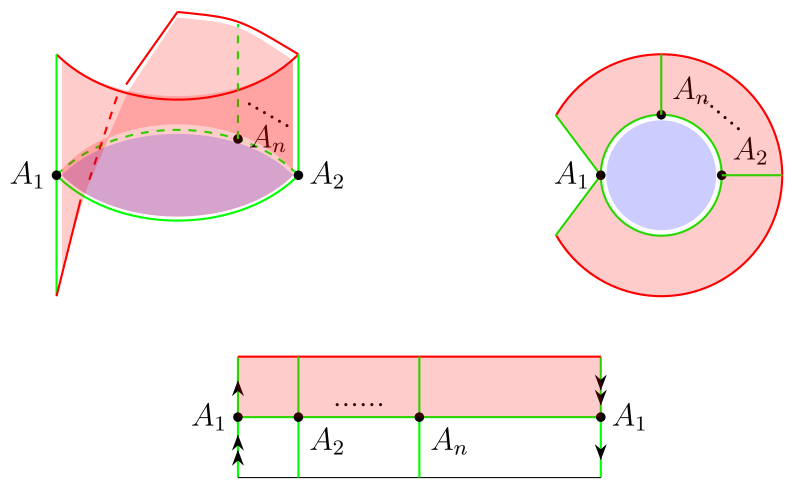


Figure 4.10: J for Nontrivial T-bundles

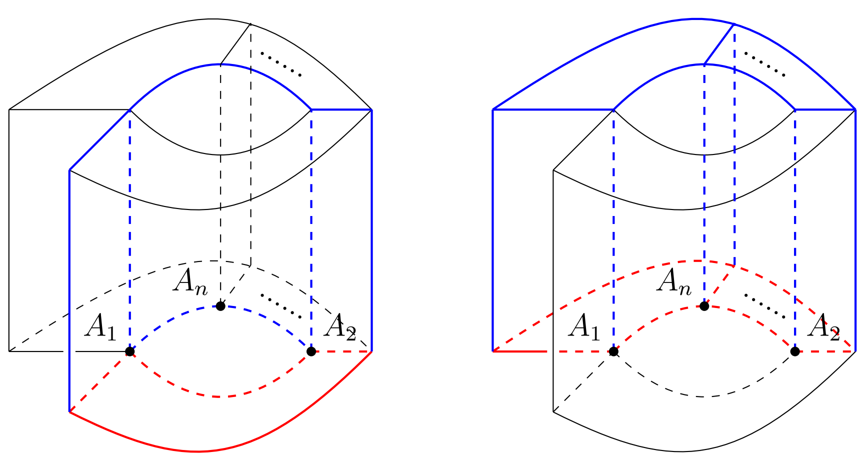


Figure 4.11: $J \times I$ for Nontrivial T-bundles

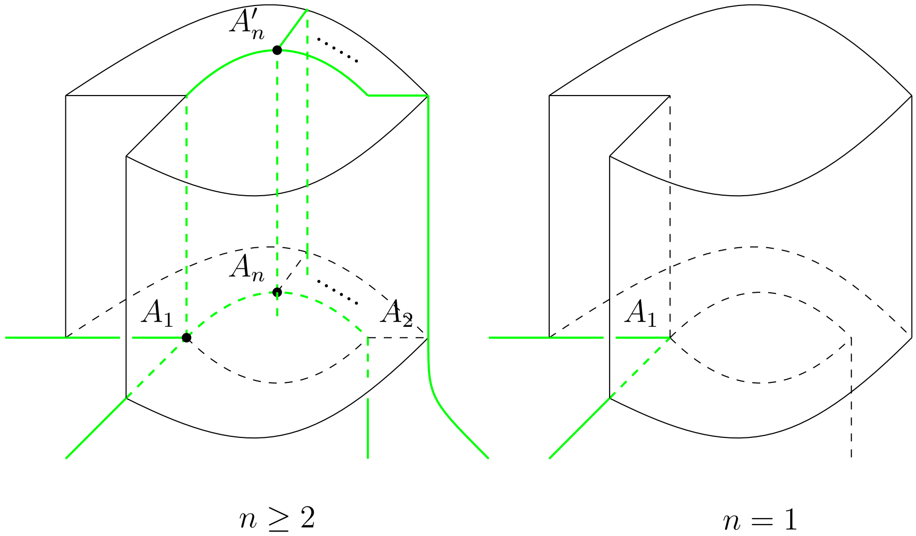


Figure 4.12: F_2 for Nontrivial T-bundles

$$(4,1,5),(-5,2,6),(-6,3,7),(-7,1,8),(-8,2,9),(-9,3,-4).$$

We now apply the nontrivial embedded disk move to the blue disk along edge 2. Applying the nontrivial embedded disk move to an embedded disk with 3 vertices is given algorithmically in Section 4.5.4. After applying the move, the red disk becomes another embedded disk with 3 vertices whose T-bundle given by

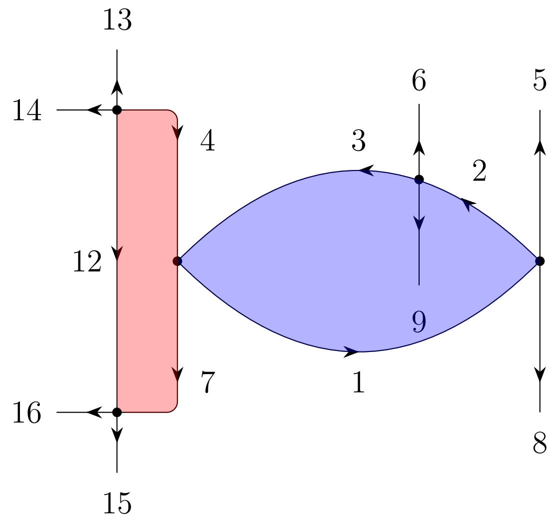
$$(11,7,15),(-15,-12,13),(-13,4,-11),$$

where edge 11 is as in Figure 4.24. This is a trivial T-bundle, so we can reduce the complexity by Case 1.

Case 4

It suffices to assume that the T-bundle of the blue disk in Figure 4.14 is nontrivial, since otherwise we can apply case 1. We apply the nontrivial embedded disk move to the blue disk along the shared edge such that the disks are changed to two disks that share exactly one vertex. The two possibilities are shown in Figure 4.14. In each case, we can apply Case 1, 2, or 3 to reduce the complexity.

Figure 4.13: Configuration for Case 3



Two embedded disks with 3 vertices that share exactly one vertex.

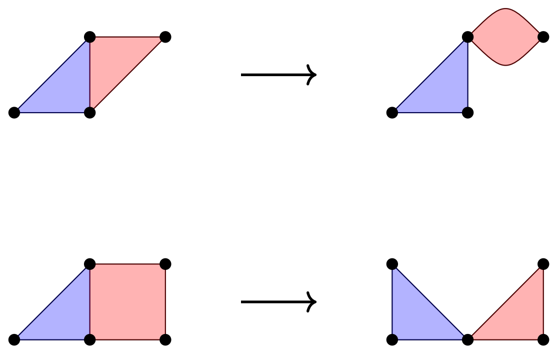
Case 5

The two possible configurations for Case 5 are shown in Figure 4.15. We apply the trivial embedded disk move to the blue disk along the thickened edges. The red disk will be changed to a disk with 2 vertices, allowing us to reduce the complexity using Case 1 or 2. If the red disk has a trivial T-bundle, the two edges on which the disks intersect can be adjacent. This is because the trivial embedded disk move will not change the triviality of disks, and in the case of adjacent edges, the second disk will become a disk with no more than 3 vertices and trivial T-bundle. Thus we can reduce the complexity using Case 1.

Case 6

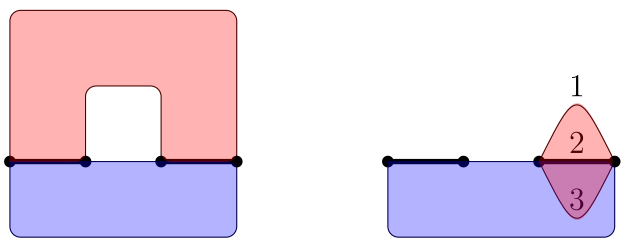
The two possible configurations in Case 6 are given in Figures 4.16 and 4.17. It is sufficient to assume that the T-bundles of the second and third disks are nontrivial, since otherwise we can reduce the complexity by Case 5. In the first case, shown in Figure 4.16, we do the trivial embedded disk move with the blue disk $[1, 2, 3, 4]$ along edges 2 and 4.

Figure 4.14: Configurations for Case 4



Left: An embedded disk with 3 vertices and an embedded disk with 3 or 4 vertices that share exactly one edge. **Right:** The resulting configuration after applying the nontrivial embedded disk move to the blue disk.

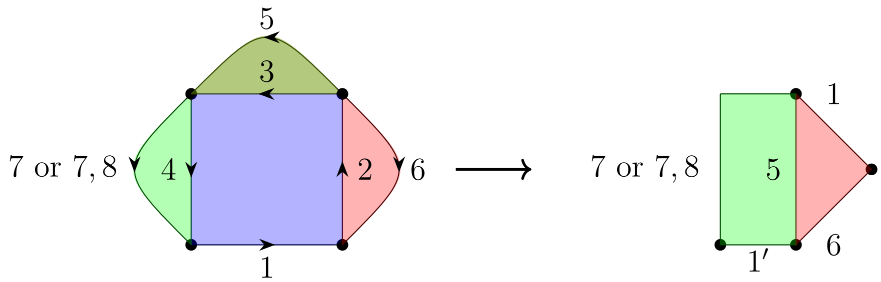
Figure 4.15: Configurations for Case 5



Left: Two embedded disks with 4 vertices that share two non-adjacent edges. **Right:** Two embedded disks with 4 vertices where the red disk intersects the blue disk at one edge twice. The red disk's attaching map is $[2, 1, 2, 3]$.

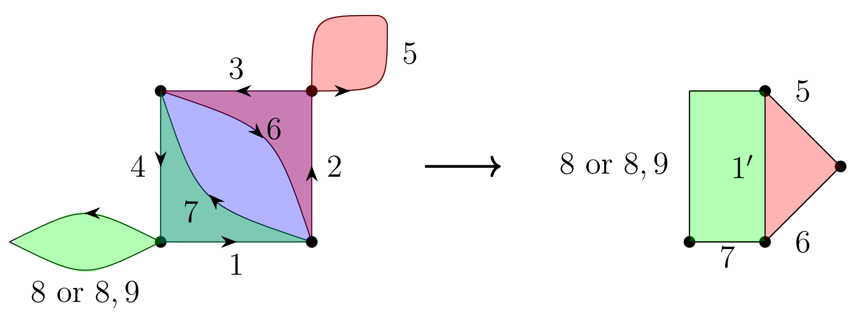
We get an embedded disk with 3 vertices and an embedded disk with 3 or 4 vertices that share exactly one edge, and the resulting fake surface has the same complexity as before. Thus, we can apply Case 4. In the second case, shown in Figure 4.17, we again do the trivial embedded disk move with the blue disk $[1, 2, 3, 4]$ along edges 2 and 4. This again gives an embedded disk with 3 vertices and an embedded disk with 3 or 4 vertices that share exactly one edge, and again the resulting fake surface has the same complexity as before. We apply Case 4 to reduce the complexity.

Figure 4.16: First Configuration for Case 6



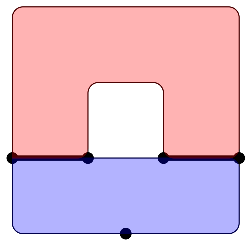
Left: There is an embedded disk $[1, 2, 3, 4]$, a disk $[2, 5, -3, 6]$, and a disk $[-4, -5, 3, 7]$ or $[-4, -5, 3, 7, 8]$. **Right:** The resulting configuration has an embedded disk with 3 vertices and an embedded disk with 3 or 4 vertices that share exactly one edge.

Figure 4.17: Second Configuration for Case 6



Left: There is an embedded disk $[1, 2, 3, 4]$, a disk $[2, 5, 3, 6]$, and a disk $[1, 7, 4, 8]$ or $[1, 7, 4, 8, 9]$. **Right:** The resulting configuration has an embedded disk with 3 vertices and an embedded disk with 3 or 4 vertices that share exactly one edge.

Figure 4.18: Configuration for Case 7



An embedded disk with 5 vertices and an embedded disk with 4 vertices that share two non-adjacent edges.

Case 7

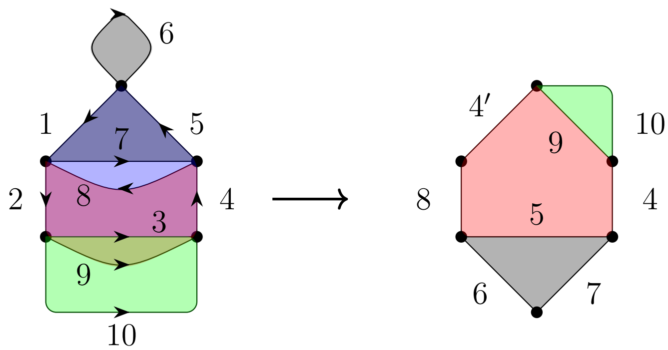
The configuration of Case 7 is shown in Figure 4.18. In this case, we do the trivial embedded disk move to the blue disk with 5 vertices along the thick edges as in Figure 4.18. This increases the complexity of the resulting fake surface by 1. However, the red disk is changed to a disk with two vertices and trivial T-bundle. We then apply the trivial embedded disk move to this disk, which decreases the complexity by 2. Thus, the resulting surface is of lower complexity than the original one.

Case 8

The first case is shown in Figure 4.19. We can assume that $[2, 9, 4, 8]$ and $[10, -3, 9, -3]$ have nontrivial T-bundles, since otherwise we can reduce the complexity by Case 7. We do the trivial embedded disk move to $[1, 2, 3, 4, 5]$ along edges 1 and 3, which increases the complexity by 1. The resulting fake surface has three embedded disks as in the right side of Figure 4.19, where the green disk with 2 vertices and the red disk with 5 vertices have nontrivial T-bundles. We then do the nontrivial embedded disk move to the green disk to remove edge 9 from the red disk and go back to the original complexity. The new red disk has 4 edges, and so we can apply Case 4 to the new red disk and the gray disk. We can use a similar method for the second case in Figure 4.20.

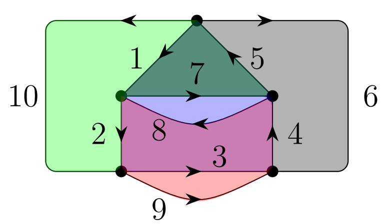
□

Figure 4.19: First Configuration for Case 8



Left: There are embedded disks [1,2,3,4,5], [1,7,5,6], [2,9,4,8], and [10,-3,9,-3]. **Right:** The resulting configuration has an embedded disk with 2 vertices, and embedded disk with 5 vertices, and an embedded disk with 3 vertices.

Figure 4.20: Second Configuration for Case 8



Embedded disks $[1, 2, 3, 4, 5]$, $[10, -2, 7, 5]$, $[2, 9, 4, 8]$, and $[6, 4, -7, -1]$.

4.5.2 Fake Surfaces with Disconnected 1-Skeleta

Using the classification of contractible fake surfaces up to complexity 5, which is presented in detail in Chapter 5, it is easy to check that all satisfy one of the conditions of Lemma 4.11. Thus, we can always 3-deform each surface to another of lower complexity. However, the classification is for fake surfaces with connected 1-skeleta. When we apply the reduction to lower complexity, we may get a fake surface whose 1-skeleton has 2 connected components. We henceforth use $\mathcal{F}_C(s,t)$ to refer to contractible fake surfaces of complexity t whose 1-skeleton has s connected components.

As discussed in the proof of Lemma 4.11, only the trivial embedded disk move to embedded disks with one or two vertices from Case 1 may give a fake surface in $\mathcal{F}_C(2,t)$. In this case, the number of vertices will decrease by at least 2. Thus, it remains to address the case of these fake surfaces up to complexity 3.

In [21], it is shown the fake surfaces in $\mathcal{F}_C(2,1)$ satisfy the stable Andrews–Curtis and the Zeeman conjectures, so it remains to deal with the fake surfaces in $\mathcal{F}_C(2,2)$ and $\mathcal{F}_C(2,3)$.

By Lemma 2.9 in [23], each surface $F \in \mathcal{F}_C(2,3)$ can be constructed in one of the following two ways:

Case 1

In this case, we construct a fake surface $F = F_1 \# F_2$ where $F_1 \in \mathcal{F}_C(1,2)$ and $F_2 \in \mathcal{F}(1,1)$ such that $H_1(F_2) = 0$ and $H_2(F_2) = \mathbb{Z}$. Here, $\#$ represents the connected sum of two fake surfaces along their regions. It is straightforward to check that there are three possibilities for F_2. Their attaching maps are as follows, attached onto the 1-skeleton in Figure 4.21:

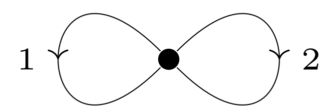


Figure 4.21: 1-Skeleton for F_2

Fake Surface 1: $[-2, 1, 1], [1, 2], [2]$
Fake Surface 2: $[-2, -1, 2, 1], [2], [1]$
Fake Surface 3: $[-2, 1, -2, -1], [2], [1]$

If F_2 is the first fake surface above, through computation about homology groups, the connected sum between F_1 and F_2 can not be done along the unique embedded disk [2] of F_2. Thus $F = F_1 \# F_2$ contains at least one embedded disk with one vertex such that we can reduce the complexity. When F_2 is either of the other two surfaces, we can always reduce the complexity since both of the surfaces contain two embedded disks.

Case 2

In this case, $F = F_1 \cup F_2$, where F_1 is obtained by removing one disk in the interior of a 2-cell of one surface in $\mathcal{F}_C(1, 3)$, and F_2 is obtained by attaching the boundary of a disk to the center of a Möbius band. F_1 and F_2 are attached along their boundaries. We can collapse F_2 as follows: the 1-skeleton of F_2 is a circle disjoint from the 1-skeleton of F_1. Based on the construction of F_2, F_2 contains an embedded disk with the circle as the boundary and the T-bundle of the disk is nontrivial, as shown by the left blue disk in Figure 4.22. We apply the inverse of the nontrivial embedded disk move, called the loop move in [30], to the blue disk. The blue disk becomes a disk with one vertex and trivial T-bundle. Finally, we do the trivial embedded disk move to the new blue disk and

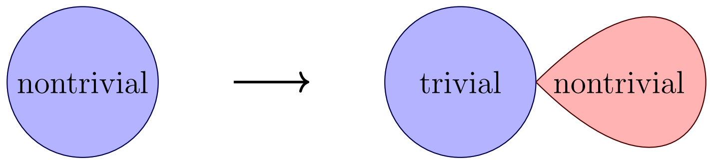


Figure 4.22: Applying Inverse of Nontrivial Embedded Disk Move

remove the connected component of 1-skeleton of F.

We can apply the same argument to the two ways to construct the surfaces $F \in \mathcal{F}_C(2,2)$, following the construction in [23], and use the same method to reduce.

Thus, we can use Lemma 4.11 to reduce each $F \in \mathcal{F}_C(1,t)(1 < t \leq 5)$. When a reduction takes us outside $\mathcal{F}_C(1,t)$, we can use the above method to make it return to $\mathcal{F}_C(1,t)$ or reduce to a point. Since all contractible fake surfaces up to complexity 5 meet one of the criteria of Lemma 4.11 (see Chapter 5), this completes the proof of Theorem 4.7 in all cases.

4.5.3 Trivial Embedded Disk Move with Four Vertices

In this section, we give an algorithmic description of the trivial embedded disk move in the case of 4 vertices. Suppose that the embedded disk is given by $[1,2,3,4]$ and the neighborhood of the embedded disk is given by the following subsets of disk attaching maps:

$$\begin{gathered}(5,1,6),(-6,2,7),(-7,3,8),(-8,4,-5),\\(9,1,10),(-10,2,11),(-11,3,12),(-12,4,-9),\end{gathered}$$

as in the left side of Figure 4.23. Two of edges $5,6,\ldots,12$ may represent the same edge depending on the surface.

In addition to the 8 subsets of disk attaching maps above, the following 4 subsets of disk attaching maps may be changed when executing the move:

$$(5, -9), (-8, 12), (-7, 11), \text{ and } (-6, 10).$$

In this case, there are two ways to do the operation, depending on the “front” edge, as having edge 1 or edge 3 in the front is equivalent and having edge 2 or edge 4 in the front is equivalent.

Based on edge 1 being the “front” edge as in the left side of Figure 4.23, the neighborhood of the new surface is shown in the right side of Figure 4.23. During the move, edge 1 is removed. Edges 4 and 9 are combined to a single edge denoted by $4(-9)$, and edges 2 and 10 are combined to a single edge denoted by $10(-2)$. Three new edges, 13, 14, and 15, are introduced. The embedded disk $[1, 2, 3, 4]$ is changed to the embedded disk $[-3, 13, 15, -14]$. We now give explicit descriptions for how the aforementioned 12 subsets of disk attaching maps are changed. In 4 cases, the number of edges in the attaching map is not changed:

$$\begin{aligned}
(5, 1, 6) &\mapsto (5, -15, 6), \\
(9, 1, 10) &\mapsto (-4(9), -3, 10(-2)), \\
(-7, 3, 8) &\mapsto (-7, 15, 8), \\
(-11, 3, 12) &\mapsto (-11, 3, 12).
\end{aligned}$$

In 4 cases, the number of the edges in the attaching map increases:

$$\begin{aligned}
(5, -9) &\mapsto (5, -14, 4(-9)), \\
(-8, 12) &\mapsto (-8, -14, 12), \\
(-7, 11) &\mapsto (-7, -13, 11), \\
(-6, 10) &\mapsto (-6, -13, 10(-2)).
\end{aligned}$$

Finally, in 4 cases, the number of the edges decreases:

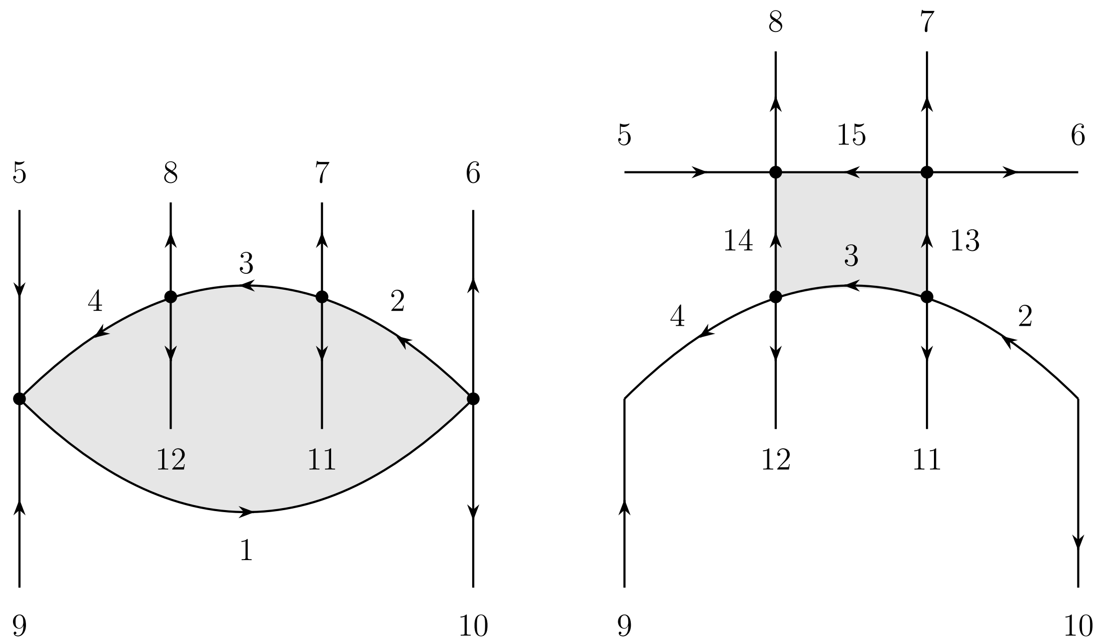


Figure 4.23: Trivial Embedded Disk Move with 4 Vertices

$$\begin{aligned}
(-6,2,7) &\mapsto (-6,7),\\
(-10,2,11) &\mapsto (-10(2),11),\\
(-8,4,-5) &\mapsto (-8,-5),\\
(12,4,-9) &\mapsto (12,4(-9)).
\end{aligned}$$

In this case, we say that we do the trivial embedded disk move to the disk $[1,2,3,4]$ along edges 2 and 4, as these are the edges that are “removed” by the move.

The other case comes from rotating the disk $[1,2,3,4]$ by one edge, which relabels the edges as follows:

$$\begin{aligned}
&1 \mapsto 4, 2 \mapsto 1, 3 \mapsto 2, 4 \mapsto 3,\\
&5 \mapsto -8, 6 \mapsto -5, 7 \mapsto 6, 8 \mapsto 7,\\
&9 \mapsto -12, 10 \mapsto -9, 11 \mapsto 10, 12 \mapsto 11.
\end{aligned}$$

We can then apply the move in exactly the same way.

4.5.4 Nontrivial Embedded Disk Move with 3 Vertices

In this section, we give an algorithmic description of the nontrivial embedded disk move in the case of 3 vertices. Suppose that the embedded disk is given by $[1, 2, 3]$ and the neighborhood of the embedded disk is given by the following subsets of disk attaching maps:

$$(4, 1, 5), (-5, 2, 6), (-6, 3, 7), (-7, 1, 8), (-8, 2, 9), (-9, 3, -4),$$

as in the left side of Figure 4.24. Two of edges $4, 5, \ldots, 9$ may represent the same edge depending on the surface.

In addition to the 6 subsets of disk attaching maps above, the following 3 subsets of disk attaching maps may be changed when executing the move:

$$(4, 7), (-6, 9), \text{ and } (-5, 8).$$

In this case, there are three ways to do reduction, again depending on the "front" edge. Each of edge 1, edge 2, and edge 3 being the front edge gives a distinct fake surface.

Based on edge 1 being the "front" edge as in the left side of Figure 4.24, the neighborhood of new surface is shown in the right side of Figure 4.24. During the move, edge 1 is removed. Edges -2 and 8 are combined to new edge denoted by $8(-2)$. Two new edges, 10 and 11, are introduced. The embedded disk $[1, 2, 3]$ is changed to embedded disk $[3, -11, -10]$. We now give explicit descriptions for how the aforementioned 9 subsets of disk attaching maps are changed. In 5 cases, the number of edges in the attaching map is not changed:

$$\begin{aligned}
(4, 1, 5) &\mapsto (4, -11, 5),\\
(-6, 3, 7) &\mapsto (-6, 11, 7),\\
(-7, 1, 8) &\mapsto (-7, -3, 8(-2)),\\
(-9, 3, -4) &\mapsto (-9, 3, -4),\\
(4, 7) &\mapsto (4, 7).
\end{aligned}$$

In 2 cases, the number of edges increases:

$$(-6, 9) \mapsto (-6, -10, 9).$$
$$(-5, 8) \mapsto (-5, -10, 8(-2)).$$

Finally, in 2 cases, the number of edges decreases:

$$(-5, 2, 6) \mapsto (-5, 6),$$
$$(-8, 2, 9) \mapsto (-8(2), 9).$$

In this case, we say that we do the operation with $[1, 2, 3]$ along the edge 2, as it is the edge that is "removed" by the move.

The other two cases come from rotating the disk $[1, 2, 3]$ by one or two edges and relabel the edges. If we rotate by one edge, we relabel as follows:

$$1 \mapsto 3, 2 \mapsto 1, 3 \mapsto 2,$$
$$4 \mapsto -6, 5 \mapsto -4, 6 \mapsto 5,$$
$$7 \mapsto 9, 8 \mapsto 7, 9 \mapsto 8.$$

If we instead rotate by two edges, we relabel as follows:

$$1 \mapsto 2, 2 \mapsto 3, 3 \mapsto 1,$$
$$4 \mapsto -5, 5 \mapsto 6, 6 \mapsto -4,$$
$$7 \mapsto 8, 8 \mapsto 9, 9 \mapsto 7.$$

In each case, we apply the move in exactly the same way.

4.6 Fake Surfaces for AK(n)

The goal of this line of investigation is to use the fake surface approach to prove new results about the stable Andrews–Curtis conjecture and the Zeeman conjecture. As

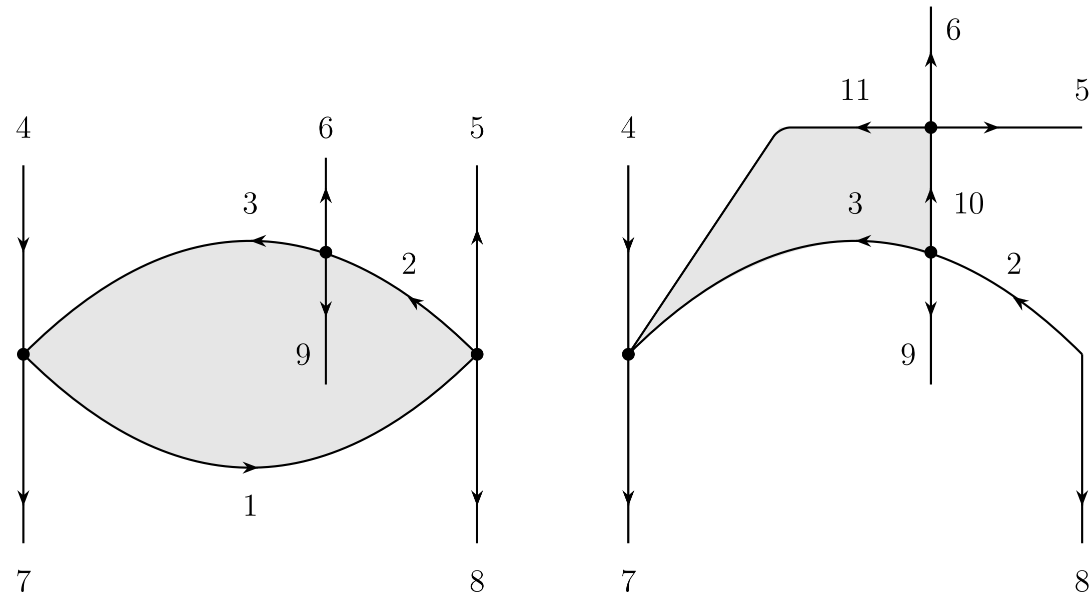


Figure 4.24: Nontrivial Embedded Disk Move with 3 Vertices

discussed in Chapters 2 and 3, the family $\mathrm{AK}(n) = \langle x, y \mid xyx = yxy, x^n = y^{n+1}\rangle$ for $n \geq 3$ is the most prominent family of potential counterexamples to these conjectures.

While our classification recovers the already well-established result that AK(2) is AC-trivial, it does not establish the same for $\mathrm{AK}(n)$ with $n \geq 3$. In other words, we cannot yet prove that a fake surface corresponding to the presentation $\mathrm{AK}(n)$ for $n \geq 3$ is 3-deformation equivalent to a fake surface for which the conjecture is known to hold. Equivalently, we cannot prove that any presentation arising from a fake surface for which the conjecture is known to hold is stably AC-equivalent to $\mathrm{AK}(n)$ via the stable AC-moves.

Note that we can construct a fake surface for $\mathrm{AK}(n)$ in the standard way, as discussed in Section 4.4.1, by creating a 2-complex with one vertex, two edges, and two 2-cells, and then applying the banana-pineapple trick to standardize the local behavior to that of a fake surface. Using this approach, we obtain a fake surface of complexity at least $3n+4$. In particular, for AK(3), this method yields a fake surface of complexity 13, which is beyond the reach of our current classification.

It is therefore valuable to construct fake surfaces for $\mathrm{AK}(n)$ with lower complexity.

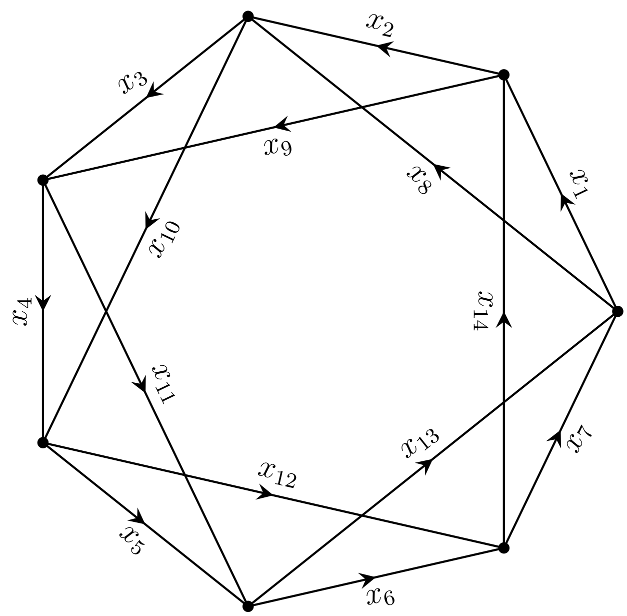


Figure 4.25: One-skeleton for AK(3) Fake Surface

In this section, we establish the following result in the case of $n = 3$:

Theorem 4.12 (Fake Surface for AK(n))

For each $n \geq 3$, there exists a fake surface F of complexity $2n + 1$ such that the presentation obtained by collapsing a maximal tree of the 1-skeleton of F to a basepoint is stably AC-equivalent to AK(n).

This result can be equivalently stated in the topological setting: *there exists a fake surface F of complexity $2n+1$ that is 3-deformation equivalent to the standard 2-complex realization of $AK(n)$.*

In the remainder of this section, we prove this theorem for the case $n = 3$, giving an explicit construction. The general construction for arbitrary n will be presented in forthcoming joint work with Jon McCammond [12].

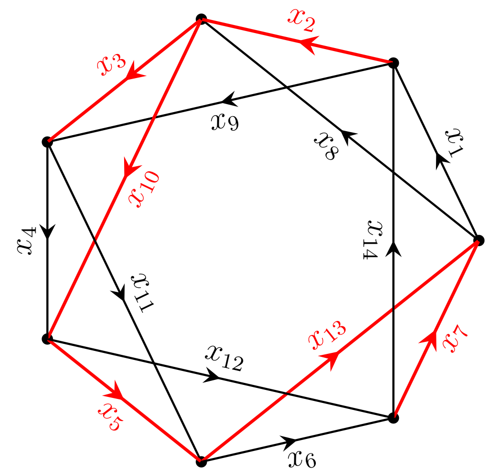


Figure 4.26: Maximal Tree of 1-skeleton of AK(3) Fake Surface

The edges of the maximal tree, $[x_2, x_3, x_5, x_7, x_{10}, x_{13}]$, are shown in red.

4.6.1 Fake Surface for AK(3)

To construct the surface for AK(3), we start with the 1-skeleton in Figure 4.25 with $2(3) + 1 = 7$ vertices and attach 2-cells according to the following:

Disk 1 : $x_1x_9x_3^{-1}x_8^{-1}$
Disk 2 : $x_3x_{11}x_{13}x_7^{-1}x_{12}^{-1}x_{10}^{-1}$
Disk 3 : $x_7x_8x_{10}x_4^{-1}x_9^{-1}x_{14}^{-1}$
Disk 4 : $x_4x_{12}x_6^{-1}x_{11}^{-1}$
Disk 5 : $x_6x_{14}x_2x_8^{-1}x_{13}^{-1}$
Disk 6 : $x_2x_{10}x_5x_{11}^{-1}x_9^{-1}$
Disk 7 : $x_5x_{13}x_1x_{14}^{-1}x_{12}^{-1}$
Disk 8 : $x_1x_2x_3x_4x_5x_6x_7$

To prove the theorem in this case, we select a maximal tree of the 1-skeleton (see Section 3.2.2). We choose the maximal tree $T = [x_{10}, x_2, x_7, x_3, x_5, x_{13}]$, shown in red in Figure 4.26. We will demonstrate that the presentation obtained by collapsing T to a basepoint is stably AC-equivalent to AK(3).

After collapsing the maximal tree, we obtain the following presentation:

$$\begin{aligned}\langle x_1, x_4, x_6, x_8, x_9, x_{11}, x_{12}, x_{14} \mid & x_1x_9x_8^{-1}, x_{11}x_{12}^{-1}, x_8x_4^{-1}x_9^{-1}x_{14}^{-1}, \\ & x_4x_{12}x_6^{-1}x_{11}^{-1}, x_6x_{14}x_8^{-1}, x_{11}^{-1}x_9^{-1}, \\ & x_1x_{14}^{-1}x_{12}^{-1}, x_1x_4x_6\rangle.\end{aligned}$$

Now we can simplify this presentation using stable AC-moves. In particular, we use repeated applications of substitution and removal as presented in Lemma 3.2. First, we use the second relation to substitute all occurrences of x_{11} with x_{12}, and then remove this second relation. What remains is:

$$\begin{aligned}\langle x_1, x_4, x_6, x_8, x_9, x_{12}, x_{14} \mid & x_1x_9x_8^{-1}, x_8x_4^{-1}x_9^{-1}x_{14}^{-1}, x_4x_{12}x_6^{-1} \\ & x_{12}^{-1}, x_6x_{14}x_8^{-1}, x_{12}^{-1}x_9^{-1}, x_1x_{14}^{-1}x_{12}^{-1}, x_1x_4x_6\rangle.\end{aligned}$$

Next, we use the third-to-last relation to substitute all occurrences of x_{12} with x_9^{-1}, and then remove this relation. This leaves us with:

$$\begin{aligned}\langle x_1, x_4, x_6, x_8, x_9, x_{14} \mid & x_1x_9x_8^{-1}, x_8x_4^{-1}x_9^{-1}x_{14}^{-1}, x_4x_9^{-1}x_6^{-1}x_9, \\ & x_6x_{14}x_8^{-1}, x_1x_{14}^{-1}x_9, x_1x_4x_6\rangle.\end{aligned}$$

We then use the second relation to substitute x_4 with $x_9^{-1}x_{14}^{-1}x_8$, and remove this relation. We have:

$$\begin{aligned}\langle x_1, x_6, x_8, x_9, x_{14} \mid & x_1x_9x_8^{-1}, x_{14}^{-1}x_8x_9^{-1}x_6^{-1}, x_6x_{14}x_8^{-1}, \\ & x_1x_{14}^{-1}x_9, x_1x_9^{-1}x_{14}^{-1}x_8x_6\rangle.\end{aligned}$$

Now we use the first relation to substitute x_9 with $x_1^{-1}x_8$ in the second relation:

$$\begin{aligned}\langle x_1, x_6, x_8, x_9, x_{14} \mid & x_1x_9x_8^{-1}, x_{14}^{-1}x_1x_6^{-1}, x_6x_{14}x_8^{-1}, \\ & x_1x_{14}^{-1}x_9, x_1x_9^{-1}x_{14}^{-1}x_8x_6\rangle.\end{aligned}$$

Now, using the second relation, we substitute x_6 with $x_{14}^{-1}x_1$ and remove this relation.

We are left with:

$$\begin{aligned}\langle x_1, x_8, x_9, x_{14} \mid\ & x_{14}^{-1}x_1x_{14}x_8^{-1}, x_1x_{14}^{-1}x_9, \\ & x_1x_9^{-1}x_{14}^{-1}x_8x_{14}^{-1}x_1, x_1x_9x_8^{-1}\rangle.\end{aligned}$$

Now we use the first relation to substitute x_8 with $x_{14}^{-1}x_1x_{14}$ and remove this relation. We have:

$$\langle x_1, x_9, x_{14} \mid x_1x_{14}^{-1}x_9, x_1x_9^{-1}x_{14}^{-1}x_{14}^{-1}x_1x_1, x_1x_9x_{14}^{-1}x_1^{-1}x_{14}\rangle.$$

Finally, we use the first relation to substitute x_9 with $x_{14}x_1^{-1}$ and remove this relation, which yields:

$$\langle x_1, x_{14} \mid x_1x_1x_{14}^{-1}x_{14}^{-1}x_{14}^{-1}x_1x_1, x_1x_{14}x_1^{-1}x_{14}^{-1}x_1^{-1}x_{14}\rangle.$$

Setting $x = x_1$ and $y = x_{14}$ and cyclically permuting the relators, we obtain

$$\langle x, y \mid xyx = yxy, x^3 = y^4\rangle,$$

which is precisely AK(3). This completes the proof of Theorem 4.12 for the case $n = 3$.

4.7 Algorithmic 3-Deformations of Fake Surfaces

To prove the stable Andrews–Curtis conjecture, one would need to show that all contractible fake surfaces satisfy the conjecture. However, we are also interested in ruling out specific potential counterexamples to the conjecture. For this purpose, we can attempt to 3-deform specific fake surfaces, even if this does not lead to a general result for all surfaces of a given type or complexity. This approach would be especially valuable for the AK(3) fake surface constructed in Section 4.6, which corresponds to the minimal-length potential counterexample to the (stable) Andrews–Curtis conjecture [18]. In this section, we discuss computational approaches toward this goal.

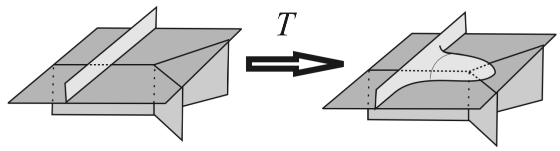


Figure 4.27: The T Move

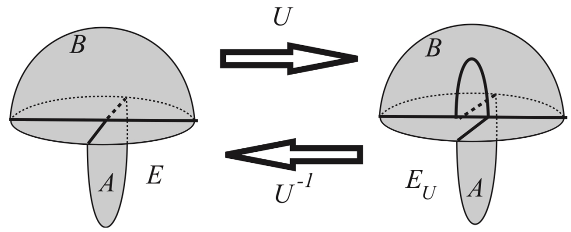


Figure 4.28: The U Move

4.7.1 T and U Moves

In [30], Matveev introduces four moves, called $T^{\pm 1}$ and $U^{\pm 1}$, which generate the entire space of 3-deformations. The T move and its inverse are shown in Figure 4.27, and the U move and its inverse are shown in Figure 4.28.

Both the T move and the U move increase the number of vertices by 1. After applying a U move, the new 2-cell will have a nontrivial T-bundle, so the resulting fake surface is a non-spine.

Remark

As discussed in Section 4.4.1, this is one way in which any fake surface can be 3-deformed into a non-spine fake surface.

Theorem 4.13 ([30], Theorem 1.3.8)

Let K, L be fake surfaces. Then K is 3-deformation equivalent to L if and only if one can transform K to L by a sequence of the moves $T^{\pm 1}, U^{\pm 1}$.

Since such a small set of moves generates 3-deformation equivalence classes, it is natural to ask whether we can extend the computational approach from Chapter 2 to this setting. In particular, we would like to investigate whether a computer search can prove that the fake surface given in Section 4.6 satisfies the stable Andrews–Curtis conjecture, either by 3-deforming to a spine or to a fake surface of complexity at most 5 (which would be sufficient by Theorem 4.7).

However, a naive computer search quickly becomes intractable due to the large number of possible $T^{\pm 1}$ and $U^{\pm 1}$ moves at each step, leading to an exponentially growing search space. In particular, at each vertex we can apply $\binom{4}{2} = 6$ distinct U moves, as we can apply it along any pair of edges at the vertex, and for each edge that is not a self-loop, there are 6 distinct T moves. Thus, for a complexity n fake surface, there may be as many as $6n + 6(2n) = 18n$ moves.

4.7.2 Embedded Disk Moves

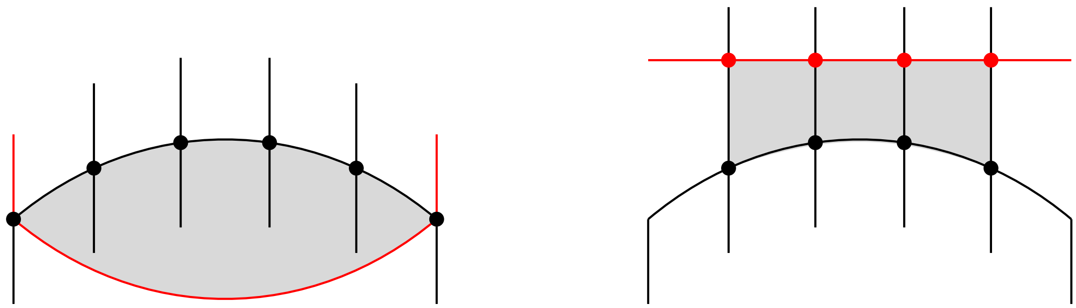
We can also consider the embedded disk moves presented in Sections 4.5.1, 4.5.3, and 4.5.4, which are a subset of the $T^{\pm 1}, U^{\pm 1}$ moves. These moves can be considered in general for embedded disks with any number of vertices. The general form for the trivial embedded disk move is shown in Figure 4.29, and the general form for the nontrivial embedded disk move is shown in Figure 4.30. These moves give us an alternative space of actions by which to 3-deform fake surfaces. Here, the branching factor is smaller, as the moves can only be applied to an embedded disk.

Remark

As mentioned in Section 4.5.1, the trivial embedded disk move in the case of 3 vertices is the T^{-1} move, and the nontrivial embedded disk move in the case of 2 vertices is the U^{-1} move.

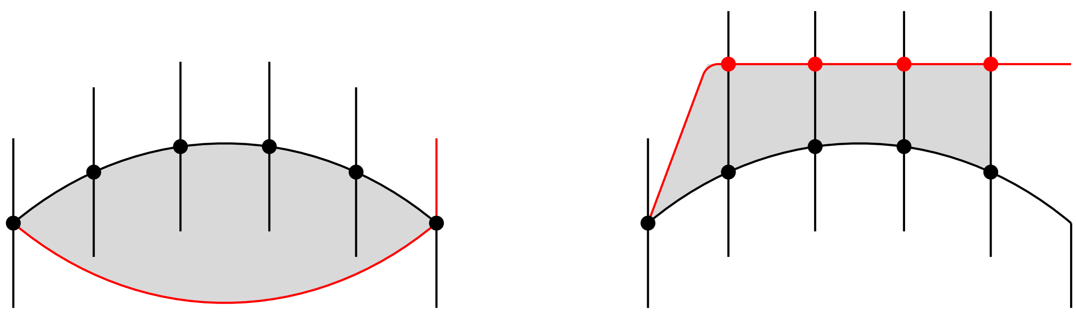
We execute a best-first search over fake surfaces, where we score fake surfaces by complexity and by number of nontrivial T-bundles. We set a fixed number of surfaces T over which to search, and at each stage sample k moves. No search was able to reduce

Figure 4.29: Trivial T-Bundle Embedded Disk Move



The red edge slides up. If the original embedded disk had n vertices, there are $2n - 4$ after the move. In particular, $n-2$ new vertices are created while 2 vertices are removed.

Figure 4.30: Nontrivial T-Bundle Embedded Disk Move



The red edge slides up. If the original embedded disk had n vertices, there are $2n - 3$ after the move. In particular, $n-2$ new vertices are created while one vertex is removed.

the complexity or decrease the number of nontrivial T-bundles to 0.

4.7.3 Computational Implementation

We provide pseudocode for implementing the embedded disk moves in Algorithms 4 and 5 and a search algorithm to explore 3-deformations of fake surfaces in Algorithm 6.

Notes

The induction scheme for establishing new cases of the stable Andrews–Curtis and Zeeman conjectures is joint work with Yang Qiu and Zhenghan Wang [14]. The construction

Algorithm 4 Trivial Embedded Disk Move

function TrivialEDMove(S, D, e)
    Assert that D is a trivial bundle
    Let e be the front edge; list remaining edges as back edges
    Determine all perpendicular edge pairs using adjacency
    Introduce new vertical and horizontal edges
    Construct a replacement dictionary describing local edge substitutions
    Apply all replacements throughout the surface S
    Remove the original disk D
    Insert the new disks created by the move geometry
    Update edge indices and metadata
    return modified surface S
end function

Algorithm 5 Nontrivial Embedded Disk Move

function NontrivialEDMove(S, D, e)
    Assert that D has a nontrivial T-bundle
    Let e be the front edge; list back edges
    Compute perpendicular edge pairs, alternating orientation at each step
    Introduce new vertical, horizontal, and diagonal edges
    Build the replacement dictionary encoding the move
    Apply all replacements across S
    Remove the original disk D
    Insert the structured family of disks defined by the move
    Remove boundary edges and update metadata
    return modified surface S
end function

Algorithm 6 Best-First Search Over Fake Surfaces

function SEARCH(S_0, T, k) $\triangleright$ T = max iterations, k = moves sampled per expansion
    Initialize priority queue Q
    Compute score of S_0 (complexity, number of nontrivial bundles)
    Insert (score, $S_0, \emptyset$) into Q
    Track $S_{\text{best}} \leftarrow S_0$ and its score
    for $t = 1$ to T **do**
        if Q is empty **then**
            return S_{best}
        end if
        Pop best surface ($\_, S, H$) from Q
        Determine all possible moves on S
        Sample up to k moves
        for all sampled moves m **do**
            $S' \leftarrow$ apply trivial or nontrivial embedded disk move m to a copy of S
            if S' is valid **then**
                Compute score of S'
                Insert (score, $S', H + [m]$) into Q
                if score(S') ¡ score(S_{best}) **then**
                    $S_{\text{best}} \leftarrow S'$
                end if
            end if
        end for
    end for
    return S_{best}
end function

of the fake surface for $\mathrm{AK}(n)$ originates from Jon McCammond, and we are developing it further in joint work.

Chapter 5

Classification of Fake Surfaces

In this chapter, we present the classification of acyclic cellular fake surfaces of complexities 1, 2, 3, and 4, along with complexity 5 surfaces that have no disks of boundary length 1 or 2 (which we henceforth call *small disks*). In particular, we present the following:

Theorem 5.1

There are two acyclic cellular fake surfaces of complexity one, 17 of complexity two, 238 of complexity three, and 4618 of complexity four.

In Section 5.1, we present the conventions used in the classification. In Section 5.2 we discuss the classification strategy. We present the results of the classification in Section 5.3, and then in Sections 5.4 and 5.5 we discuss consequences of the classification.

Full data for the classification can be found on GitHub at `https://github.com/lucasfagan/Fake-Surfaces`. Acyclic cellular fake surfaces of complexity 1 are classified in [21], and there are two of them: the abalone (a spine of the 3-ball) and one non-spine.

5.1 Notation and convention

We represent a fake surface F as (Γ_i^j, M) where Γ_i^j is the adjacency matrix of the ith 1-skeleton of complexity j, and M is a matrix representing the 2-cell attaching maps.

Each row of M corresponds to a disk, with entries indicating which edges appear in its boundary cycle. A negative entry denotes traversing the edge in the opposite direction. For readability, we pad shorter attaching maps with 0s to match the length of the longest disk.

The last two columns of M indicate whether each disk is embedded and whether it has a trivial T-bundle, respectively.

Our notation for fake surfaces relies on ordering the vertices and edges. We adopt the following conventions.

To establish a canonical ordering of 1-skeleta (viewed as 4-regular multigraphs), we use the following map: let $\mathsf{D} : M_n(\mathbb{Z}) \to \mathbb{Z}$ be the function that gives a decimal representation of a matrix by concatenating the rows from top to bottom into an integer with n^2 digits. For example:

$$\mathsf{D}\left(\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}\right) = 1234.$$

Explicitly, if $A = (a_{i,j})$, then $\mathsf{D}(A) = \sum_{i,j=1}^{n} 10^{n(n-i)+n-j} a_{i,j}$ as an integer.

For a given 1-skeleton of complexity n, we select the ordering of vertices that maximizes $\mathsf{D}(A)$ for the adjacency matrix $A = (a_{i,j})$. This provides a canonical labeling for each 1-skeleton and a canonical ordering of all 1-skeleta of a given complexity: from largest to smallest value of $\mathsf{D}(A)$.

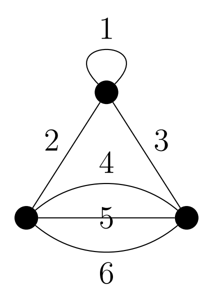
We then label the edges in the order they appear in the upper-right triangle of the chosen adjacency matrix, read left-to-right and top-to-bottom. To be precise, using our ordering of the vertices, let (i_1, j_1) and (i_2, j_2) be edges satisfying $i_1 \leq j_1$ and $i_2 \leq j_2$. Then $(i_1, j_1) < (i_2, j_2)$ if $i_1 < i_2$, or if $i_1 = i_2$ and $j_1 < j_2$. We label edges from 1 to $2n$ according to this ordering. When multiple edges exist between the same pair of vertices, their relative order can be chosen arbitrarily.

We provide an example to make this clear. Consider the graph Γ given by the adjacency matrix

$$A = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 0 & 3 \\ 1 & 3 & 0 \end{bmatrix}.$$

Observe that this ordering of the vertices maximizes $\mathsf{D}(A)$. Following our convention, the self-loop at the first vertex is edge 1, the two other edges incident to that vertex are edges 2 and 3, and the three edges connecting the other two vertices are edges 4, 5, and 6. There are 12 valid edge labelings satisfying these constraints (corresponding to the $2 \times 3! = 12$ choices for ordering the multiple edges). One such labeling is shown in Fig. 5.1:

Figure 5.1: Valid Edge Labeling for Γ



Recall that a fake surface is *acyclic* if it is homologous to a point and *contractible* if it is homotopic to a point.

Any acyclic cellular fake surface with connected 1-skeleton and t type 2 singularities has exactly $t + 1$ disks and $2t$ edges.

5.2 Classification strategy

5.2.1 Classification of 4-regular multigraphs

We first enumerate all connected 4-regular multigraphs that can serve as 1-skeleta. This is accomplished by enumerating all valid adjacency matrices. Tables 5.1 and 5.2 provide combinatorial data on the 1-skeleta for each complexity.

Table 5.1: 1-Skeleta by Complexity and Number of Self-Loops

Complexity	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Total**
1	0	0	1						1
2	1	0	1						2
3	1	1	1	1					4
4	3	2	3	1	1				10
5	6	7	7	5	2	1			28
6	19	21	28	16	10	2	1		97
7	50	85	98	72	36	14	3	1	359

Table 5.2: 1-Skeleta by Length of Shortest Cycle

Complexity	**1**	**2**	**3**	**Total**
1	1	0	0	1
2	1	1	0	2
3	3	1	0	4
4	7	3	0	10
5	22	5	1	28
6	78	18	1	97
7	309	48	2	359

5.2.2 Creating Exhaustive List of Fake Surfaces

We classify fake surfaces for each 1-skeleton separately. Given a 1-skeleton with oriented, labeled edges, we first enumerate all possible ways to attach oriented disks to the 1-skeleton subject to the constraint that each vertex has a type 2 singularity. Next, we enumerate all combinations of disks that may form a fake surface and verify that each vertex indeed has a type 2 singularity. If this condition is satisfied, we check that the surface is acyclic by collapsing a maximal tree of the 1-skeleton and verifying that the determinant of the resulting boundary map $\mathbb{Z}^{n+1} \to \mathbb{Z}^{2n}$ equals ± 1. We then store these acyclic fake surfaces for further analysis.

5.2.3 Removing Redundancy

The final step is to remove redundancy from our list of fake surfaces. We classify fake surfaces up to PL homeomorphism, which corresponds to the equivalence relation generated by finite sequences of the following combinatorial operations:

1. Relabel the edges of the graph (i.e., apply a graph automorphism);
2. Change the orientation of an edge (i.e., negate all appearances of that edge label);
3. Change the orientation of a disk (i.e., reverse the cyclic order of edges in its boundary and negate each edge label);
4. Cyclically rotate the boundary cycle of a disk.

Remark

Two cellular fake surfaces are connected by a finite sequence of the above operations if and only if they are PL homeomorphic to each other. That is, after endowing both surfaces with triangulations, the two simplicial complexes become simplicially isomorphic after subdivision [41]. In one direction, a sequence of these operations induces a homeomorphism between two fake surfaces that maps vertices to vertices, edges to edges, and 2-cells to 2-cells. Conversely, given triangulations on two PL homeomorphic fake surfaces, we can subdivide their 1-skeletons so that they have the same simplicial structure, and then extend the isomorphism to 2-cells (which are topologically open disks).

We then output a complete list of distinct surfaces, with one representative from each equivalence class. The code for this classification can be found at `https://github.com/lucasfagan/Fake-Surfaces`.

5.3 Fake Surfaces up to Complexity 5

5.3.1 Complexity 1

There is a unique 1-skeleton for complexity 1, which is given by $\mathbf{\Gamma^1} = (\Gamma^1_1) = (\begin{bmatrix} 4 \end{bmatrix})$ and shown in Fig. 5.2.

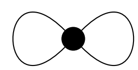


Figure 5.2: 1-Skeleton for Complexity 1

There are two acyclic fake surfaces of complexity 1, first classified in [21]. We reproduce them here using the notation introduced in Section 5.1.

$$\left(\Gamma^1_1, \left[\begin{array}{ccccc|cc} 1 & 2 & 2 & -1 & -2 & N & N \\ 1 & 0 & 0 & 0 & 0 & Y & Y \end{array}\right]\right)$$

$$\left(\Gamma^1_1, \left[\begin{array}{ccccc|cc} 1 & 2 & 2 & 1 & -2 & N & Y \\ 1 & 0 & 0 & 0 & 0 & Y & Y \end{array}\right]\right)$$

The first is a non-spine, since the top disk has a nontrivial T-bundle, while the second is a spine of the 3-ball known as the *abalone*.

Note that at a vertex with no self-loops, edge orientations are determined by the cyclic ordering around the vertex. However, these two surfaces demonstrate that edge orientations for self-loops are essential: using unoriented edges in the attaching maps would fail to distinguish between these two non-homeomorphic surfaces.

5.3.2 Complexity 2

For complexity 2, there are two 1-skeleta, ordered according to our convention in the following way and shown in Fig. 5.3.

$$\mathbf{\Gamma^2} = (\Gamma_1^2, \Gamma_2^2) = \left(\begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}, \begin{bmatrix} 0 & 4 \\ 4 & 0 \end{bmatrix} \right)$$

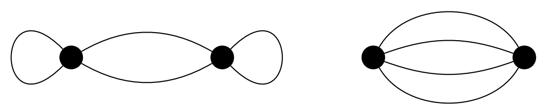


Figure 5.3: 1-Skeleta for Complexity 2

One well-known example of a complexity 2 fake surface is *Bing's house*, also called the *house with two rooms*. In our notation, this surface is presented as follows:

$$\left(\Gamma_1^2, \left[\begin{array}{cccccccccc|cc} 3 & -1 & -3 & 2 & -1 & -2 & -4 & 2 & -3 & -4 & N & Y \\ 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Y & Y \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Y & Y \end{array} \right] \right).$$

There is another spine of the 3-ball, called the *mutant* of the house with two rooms in [30], which once again demonstrates the importance of edge orientations:

$$\left(\Gamma_1^2, \left[\begin{array}{cccccccccc|cc} 3 & -1 & -3 & 2 & 1 & -2 & 4 & 2 & -3 & -4 & N & Y \\ 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Y & Y \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Y & Y \end{array} \right] \right).$$

Note that without edge orientations, these two distinct surfaces would be indistinguishable in our notation.

There are 17 acyclic cellular fake surfaces of complexity 2 in total: 15 arising from the first 1-skeleton and 2 from the second. All remaining acyclic cellular fake surfaces of complexity 2 (beyond the examples shown above) can be found in Appendix B.

5.3.3 Complexity 3

For complexity 3, there are four 1-skeleta, presented and ordered according to our convention in the following way and shown in Fig. 5.4.

$$\mathbf{\Gamma^3} = (\Gamma^3_1, \Gamma^3_2, \Gamma^3_3, \Gamma^3_4) = \left(\begin{bmatrix} 2 & 2 & 0 \\ 2 & 0 & 2 \\ 0 & 2 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 1 \\ 1 & 0 & 3 \\ 1 & 3 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 2 & 2 \\ 2 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix} \right)$$

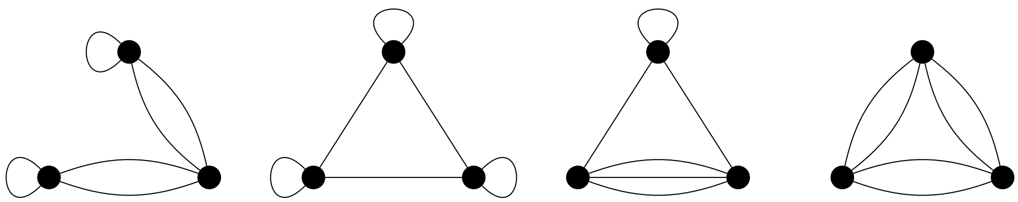


Figure 5.4: 1-Skeleta for Complexity 3

There are 238 acyclic cellular fake surfaces of complexity 3, with 14, 65, 97, and 62 arising from the first, second, third, and fourth 1-skeleton, respectively. All are listed at `https://github.com/lucasfagan/Fake-Surfaces`.

5.3.4 Complexity 4

For complexity 4, there are 10 1-skeleta, presented and ordered according to our convention in the following way and shown in Fig. 5.5.

$$\mathbf{\Gamma^4} = \left(\begin{bmatrix} 2 & 2 & 0 & 0 \\ 2 & 0 & 2 & 0 \\ 0 & 2 & 0 & 2 \\ 0 & 0 & 2 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 2 & 0 & 0 \\ 2 & 0 & 1 & 1 \\ 0 & 1 & 2 & 1 \\ 0 & 1 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 2 & 0 & 0 \\ 2 & 0 & 1 & 1 \\ 0 & 1 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \\ 1 & 0 & 2 & 1 \\ 0 & 1 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \\ 1 & 0 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{bmatrix}, \right.$$
$$\left. \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 0 & 1 & 2 \\ 1 & 1 & 0 & 2 \\ 0 & 2 & 2 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 3 & 1 & 0 \\ 3 & 0 & 0 & 1 \\ 1 & 0 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 2 & 2 & 0 \\ 2 & 0 & 0 & 2 \\ 2 & 0 & 0 & 2 \\ 0 & 2 & 2 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 2 & 1 & 1 \\ 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 2 \\ 1 & 1 & 2 & 0 \end{bmatrix} \right)$$

There are 4618 acyclic cellular fake surfaces of complexity 4. The 1-skeleton with the largest number of surfaces is Γ^4_2 with 1171. The one with the fewest is Γ^4_9 with 35. All of these are listed at `https://github.com/lucasfagan/Fake-Surfaces`.

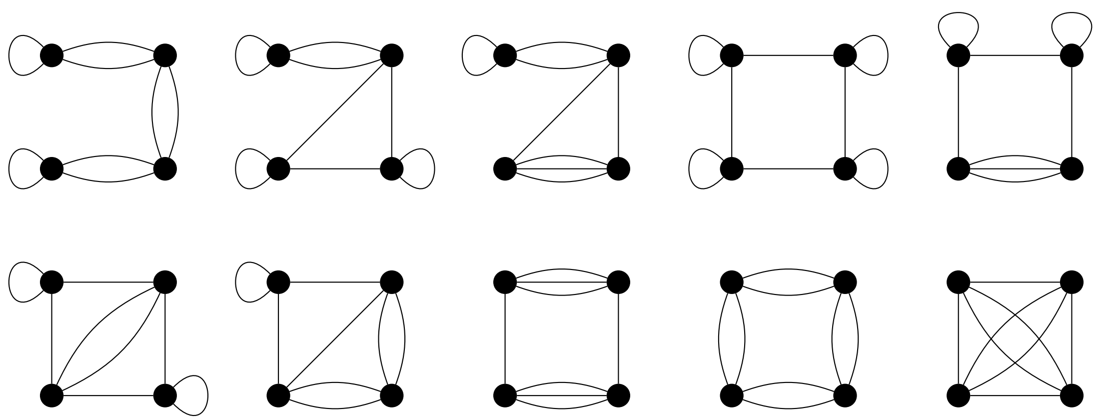


Figure 5.5: 1-Skeleta for Complexity 4

5.3.5 Complexity 5

For complexity 5, there are 28 1-skeleta. We do not list the adjacency matrices, but the list can be found at `https://github.com/lucasfagan/Fake-Surfaces`. They are shown in order in Fig. 5.6.

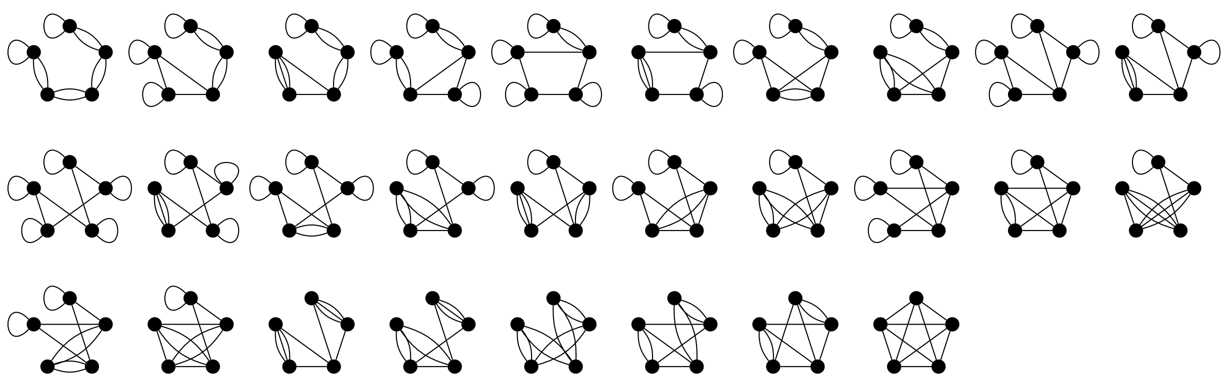


Figure 5.6: 1-Skeleta for Complexity 5

Notably, complexity 5 is the first at which there exists a 1-skeleton that is a simple graph (i.e., having neither multiple edges nor self-loops): the complete graph K_5.

Due to computational limitations, we classified only those complexity 5 surfaces without small disks. Since disks of boundary length 1 or 2 are necessarily embedded when the complexity is greater than 1, this restriction nonetheless allowed us to verify that all complexity 5 surfaces possess embedded disks. All acyclic cellular fake surfaces of complexity

5 without small disks are listed at `https://github.com/lucasfagan/Fake-Surfaces`.

5.4 Spine and Non-Spine Fake Surfaces

As mentioned in Section 4.3, fake surfaces can be divided into spines and non-spines, depending on whether they can be realized as spines of 3-manifolds. We recall that this dichotomy is unique to dimension 2; in higher dimensions, all generic n-polyhedra appear as spines of $(n+1)$-manifolds. The numbers of contractible fake surfaces $\{C_n\}$ of complexity n form an interesting sequence that has not appeared elsewhere [37]. Table 5.3 reveals that as complexity increases, the proportion of surfaces that are spines decreases. This empirical trend provides evidence for the following conjecture:

Table 5.3: Number of Contractible Fake Surfaces by Complexity and Count of Disks with Nontrivial T-Bundles

Complexity	**Spines**	**Non-spines**					**Total**	**% Spines**
	0	**1**	**2**	**3**	**4**	**5**		
1	1	1	0				2	50
2	3	6	6	2			17	16.7
3	20	54	89	62	13		238	8.4
4	128	607	1450	1533	745	155	4618	2.8

Conjecture 5.2

The proportion of spines among contractible fake surfaces of complexity n goes to 0 as $n \to \infty$.

5.5 Contractibility and Embedded Disk Conjectures

5.5.1 Contractibility conjecture

Ikeda proved that any acyclic cellular fake surface that is a spine and has complexity at most 2 is contractible. Robertson [40] showed that any acyclic cellular fake surface of complexity at most 3 is contractible, and conjectured that the same holds for complexity 4. In this section, we prove this conjecture. This is the best possible general result, as the dodecahedral Poincaré homology sphere has a spine of complexity 5 that is acyclic but not contractible.

Theorem 5.3

All acyclic cellular fake surfaces of complexity 1, 2, 3, and 4 are contractible.

Proof. The proof is by explicit computer verification using our classification. The code to verify contractibility is provided in Appendix A and at `https://github.com/lucasfagan/Fake-Surfaces`. For each acyclic cellular fake surface of complexity 1 through 4, we verified that its fundamental group is trivial, which combined with acyclicity implies contractibility. □

5.5.2 Embedded Disk Conjecture

Embedded disks play a crucial role in proving the stable Andrews–Curtis conjecture and the Zeeman conjecture. For contractible fake surfaces of complexity 1, Ikeda [21] proved the these conjectures using the existence of embedded disks. In Section 4.4, we extend this proof to contractible fake surfaces of complexity up to 5. Our proof relies on the following result:

Theorem 5.4

All contractible fake surfaces up to complexity 5 have at least one embedded disk.

Proof. For complexity up to 4, we verify this explicitly using our complete classification. For complexity 5, we do not have a complete enumeration of all acyclic cellular fake surfaces. However, we do have a complete list of all acyclic cellular fake surfaces without small disks (i.e., disks of boundary length 1 or 2). Since small disks are always embedded when the complexity is greater than 1, all surfaces in our classification have embedded disks. This establishes the theorem for all contractible fake surfaces of complexity up to 5. □

We conjecture that all contractible fake surfaces have embedded disks. The embedded disks can be thought of as corresponding to the weak parts of the resulting presentations of the trivial group.

Notes

The classification of contractible fake surfaces up to complexity 5 is from joint work with Yang Qiu and Zhenghan Wang [13].

Chapter 6

Future Work

6.1 Add Stabilization Moves to Greedy Search

The computational work in Chapter 2 admits several natural extensions. In practice, we regard trivializing a potential counterexample using AC-moves alone as essentially as significant as doing so with stabilization moves; thus, it is natural to include stabilization moves in the action space. However, this enlarges the branching factor of the search and consequently slows down the computation. One possible middle ground is to allow a compound move that first adds a generator using Lemma 3.2 in reverse and then immediately removes a generator using Lemma 3.2 again. For example, consider the following move: let $P = \langle x, y \mid r_1, w_1w_2xw_3w_4\rangle$ be such that w_2 and w_3 contain no x. Using Lemma 3.2, set $z = w_4w_1$ to obtain $\langle x, y, z \mid r_1, zw_2xw_3, z = w_1w_4\rangle$. We then apply Lemma 3.2 again to remove x. In the worst case, this compound move is equivalent to performing several substitution moves. In the best case, it may take us to a presentation not AC-equivalent to P.

6.2 Use Automorphism Priority in Greedy Search

Another extension of the work in Chapter 2 is to prioritize presentations not by their length, but by their minimum length achievable under automorphisms of the underlying free group, essentially making the search equivariant with respect to automorphisms. This quantity is straightforward to compute using Whitehead's algorithm (see [27], Proposition 4.20). This approach is valuable because, as noted in Section 2.1, finding a presentation that can be taken to $\langle x, y \mid x, y \rangle$ by an automorphism already suffices to trivialize it. The hope is that some presentations requiring significant length increases under ordinary search might be more easily trivialized in this framework. We have some evidence for this: for instance, greedy search fails to trivialize $\langle x, y \mid x^{-2}y^{-1}x^{-1}y^{-1}xy^{-1}, x^{-1}y^{-1}x^{-1}y^{2}xy^{-3} \rangle$ in 1M nodes, but when prioritizing by minimum automorphic length, the presentation is trivialized in only 168 nodes. This is particularly striking because the "terminal" presentation (an automorphic image of the trivial presentation) has length 141, illustrating how dramatically this search diverges from standard greedy search, which never visits a presentation above length 33 in 1M nodes. We hope to expand and investigate this idea more thoroughly in future work.

6.3 Minimal List of Potential Counterexamples in MS(n, w)

It would be valuable to minimize the number of distinct AC-equivalence classes in the MS(n, w) dataset discussed in Section 2.3.2. In particular, many of these presentations are AC-trivial, and presentations from different families sometimes turn out to be AC-equivalent. Obtaining a minimal list of potential counterexamples distilled from the original 1190 presentations would be a useful contribution. Understanding the structure of this reduced set would provide valuable insight into the landscape of potential counterexamples and might reveal patterns or families that warrant deeper theoretical investigation.

6.4 Study New Presentations from AK(3) Fake Surface

In Section 4.6, we show that a specific fake surface represents AK(3) by collapsing a maximal tree, reading off the resulting presentation, and proving that this presentation is stably AC-equivalent to AK(3). It would be worthwhile to perform a comprehensive search over all maximal trees in this fake surface to determine whether any of the resulting presentations can be trivialized by our solver from Chapter 2. Each presentation obtained in this way is stably AC-equivalent to AK(3), and if AK(3) is stably AC-trivial, it is plausible that some of these presentations may be easier to trivialize than others.

6.5 Study Impact of T and U Moves on Group Presentations

As noted in Section 4.7.1, Matveev's $T^{\pm 1}$ and $U^{\pm 1}$ moves generate the entire 3-deformation equivalence class. It would be worthwhile to study how these moves affect the presentations obtained by collapsing a maximal tree, and to determine whether this yields insight or suggests new moves that could be incorporated into the search algorithms in Chapter 2.

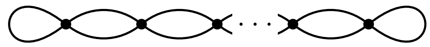
6.6 Improve Embedded Disk Move Search

We hope to improve the fake surface 3-deformation search described in Section 4.7 by developing more sophisticated strategies for choosing where to apply moves. Currently, our algorithm exhaustively applies all possible moves at each step, but this approach becomes computationally expensive as complexity increases. It would be valuable to develop heuristics that prioritize moves more likely to lead to collapses, perhaps by identifying structural features such as embedded disks with small boundary length or disks whose removal would simplify the 1-skeleton. Additionally, we could explore incorporating machine learning techniques similar to those discussed in Section 2.2.3 for the Andrews–Curtis conjecture, training models to recognize promising intermediate fake surfaces along successful 3-deformation paths. Another direction is to study sequences of moves rather than individual moves, identifying compound operations that reliably reduce complexity. We might also investigate whether the relationship between presentations and fake surfaces, discussed in Section 3.2.2, could inform the search: presentations that are easier to trivialize computationally might correspond to fake surfaces with more tractable 3-deformation paths, suggesting that we could use computational trivialization difficulty as a guide for exploring the fake surface search space.

6.7 Extend Fake Surface Classification to Higher Complexity

We hope to extend the classification of fake surfaces to complexity 7 and beyond. Beyond generalizing the results of Chapters 4 and 5 to higher complexity, the broader goal is to gain a deeper understanding of the 3-deformation class of the fake surface in Section 4.6, with the eventual aim of proving that it 3-deforms to a point. The primary computational challenge is the exponential growth in the number of fake surfaces as complexity increases: we have seen 2, 17, 238, and 4618 contractible fake surfaces at complexities 1 through 4, respectively, and this trend suggests that complexity 6 and 7 will involve tens or hundreds of thousands of surfaces. To make this feasible, we will need to optimize our enumeration algorithms further. One promising approach is to focus initially on

Figure 6.1: 1-Skeleton for Potential Counterexample to Embedded Disk Conjecture



All embedded disks in this 1-skeleton must have no more than two vertices, no matter the complexity. As the average number of vertices per disk increases as the complexity grows, this 1-skeleton seems like a natural candidate on which to find a counterexample to the embedded disk conjecture.

special subclasses, such as those with highly symmetric 1-skeleta or those satisfying additional constraints like planarity. The classification at higher complexity would allow us to test Conjecture 5.2 about the decreasing proportion of spines, verify whether the embedded disk conjecture continues to hold, and potentially discover new structural patterns that could inform theoretical approaches to the stable Andrews–Curtis and Zeeman conjectures.

6.8 Construct Counterexample to Embedded Disk Conjecture

There are some reasons to believe that the embedded disk conjecture is false. One is purely combinatorial: the 1-skeleton of a complexity n fake surface has $2n$ edges and $n+1$ disks. Each edge appears 3 times across all the attaching maps, so the average number of edges per disk is $6n/(n+1)$, which grows as n increases. However, for any complexity, there exists a 1-skeleton on which all embedded disks must have at most 2 vertices, which is given in Figure 6.1. It seems worthwhile to explore if a counterexample can be generated in this way.

Appendices

Appendix A

Code to Check Contractibility of Acyclic Fake Surface

In order to check if an acyclic fake surface is contractible, it suffices to check that it is simply connected by the Whitehead theorem and the Hurewicz theorem. We provide a Sage code snippet that takes as input a fake surface and a choice of maximal tree, and outputs if the fundamental group is trivial. Of course, deciding if a group presentation is the trivial group is undecideable, but this only leaves open the possibility of false negatives. Sage is able to detect the triviality of the fundamental group of every fake surface through complexity 5, with the lone exception being the known counterexample of the complexity 5 spine of the Poincaré homology sphere.

The code can also be found at `https://github.com/lucasfagan/Fake-Surfaces`.

```
# sample data from complexity 2 to demonstrate formatting
acyclic_surface = [[4,2,-1,-1,-2,3,2,1,-3],[2,-3],[4]]
maximal_tree = [2]

def is_contractible(acyclic_surface, maximal_tree):
    # remove the elements in the maximal tree
    collapsed_surface = [[edge for edge in disk if (edge not in maximal_tree and -1*edge not in maximal_tree)] for disk in
```

```
acyclic_surface]
    # for conveinence, reduce the remaining edges to be labeled with 1 through n+1
    for removed_edge in sorted(maximal_tree, reverse=True):
        collapsed_surface = [[edge-1 if edge>removed_edge else edge for edge in disk] for disk in collapsed_surface]
        collapsed_surface = [[edge+1 if edge<-1*removed_edge else edge for edge in disk] for disk in collapsed_surface]

    # create a free group with one generator for each remaining edge
    F = FreeGroup(len(maximal_tree)+2,'a')

    group_relations = []

    # convert the disks to relations
    for disk in collapsed_surface:
        relation = F.one()  # Initialize with identity
        for edge in disk:
            if edge < 0:
                relation *= (F.gen(-edge-1))**-1  # Inverse of generator
            else:
                relation *= F.gen(edge-1)  # Multiply with generator

        # Append the relation
        group_relations.append(relation)

    # Create quotient group
    G = F.quotient(group_relations)
    # Simplify the group
    G = G.simplified()

    # Check if group is trivial
```

```
    if G.order()!=1:
        return False
    return True

print(is_contractible(acyclic_surface, maximal_tree))
```

Saved as a Sage file, we can simply run this with the command

```
sage filename.sage
```

Appendix B

Fake Surfaces of Complexity 2 and 3

All acyclic cellular fake surfaces of complexity 2 and 3 are listed below. The rest of the data can be found at `https://github.com/lucasfagan/Fake-Surfaces`.

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{42-1-1-2431-3} & NN \\ \text{2-3} & YN \\ \text{4} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{42-1-231-3-42-3} & NY \\ \text{4} & YN \\ \text{-1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{43-1-2-42-3} & NY \\ \text{2-1-1-3} & NN \\ \text{4} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{43-1-1-2-431-2} & NY \\ \text{2-3} & YN \\ \text{4} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{3-23-1-3-42-1-2-4} & NN \\ \text{4} & YN \\ \text{-1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{21-3-42-1-1-3} & NN \\ \text{42-3} & NY \\ \text{4} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{43-2-42-1-231-3} & NY \\ \text{4} & YY \\ \text{-1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{42-3431-32-1-2} & NN \\ \text{4} & YY \\ \text{-1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{443-1-2-42-3} & NY \\ \text{2-1-3} & NY \\ \text{-1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{431-23-1-242-3} & NN \\ \text{4} & YY \\ \text{-1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{442-32-1-2} & NN \\ \text{43-1-3} & NN \\ \text{-1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^2, \left[\begin{array}{cc} \text{4-31-42-41-3} & NN \\ \text{1-2} & YN \\ \text{2-3} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{42-1-3-43-23-1-2} & NY \\ \text{4} & YY \\ \text{-1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{3-1-3-42-1-2} & NY \\ \text{442-3} & NY \\ \text{-1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^2, \left[\begin{array}{cc} \text{4-12-43-41-2} & NY \\ \text{2-3} & YN \\ \text{1-3} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{3-1-32-1-2-42-3-4} & NY \\ \text{4} & YY \\ \text{-1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^2, \left[\begin{array}{c|c} \text{43-1-1-32-3} & NY \\ \text{2-1-2-4} & NN \\ \text{4} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-3-5-6-652} & NN \\ \text{2-3-4-64} & NY \\ \text{1-23} & NN \\ \text{4-5} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-3-5-6-642} & NY \\ \text{1-2-543} & NY \\ \text{4-56} & NY \\ \text{2-3} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-3-54-5-652} & NY \\ \text{2-3-4-64} & NN \\ \text{1-23} & NN \\ \text{6} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4643-1-32} & NN \\ \text{2-3-5665} & NY \\ \text{4-5} & YN \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-5642-3-452} & NN \\ \text{4-5-6-6} & NY \\ \text{1-23} & NY \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4-6-652-32} & NN \\ \text{1-3-543} & NN \\ \text{4-5-6} & NN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-45-4-6-643} & NN \\ \text{1-3-5652} & NN \\ \text{2-3} & YN \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4-64-56653} & NY \\ \text{2-3-45} & NN \\ \text{1-32} & NN \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-54-56653} & NY \\ \text{1-3-4642} & NN \\ \text{2-3} & YN \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-565-4643} & NN \\ \text{11-2-542} & NN \\ \text{2-3} & YY \\ \text{6} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-564-5-6-643} & NN \\ \text{2-3-54} & NN \\ \text{1-32} & NN \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5-65-4-643} & NY \\ \text{11-3-542} & NY \\ \text{2-3} & YY \\ \text{6} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5-642-321-3-4-653} & NY \\ \text{4-5} & YN \\ \text{6} & YN \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4652-3-543} & NN \\ \text{11-32} & NN \\ \text{4-56} & NY \\ \text{6} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5-653-1-32} & NN \\ \text{2-3-4664} & NY \\ \text{4-5} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5653-23-1-3-4-642} & NN \\ \text{4-5} & YY \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-54-56652} & NY \\ \text{2-3-4-64} & NY \\ \text{1-23} & NY \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4-64-5653} & NY \\ \text{11-3-452} & NN \\ \text{2-3} & YY \\ \text{6} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-321-2-5653-2-4-643} & NY \\ \text{4-5} & YN \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{2-3-4-65-4665} & NY \\ \text{1-3-542} & NY \\ \text{1-23} & NN \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-565-4643} & NN \\ \text{11-2-452} & NN \\ \text{2-3} & YN \\ \text{6} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-5431-2-565-4-642} & NY \\ \text{2-3} & YY \\ \text{6} & YY \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-23-1-3-542} & NN \\ \text{2-3-465} & NY \\ \text{4-56} & NN \\ \text{6} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-46653-23} & NN \\ \text{1-2-5642} & NN \\ \text{4-5} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4643-2-5-653-1-32} & NN \\ \text{4-5} & YN \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-32-3-5-643} & NN \\ \text{1-2-542} & NY \\ \text{4-56} & NY \\ \text{6} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-3-464-542} & NY \\ \text{1-2-5-653} & NN \\ \text{2-3} & YN \\ \text{6} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5421-3-4-65-4653} & NY \\ \text{2-3} & YN \\ \text{6} & YN \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4521-3-565-4643} & NN \\ \text{2-3} & YN \\ \text{6} & YY \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4531-3-4-64-5-652} & NY \\ \text{2-3} & YY \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-23-1-3-5-642} & NY \\ \text{2-3-4-65} & NY \\ \text{4-5} & YY \\ \text{6} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5-64-5643-1-2-453} & NN \\ \text{2-3} & YN \\ \text{6} & YN \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5-653-1-2-45-4643} & NY \\ \text{2-3} & YY \\ \text{6} & YY \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-2-5-643-1-23} & NN \\ \text{2-3-564} & NY \\ \text{4-5} & YY \\ \text{6} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-4531-2-464-5652} & NN \\ \text{2-3} & YY \\ \text{6} & YY \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-32-3-46531-2-4-652} & NN \\ \text{4-5} & YN \\ \text{6} & YN \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-321-2-543} & NN \\ \text{2-3-565-4-64} & NY \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-543-1-2-464-5653} & NY \\ \text{2-3} & YN \\ \text{6} & YY \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-464-5431-3-5652} & NY \\ \text{2-3} & YN \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{2-3-565-464} & NN \\ \text{1-23-1-2-543} & NY \\ \text{6} & YY \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-565-453-1-2-4643} & NN \\ \text{2-3} & YY \\ \text{6} & YY \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-23-1-3-4-642-3-5-652} & NN \\ \text{4-5} & YY \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-543-1-32} & NN \\ \text{2-3-565-4-64} & NN \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-4642-321-2-5-653} & NY \\ \text{4-5} & YY \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-32-1-2-4643} & NN \\ \text{2-3-565} & NY \\ \text{4-5} & YN \\ \text{6} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-23-1-3-542} & NN \\ \text{2-3-4-65-465} & NN \\ \text{6} & YN \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-46431-3-5-65-452} & NN \\ \text{2-3} & YN \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-23-1-2-5653} & NY \\ \text{2-3-464} & NY \\ \text{4-5} & YN \\ \text{6} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-321-2-453} & NY \\ \text{2-3-464-565} & NY \\ \text{6} & YY \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4652-3-5643-1-32} & NY \\ \text{4-5} & YN \\ \text{6} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-23-1-3-5652} & NN \\ \text{2-3-464} & NY \\ \text{4-5} & YY \\ \text{6} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5-64-542-3-4-653} & NN \\ \text{1-32} & NY \\ \text{6} & YN \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5642-3-5-643-1-32} & NN \\ \text{4-5} & YN \\ \text{6} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4-6-65-4652} & NY \\ \text{1-3-453} & NY \\ \text{2-3} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-54-5-652-3-4642} & NN \\ \text{1-23} & NN \\ \text{6} & YY \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}3\text{-}4\text{-}642\text{-}3\text{-}54\text{-}5652 & NN \\ 1\text{-}23 & NN \\ 6 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}3\text{-}4\text{-}64\text{-}5653\text{-}2\text{-}452 & NN \\ 1\text{-}23 & NN \\ 6 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}5\text{-}64\text{-}5642\text{-}32 & NN \\ 1\text{-}3\text{-}543 & NY \\ 6 & YN \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}321\text{-}2\text{-}5\text{-}642\text{-}3\text{-}543 & NN \\ 4\text{-}56 & NY \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}321\text{-}2\text{-}542\text{-}3\text{-}4653 & NY \\ 4\text{-}56 & NY \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}23\text{-}2\text{-}4\text{-}64\text{-}5652 & NY \\ 1\text{-}3\text{-}543 & NY \\ 6 & YY \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}4521\text{-}32\text{-}3\text{-}4\text{-}653 & NY \\ 4\text{-}5\text{-}6 & NY \\ 6 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}465\text{-}4\text{-}652\text{-}3\text{-}543 & NN \\ 1\text{-}32 & NY \\ 6 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}23\text{-}1\text{-}3\text{-}54\text{-}5\text{-}652 & NN \\ 2\text{-}3\text{-}4\text{-}64 & NY \\ 6 & YY \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}5\text{-}64\text{-}543\text{-}2\text{-}4\text{-}653 & NY \\ 1\text{-}32 & NY \\ 6 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}23\text{-}1\text{-}2\text{-}4652\text{-}3\text{-}453 & NY \\ 4\text{-}56 & NY \\ 6 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}32\text{-}3\text{-}4\text{-}64\text{-}5\text{-}653 & NY \\ 1\text{-}2\text{-}542 & NN \\ 6 & YY \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}5431\text{-}32\text{-}3\text{-}5\text{-}642 & NY \\ 4\text{-}56 & NN \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}542\text{-}3\text{-}5\text{-}65\text{-}4\text{-}643 & NY \\ 1\text{-}32 & NN \\ 6 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 2\text{-}3\text{-}5\text{-}65\text{-}464 & NY \\ 11\text{-}3\text{-}452 & NN \\ 1\text{-}23 & NY \\ 6 & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}3\text{-}464\text{-}5\text{-}652\text{-}3\text{-}542 & NY \\ 1\text{-}23 & NY \\ 6 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}5\text{-}65\text{-}4\text{-}643 & NY \\ 1\text{-}3\text{-}542\text{-}32 & NN \\ 6 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 2\text{-}3\text{-}54\text{-}5\text{-}6\text{-}65 & NY \\ 1\text{-}3\text{-}4\text{-}642 & NN \\ 1\text{-}23 & NY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}23\text{-}1\text{-}3\text{-}453\text{-}2\text{-}5\text{-}642 & NN \\ 4\text{-}56 & NY \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}5643\text{-}1\text{-}32 & NY \\ 2\text{-}3\text{-}5\text{-}64\text{-}54 & NY \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 11\text{-}23\text{-}2\text{-}452 & NN \\ 1\text{-}3\text{-}5643 & NN \\ 4\text{-}5\text{-}6 & NY \\ 6 & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}23\text{-}2\text{-}46521\text{-}3\text{-}543 & NN \\ 4\text{-}56 & NY \\ 6 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}231\text{-}3\text{-}45\text{-}4642 & NN \\ 2\text{-}3\text{-}565 & NN \\ 6 & YY \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}231\text{-}3\text{-}542 & NY \\ 2\text{-}3\text{-}4\text{-}6\text{-}65 & NY \\ 4\text{-}5\text{-}6 & NY \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}453\text{-}231\text{-}3\text{-}4652 & NY \\ 4\text{-}56 & NN \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}23\text{-}2\text{-}4\text{-}65\text{-}4652 & NY \\ 1\text{-}3\text{-}543 & NY \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} 1\text{-}2\text{-}54\text{-}5\text{-}642\text{-}32 & NN \\ 1\text{-}3\text{-}4\text{-}653 & NN \\ 6 & YN \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-32-3-465-453} & NY \\ \text{1-2-4-652} & NN \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-2-543-1-2-4653} & NN \\ \text{4-56} & NN \\ \text{2-3} & YN \\ 6 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-53-24-565421-3-63} & NN \\ 6 & YY \\ 4 & YN \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-23-2-45-4643} & NY \\ \text{1-3-5-652} & NN \\ 6 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5-6-653-2-4643} & NY \\ \text{1-32} & NN \\ \text{4-5} & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-24-565421-3-63-2-53} & NN \\ 6 & YY \\ 4 & YN \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-32-3-45-4-642} & NN \\ \text{1-2-5-653} & NN \\ 6 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-4531-2-5-642} & NY \\ \text{4-566} & NN \\ \text{2-3} & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2421-3-63-2-5654-53} & NY \\ 6 & YY \\ 4 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-56421-2-5-6-643} & NY \\ \text{2-3} & YY \\ \text{4-5} & YY \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-5-642-3-5643} & NY \\ \text{11-32} & NY \\ \text{4-5} & YN \\ 6 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-42-363-1-354-5-652} & NN \\ 6 & YY \\ 4 & YY \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-56421-3-5-6-643} & NY \\ \text{2-3} & YY \\ \text{4-5} & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-5-6-652} & NN \\ \text{2-3-4-64} & NY \\ \text{11-23} & NN \\ \text{4-5} & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-3631-2-42-35-4-5652} & NN \\ 6 & YY \\ 4 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-3-5652-1-3-4-642} & NY \\ \text{2-3} & YY \\ \text{4-5} & YN \\ 6 & YY \end{array}\right]\right) \quad \left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-2-4-6-653} & NY \\ \text{2-3-465} & NY \\ \text{11-32} & NY \\ \text{4-5} & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-565-4-531-363-242} & NN \\ 6 & YY \\ 4 & YY \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{1-3-46421-2-5-6-653} & NN \\ \text{2-3} & YN \\ \text{4-5} & YY \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-24-5631-3652-3542} & NY \\ 6 & YN \\ 4 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-242-36521-35-4-563} & NN \\ 6 & YN \\ 4 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-3-5652-1-2-4-643} & NN \\ \text{2-3} & YY \\ \text{4-5} & YY \\ 6 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-3-631-242-35-4-5652} & NN \\ 6 & YY \\ 4 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2421-3652-354-563} & NY \\ 6 & YN \\ 4 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_1^3, \left[\begin{array}{c|c} \text{11-2-5-643-1-3-5642} & NN \\ \text{2-3} & YN \\ \text{4-5} & YY \\ 6 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-5-63-1-242-354-563} & NY \\ 6 & YN \\ 4 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-42-363-1-2-5-654-53} & NN \\ 6 & YY \\ 4 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-42-3-631-35-4-5-652} & NY \\ 6 & YY \\ 4 & YY \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-4-531-3-63-24-5-652} & NN \\ 6 & YY \\ 4 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-42-3-6-63-1-2-53} & NY \\ \text{4-565} & NN \\ 4 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-565-4-53-1-363-242} & NY \\ 6 & YY \\ 4 & YY \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-563-2-4-531-365-42} & NY \\ 6 & YN \\ 4 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-53-1-2-42-3-6-63} & NN \\ \text{4-5-65} & NN \\ 4 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-53-1-2-42-3654-563} & NN \\ 6 & YN \\ 4 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-3-65-42-35442} & NY \\ \text{1-2-5-63} & NY \\ 6 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-53-2-4-421-363} & NY \\ \text{4-5-65} & NN \\ 6 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-53-1-2-42-365-4-563} & NY \\ 6 & YN \\ 4 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-421-354-563} & NY \\ \text{2-3665} & NY \\ 4 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-53-24-566542} & NY \\ \text{1-363} & NN \\ 4 & YN \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-242-354-5-63-1-3-652} & NY \\ 6 & YN \\ 4 & YY \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-56542-3-6-63} & NN \\ \text{1-35-42} & NY \\ 4 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-24-5-63-1-3542} & NY \\ \text{2-3-65} & NN \\ 6 & YN \\ 4 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-563-2-421-35-4-5-63} & NY \\ 6 & YN \\ 4 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-3-63-2-566542} & NN \\ \text{1-24-53} & NY \\ 4 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-3-654-5-63-2-53} & NN \\ \text{1-2-42} & NN \\ 6 & YN \\ 4 & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-24-53-1-2-4-5-652-363} & NN \\ 6 & YY \\ 4 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-24-53-1-2-5-63} & NY \\ \text{2-3-654} & NY \\ 6 & YN \\ 4 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-242-35-4-5-652} & NY \\ \text{1-3-63} & NN \\ 6 & YY \\ 4 & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-4-5-6521-363-24-53} & NN \\ 6 & YY \\ 4 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-35-4-42-36542} & NY \\ \text{1-2-563} & NY \\ 6 & YN \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-421-3544-5-63} & NY \\ \text{2-3-65} & NN \\ 6 & YN \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-4-53-1-363-24-5652} & NY \\ 6 & YY \\ 4 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2421-354-563} & NY \\ \text{2-3665} & NN \\ 4 & YY \\ 1 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-2-565-4-53-242} & NN \\ \text{1-3-63} & NN \\ 6 & YY \\ 4 & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-42-36631-352} & NN \\ \text{4-5-65} & NN \\ 4 & YY \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-242-352} & NY \\ \text{1-3-654-5-63} & NN \\ 6 & YN \\ 4 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2442-3652} & NY \\ \text{1-35-4-563} & NY \\ 6 & YN \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-2-4-5-63-1-24-53} & NN \\ \text{2-3-65} & NN \\ 6 & YN \\ 4 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-4-5-65-42} & NN \\ \text{11-352-363} & NN \\ 6 & YY \\ 4 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-354-5-6-652} & NY \\ \text{1-2-42-3-63} & NY \\ 4 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-531-363-2442} & NY \\ \text{4-5-65} & NN \\ 6 & YY \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-2-4-42-352} & NN \\ \text{4-565} & NN \\ \text{1-363} & NN \\ 6 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-363-244-53} & NY \\ \text{1-2-56542} & NY \\ 6 & YY \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-3-652-1-365-42} & NN \\ \text{2-354} & NN \\ 6 & YN \\ 4 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{2-35-4-4-5665} & NY \\ \text{1-2-42} & NN \\ \text{1-363} & NN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-42-3-6-652} & NN \\ \text{1-3654-53} & NY \\ 4 & YY \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-24-5652-3542} & NN \\ \text{1-3-63} & NN \\ 6 & YY \\ 4 & YN \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-354-5-63-2-42} & NN \\ \text{11-2-563} & NN \\ 6 & YN \\ 4 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-5665-4-53} & NN \\ \text{1-363-2-42} & NY \\ 4 & YY \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-35-4-5-63} & NY \\ \text{2-3-6-65} & NY \\ \text{1-242} & NN \\ 4 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-53-1-3-65-42} & NN \\ \text{2-36544} & NN \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-3-63-2-4-4-53} & NY \\ \text{1-24-5-652} & NY \\ 6 & YY \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{2-3544-5-65} & NY \\ \text{11-242} & NN \\ \text{1-3-63} & NN \\ 6 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-24-531-3652} & NN \\ \text{2-3-6544} & NY \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{11-365-42-1-3-6542} & NY \\ \text{2-35} & YY \\ 6 & YN \\ 4 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-3663-242} & NY \\ \text{11-2-53} & NN \\ \text{4-565} & NN \\ 4 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-4-5-631-352} & NN \\ \text{2-3-6-65-4} & NY \\ 4 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-4-565-421-3-6-63} & NN \\ \text{2-35} & YY \\ 4 & YN \\ 1 & YN \end{array}\right]\right)$$

$$\left(\Gamma_2^3, \left[\begin{array}{c|c} \text{2-3544-565} & NN \\ \text{11-2-42} & NN \\ \text{1-363} & NN \\ 6 & YY \end{array}\right]\right) \quad \left(\Gamma_2^3, \left[\begin{array}{c|c} \text{1-2-563-2-4-42} & NN \\ \text{1-35-4-5-63} & NY \\ 6 & YN \\ 1 & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{12-5-32-64-54-2} & NY \\ \text{34-6-3-1} & NN \\ \text{5-6} & YN \\ 1 & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{135-64-6-32-4-3} & NY \\ \text{2-65-2-1} & NN \\ \text{4-5} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-64-5-3-134-6-3} & NY \\ \text{2-54-2-1} & NN \\ \text{5-6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-5-3-136-234-64-2-1} & NY \\ \text{4-5} & YN \\ \text{5-6} & YY \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{134-2-12-56-5-3} & NN \\ \text{2-64-6-3} & NY \\ \text{4-5} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-6-3-134-54-2-1} & NN \\ \text{35-65-2} & NN \\ \text{4-6} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{136-234-64-2-12-5-3} & NN \\ \text{4-5} & YN \\ \text{5-6} & YY \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{136-54-234-5-3} & NN \\ \text{12-65-2} & NN \\ \text{4-6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{36-45-234-5-3-1} & NN \\ \text{12-64-2} & NN \\ \text{5-6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{136-2-12-54-234-5-3} & NY \\ \text{5-6} & YN \\ \text{4-6} & YN \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-6-32-54-64-2-1} & NN \\ \text{34-5-3-1} & NN \\ \text{5-6} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-54-236-56-2-1} & NY \\ \text{134-5-3} & NY \\ \text{4-6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{136-5-32-65-2-134-2} & NN \\ \text{4-5} & YN \\ \text{4-6} & YN \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{34-64-236-5-3-1} & NN \\ \text{12-65-2} & NY \\ \text{4-5} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{136-2-12-45-234-5-3} & NN \\ \text{5-6} & YN \\ \text{4-6} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{134-64-235-2-12-6-3} & NN \\ \text{4-5} & YN \\ \text{5-6} & YY \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-54-64-236-2-1} & NN \\ \text{34-5-3-1} & NN \\ \text{5-6} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-64-6-32-5-3-134-2-1} & NY \\ \text{4-5} & YY \\ \text{5-6} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{12-5-3-12-64-234-6-3} & NY \\ \text{4-5} & YN \\ \text{5-6} & YN \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{12-46-234-65-2} & NY \\ \text{35-6-3-1} & NN \\ \text{4-5} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{12-54-5-3-134-236-2} & NN \\ \text{5-6} & YN \\ \text{4-6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-65-236-5-3-134-2-1} & NY \\ \text{4-5} & YN \\ \text{4-6} & YN \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{34-54-236-5-3-1} & NN \\ \text{12-65-2} & NY \\ \text{4-6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-45-2-136-234-5-3-1} & NN \\ \text{5-6} & YN \\ \text{4-6} & YN \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-65-2-134-235-6-3-1} & NY \\ \text{4-5} & YN \\ \text{4-6} & YN \\ \text{1} & YN \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{36-46-234-5-3-1} & NY \\ \text{12-54-2} & NY \\ \text{5-6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-54-5-3-136-234-2-1} & NY \\ \text{5-6} & YN \\ \text{4-6} & YY \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-54-2-12-6-3-134-5-3} & NN \\ \text{5-6} & YN \\ \text{4-6} & YN \\ \text{1} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3,\left[\begin{array}{c|c}1134\text{-}2\text{-}135\text{-}2 & NY\\ 2\text{-}65\text{-}6\text{-}3 & NY\\ 4\text{-}5 & YY\\ 4\text{-}6 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}34\text{-}64\text{-}236\text{-}56\text{-}2\text{-}1 & NY\\ 135\text{-}2 & NN\\ 4\text{-}5 & YN\\ 1 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}34\text{-}54\text{-}6\text{-}3\text{-}1 & NY\\ 35\text{-}65\text{-}2 & NY\\ 2\text{-}64\text{-}2\text{-}1 & NN\\ 1 & YY\end{array}\right]\right)$$

$$\left(\Gamma_3^3,\left[\begin{array}{c|c}134\text{-}2\text{-}1\text{-}12\text{-}6\text{-}3 & NN\\ 35\text{-}65\text{-}2 & NY\\ 4\text{-}5 & YN\\ 4\text{-}6 & YY\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}136\text{-}56\text{-}2\text{-}134\text{-}64\text{-}2 & NN\\ 35\text{-}2 & YY\\ 4\text{-}5 & YN\\ 1 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}1134\text{-}64\text{-}2\text{-}12\text{-}6\text{-}3 & NN\\ 35\text{-}2 & YY\\ 4\text{-}5 & YN\\ 5\text{-}6 & YY\end{array}\right]\right)$$

$$\left(\Gamma_3^3,\left[\begin{array}{c|c}112\text{-}64\text{-}235\text{-}2 & NY\\ 134\text{-}6\text{-}3 & NN\\ 4\text{-}5 & YN\\ 5\text{-}6 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}2\text{-}46\text{-}56\text{-}2\text{-}134\text{-}6\text{-}3\text{-}1 & NY\\ 35\text{-}2 & YY\\ 4\text{-}5 & YN\\ 1 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}135\text{-}2\text{-}1\text{-}134\text{-}54\text{-}2 & NY\\ 36\text{-}2 & YY\\ 5\text{-}6 & YY\\ 4\text{-}6 & YN\end{array}\right]\right)$$

$$\left(\Gamma_3^3,\left[\begin{array}{c|c}2\text{-}54\text{-}2\text{-}134\text{-}5\text{-}3 & NY\\ 1136\text{-}2 & NN\\ 5\text{-}6 & YN\\ 4\text{-}6 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}136\text{-}54\text{-}56\text{-}2\text{-}135\text{-}2 & NN\\ 34\text{-}2 & YY\\ 4\text{-}6 & YN\\ 1 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}1135\text{-}2\text{-}12\text{-}64\text{-}6\text{-}3 & NN\\ 34\text{-}2 & YY\\ 4\text{-}5 & YY\\ 5\text{-}6 & YN\end{array}\right]\right)$$

$$\left(\Gamma_3^3,\left[\begin{array}{c|c}2\text{-}46\text{-}234\text{-}6\text{-}3\text{-}1 & NY\\ 35\text{-}2\text{-}1\text{-}1 & NY\\ 4\text{-}5 & YN\\ 5\text{-}6 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}34\text{-}56\text{-}46\text{-}2 & NY\\ 135\text{-}6\text{-}3 & NY\\ 2\text{-}54\text{-}2\text{-}1 & NN\\ 1 & YY\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}35\text{-}64\text{-}54\text{-}65\text{-}2\text{-}1 & NN\\ 136\text{-}2 & NN\\ 34\text{-}2 & YY\\ 1 & YN\end{array}\right]\right)$$

$$\left(\Gamma_3^3,\left[\begin{array}{c|c}112\text{-}6\text{-}32\text{-}54\text{-}2 & NN\\ 34\text{-}5\text{-}3\text{-}1 & NN\\ 5\text{-}6 & YN\\ 4\text{-}6 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}2\text{-}45\text{-}64\text{-}5\text{-}3 & NY\\ 12\text{-}65\text{-}2 & NY\\ 134\text{-}6\text{-}3 & NY\\ 1 & YY\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}2\text{-}56\text{-}45\text{-}64\text{-}5\text{-}3\text{-}1 & NY\\ 34\text{-}2\text{-}1 & NN\\ 36\text{-}2 & YY\\ 1 & YN\end{array}\right]\right)$$

$$\left(\Gamma_3^3,\left[\begin{array}{c|c}12\text{-}56\text{-}235\text{-}64\text{-}6\text{-}3 & NN\\ 134\text{-}2 & NN\\ 4\text{-}5 & YN\\ 1 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}12\text{-}54\text{-}64\text{-}2 & NY\\ 2\text{-}65\text{-}6\text{-}3 & NN\\ 34\text{-}5\text{-}3\text{-}1 & NN\\ 1 & YY\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}135\text{-}64\text{-}54\text{-}65\text{-}2 & NN\\ 34\text{-}2\text{-}1 & NN\\ 36\text{-}2 & YY\\ 1 & YN\end{array}\right]\right)$$

$$\left(\Gamma_3^3,\left[\begin{array}{c|c}2\text{-}4\text{-}32\text{-}64\text{-}54\text{-}6\text{-}3\text{-}1 & NY\\ 35\text{-}2\text{-}1 & NN\\ 5\text{-}6 & YN\\ 1 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}36\text{-}54\text{-}64\text{-}2 & NN\\ 12\text{-}65\text{-}2 & NN\\ 134\text{-}5\text{-}3 & NY\\ 1 & YY\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}112\text{-}64\text{-}2 & NY\\ 35\text{-}65\text{-}2 & NY\\ 134\text{-}6\text{-}3 & NN\\ 4\text{-}5 & YY\end{array}\right]\right)$$

$$\left(\Gamma_3^3,\left[\begin{array}{c|c}2\text{-}45\text{-}2\text{-}134\text{-}56\text{-}5\text{-}3 & NN\\ 136\text{-}2 & NN\\ 4\text{-}6 & YN\\ 1 & YN\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}2\text{-}45\text{-}64\text{-}5\text{-}3 & NN\\ 12\text{-}65\text{-}2 & NY\\ 34\text{-}6\text{-}3\text{-}1 & NY\\ 1 & YY\end{array}\right]\right)\quad\left(\Gamma_3^3,\left[\begin{array}{c|c}1134\text{-}5\text{-}3 & NN\\ 12\text{-}64\text{-}2 & NY\\ 2\text{-}54\text{-}6\text{-}3 & NY\\ 5\text{-}6 & YY\end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{135-2-12-6-3} & NN \\ \text{34-54-2} & NY \\ \text{4-65-6} & NN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-462-34-2-54-35} & NY \\ \text{1-36} & YY \\ \text{1-2} & YY \\ \text{5-6} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-34-2-53-46} & NN \\ \text{1-45-65} & NY \\ \text{2-36} & YY \\ \text{1-2} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-6-3-135-2-1} & NN \\ \text{34-64-2} & NN \\ \text{4-56-5} & NN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-21-462-452-35} & NY \\ \text{1-36} & YY \\ \text{3-4} & YY \\ \text{5-6} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-352-46-56} & NN \\ \text{1-21-45} & NN \\ \text{2-36} & YY \\ \text{3-4} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{135-212-6-3} & NN \\ \text{34-65-64-2} & NN \\ \text{4-5} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-2-54-1-63-12-35} & NY \\ \text{2-46} & YY \\ \text{3-4} & YN \\ \text{5-6} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-461-35-65} & NY \\ \text{2-34-36} & NY \\ \text{2-45} & YY \\ \text{1-2} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{12-6-3-134-2} & NN \\ \text{35-64-65-2} & NY \\ \text{4-5} & YN \\ \text{1} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-361-2-64-1-54-2} & NN \\ \text{2-35} & YY \\ \text{3-4} & YY \\ \text{5-6} & YN \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-45-64-2-56} & NY \\ \text{1-21-35} & NN \\ \text{2-36} & YY \\ \text{3-4} & YY \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-45-64-56-4-3} & NN \\ \text{135-2} & NN \\ \text{36-2-1} & NN \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-362-352-43-45} & NN \\ \text{1-46} & YY \\ \text{1-2} & YN \\ \text{5-6} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-2-652-45-63-2} & NN \\ \text{1-35} & YY \\ \text{1-46} & YY \\ \text{3-4} & YN \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{2-46-54-56-4-3} & NY \\ \text{136-2} & NN \\ \text{35-2-1} & NN \\ \text{1} & YN \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-2-53-1-54-12-36} & NY \\ \text{2-46} & YY \\ \text{3-4} & YY \\ \text{5-6} & YN \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-21-36-54-1-65} & NY \\ \text{2-35} & YY \\ \text{2-46} & YY \\ \text{3-4} & YN \end{array}\right]\right)$$

$$\left(\Gamma_3^3, \left[\begin{array}{c|c} \text{1135-65-2} & NN \\ \text{4-54-6} & NN \\ \text{36-2-1} & NN \\ \text{34-2} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-462-43-2-54-36} & NN \\ \text{1-35} & YY \\ \text{1-2} & YY \\ \text{5-6} & YY \end{array}\right]\right) \quad \left(\Gamma_4^3, \left[\begin{array}{c|c} \text{1-34-1-63-2} & NY \\ \text{3-46-5} & NN \\ \text{1-2-65} & NN \\ \text{2-45} & YY \end{array}\right]\right)$$

Appendix C

Dijkgraaf-Witten TQFTs

Separately from the rest of this thesis, we plan to study a TQFT approach to disproving the stable Andrews–Curtis conjecture. As a practice exercise we studied Dijkgraaf-Witten TQFTs. We prove that Proposition C.3 that in a specific case the dimension of the vector space can be arbitrarily large.

Lemma C.1 (Commuting cycles with one overlap)
Let $\sigma_1 : X_1 \to X_1$ and $\sigma_2 : X_2 \to X_2$ be cyclic permutations of the sets X_1 and X_2. If $X_1 \cap X_2 = \{*\}$, then $\sigma_1\sigma_2 = \sigma_2\sigma_1' = \sigma_2'\sigma_1$, where σ_1' is σ_1 with $*$ replaced by $\sigma_2^{-1}(*)$ and σ_2' is σ_2 with $*$ replaced by $\sigma_1(*)$.

Now we define $a_2 = (1\ 2\ 3\ 4\ 5\ 6\ 7)$ and $b_2 = (1\ 5\ 7\ 6\ 4\ 2\ 3)$. Then for $n \geq 2$, recursively define

$$a_{n+1} = (n+1\ n+2\ n+3\ n+4)g_n^{-1}a_ng_n.$$

and

$$b_{n+1} = (n\ n+3\ n+1\ n+2)g_n^{-1}b_ng_n$$

where

$$g_n = (n+1\ n+4)(n+2\ n+5)\cdots(3n+1, 3n+4).$$

Note $a_n = (1\ 2 \cdots 3n+1)$ and moreover that $g_n^{-1}a_ng_n$ and $g_n^{-1}b_ng_n$ are both "symbol replacements" of a_n and b_n (i.e., replacing $n+1$ for $n+4$ and so on, up through replacing $3n+1$ with $3n+4$) and thus satisfy all the same properties while fixing $n+1, n+2, n+3$.

Proposition C.2

For any $n \geq 2$, we have $a_nb_na_n = b_na_nb_n$ and the centralizer $C_{S_{3n+1}}(a_n, b_n) = 1$.

Proof. We show that for all $n \geq 2$, $a_nb_na_n = b_na_nb_n$ by showing that $(a_nb_na_n)^2 = (a_nb_n)^3$. We do this recursively. First note that $(a_2b_2a_2)^2 = (a_2b_2)^3 = 1$. Now assume that $(a_nb_na_n)^2 = (a_nb_n)^3 = 1$. Then

$$\begin{aligned}
(a_{n+1}b_{n+1})^3 &= ((n+1\ n+2\ n+3\ n+4)g_n^{-1}a_ng_n(n\ n+3\ n+1\ n+2)g_n^{-1}b_ng_n)^3 \\
&= ((n+1\ n+2\ n+3\ n+4)(n+4\ n+3\ n+1\ n+2)g_n^{-1}a_ng_ng_n^{-1}b_ng_n)^3 \\
&= ((n+1\ n+3\ n+2)g_n^{-1}a_ng_ng_n^{-1}b_ng_n)^3 \\
&= (n+1\ n+3\ n+2)^3(g_n^{-1}a_ng_ng_n^{-1}b_ng_n))^3 \\
&= g_n^{-1}(a_nb_n)^3g_n = 1.
\end{aligned}$$

where the first step follows by the lemma since the only the only overlap is n and $g_n^{-1}a_ng_n$ permutes n to $n+4$.

Using the work above, we have

$$\begin{aligned}
(a_{n+1}b_{n+1}a_{n+1})^2 &= (g_n^{-1}a_ng_ng_n^{-1}b_ng_n(n+1\ n+3\ n+2)(n+1\ n+2\ n+3\ n+4)g_n^{-1}a_ng_n)^2 \\
&= (g_n^{-1}a_ng_ng_n^{-1}b_ng_n(n+3\ n+4)g_n^{-1}a_ng_n)^2 \\
&= (g_n^{-1}a_ng_ng_n^{-1}b_ng_ng_n^{-1}a_ng_n(n\ n+3))^2 \\
&= g_n^{-1}(a_nb_na_n)^2g_n(n\ n+3)^2 = 1,
\end{aligned}$$

where the last line follows since $g_n^{-1}a_nb_na_ng_n$ does not contain an n. Indeed, for $n = 2$ this can be calculated, and we see that going from n to $n+1$, conjugating by g_n ensures there cannot be a $n+1$ present (since it becomes an $n+4$), and then we simply multiply by $(n\ n+3)$.

Thus we have that for all $n \geq 2$, $a_n b_n a_n = b_n a_n b_n$.

Moreover, letting S_{3n+1} act on itself by conjugation, we have by orbit-stabilizer that

$$|C_{S_{3n+1}}(a_n)| = \frac{(3n+1)!}{|\{\text{length } 3n+1 \text{ cycles}\}|} = \frac{(3n+1)!}{(3n)!} = 3n+1.$$

Since a_n has order $3n+1$, we must have that the centralizer of a_n is just the powers of a_n, i.e., $C_{S_{3n+1}}(a_n) = \{a_n^i\}_{i=0}^{3n}$. Thus it suffices to show that b_n is not a power of a_n. Note a_n^i always permutes k to $k+i \mod n$, which b_n does not satisfy for any i, since, for example, b_n always permutes 1 to $3n-1$ and $3n-1$ to $3n+1$. Thus $C_{S_{3n+1}}(a_n, b_n) = 1$. $\square$

Proposition C.3

Let Y be the trefoil complement in D^3 with $G = S_{3n+1}$. Let the S^2 boundary of D^3 be labeled by (ρ_1, std) and the T^2 boundary of the knot be labeled by $(\rho_2, 1)$, where ρ_1 is the trivial map, $\rho_2 : \mathbb{Z} \times \mathbb{Z} \to S_{3n+1}$ is the map $(x, y) \mapsto (a_n, b_n a_n^2 b_n a_n^{-4})$, std is the standard $3n$-dimensional representation of S_{3n+1}, and 1 is the trivial representation. Then $\dim V(Y) \geq 3n$.

Proof. Let $\rho : \pi_1(Y) = \langle \alpha, \beta \mid \alpha\beta\alpha = \beta\alpha\beta \rangle \to S_{3n+1}$ be the map $\alpha \mapsto a_n$ and $\beta \mapsto b_n$. Then the induced maps $\rho|_{S^2}$ and $\rho|_{T^2}$ given by $\rho|_\Sigma : \pi_1(\Sigma) \to \pi_1(Y) \xrightarrow{\rho} S_{3n+1}$ are precisely ρ_1 and ρ_2. Note the centralizer $C_{S_{3n+1}}(\operatorname{im} \rho_1) = S_{3n+1}$. Now we get a representation of $C_{S_{3n+1}}(\operatorname{im} \rho) = 1$ by $1 \to S_{3n+1} \times C_{S_{3n+1}}(\operatorname{im} \rho_2) \xrightarrow{\mathrm{std} \otimes 1} \mathrm{End}(\mathbb{C}^{3n} \otimes \mathbb{C})$. The trivial component of this representation is all of $\mathbb{C}^{3n} \otimes \mathbb{C} \cong \mathbb{C}^{3n}$ since the representation is trivial. Thus in the direct sum decomposition of $V(Y)$, there is a factor of $\mathbb{C}^{3n}$. $\square$

Bibliography

[1] Selman Akbulut and Robion Kirby. “A potential smooth counterexample in dimension 4 to the Poincare conjecture, the Schoenflies conjecture, and the Andrews–Curtis conjecture”. In: *Topology* 24.4 (1985), pp. 375–390.

[2] James J. Andrews and Morton L. Curtis. “Free groups and handlebodies”. In: *Proceedings of the American Mathematical Society* 16.2 (1965), pp. 192–195.

[3] Kshitij Bansal et al. “HOList: An environment for machine learning of higher-order theorem proving”. In: *arXiv preprint arXiv:1904.03241* (2019).

[4] Jonathan Barmak. “An exotic presentation of $\mathbb{Z} \times \mathbb{Z}$ and the Andrews–Curtis conjecture”. In: *Algebraic & Geometric Topology* 25 (Mar. 2025), pp. 345–355. DOI: 10.2140/agt.2025.25.345.

[5] Martin R. Bridson. “The complexity of balanced presentations and the Andrews–Curtis conjecture”. In: *arXiv preprint arXiv:1504.04187* (2015).

[6] Robert G. Burns and Olga Macedonska. “Balanced Presentations of the Trivial Group”. In: *Bulletin of the London Mathematical Society* 25 (1993), pp. 513–526.

[7] III C. F. Miller and Paul E. Schupp. “Some presentations of the trivial group”. In: *Groups, languages, algorithms* 378 (2005), pp. 113–115.

[8] Burtis G Casler. “An imbedding theorem for connected 3-manifolds with boundary”. In: *Proceedings of the American Mathematical Society* 16.4 (1965), pp. 559–566.

[9] M. M. Cohen. “Dimension estimates in collapsing $X \times I^q$”. In: *Topology* 14.3 (1975). MR 0375335 (51 #11530), pp. 253–256.

[10] M. M. Cohen. “Whitehead torsion, group extensions, and Zeeman’s conjecture in high dimensions”. In: *Topology* 16.1 (1977). MR 0436155 (55 #9105), pp. 79–88.

[11] Francesco Costantino. “A short introduction to shadows of 4-manifolds”. In: *arXiv preprint math/0405582* (2004).

[12] Lucas Fagan and Jon McCammond. “Fake Surfaces for some Potential Andrews–Curtis Counterexamples”. Manuscript in preparation. 2025.

[13] Lucas Fagan, Yang Qiu, and Zhenghan Wang. *Classification of Cellular Fake Surfaces*. 2024. arXiv: `2406.09439 [math.GT]`. URL: `https://arxiv.org/abs/2406.09439`.

[14] Lucas Fagan, Yang Qiu, and Zhenghan Wang. “Stable Andrews-Curtis Conjecture via Fake Surfaces and Zeeman Conjecture”. In: *arXiv preprint math/2412.12293* (2024).

[15] Ximena Fernandez. “Morse theory for group presentations”. In: *arXiv preprint arXiv:1912.00115* (2019).

[16] Michael H Freedman and Frank Quinn. *Topology of 4-Manifolds (PMS-39), Volume 39*. Vol. 46. Princeton University Press, 2014.

[17] David Gillman and Dale Rolfsen. “The Zeeman conjecture for standard spines is equivalent to the Poincaré conjecture”. In: *Topology* 22.3 (1983). MR 710105 (85b:57013), pp. 315–323.

[18] George Havas and Colin Ramsay. “Breadth-first search and the Andrews–Curtis conjecture”. In: *International Journal of Algebra and Computation* 13.1 (2003), pp. 61–68. DOI: `10.1142/S0218196703001365`.

[19] Naoe Hironobu. "Mazur Manifolds and Corks with Small Shadow Complexities". In: *Osaka Journal of Mathematics* 55.3 (2018), pp. 479–498.

[20] Cynthia Hog-Angeloni, Wolfgang Metzler, and Allan J Sieradski. *Two-dimensional homotopy and combinatorial group theory*. Vol. 197. Cambridge University Press, 1993.

[21] Hiroshi Ikeda. "Acyclic fake surfaces". In: *Topology* 10.1 (1971), pp. 9–36.

[22] Yuya Koda, Bruno Martelli, and Hironobu Naoe. "Four-manifolds with shadow-complexity one". In: *Annales de la Faculté des sciences de Toulouse: Mathématiques*. Vol. 31. 4. 2022, pp. 1111–1212.

[23] Yuya Koda and Hironobu Naoe. "Shadows of acyclic 4–manifolds with sphere boundary". In: *Algebraic & Geometric Topology* 20.7 (2020), pp. 3707–3731.

[24] Boris Lishak. "Balanced finite presentations of the trivial group". In: *Journal of Topology and Analysis* 9.02 (2017), pp. 363–378.

[25] Alexei Lisitsa. "New Andrews–Curtis trivializations for Miller–Schupp group presentations". In: *Examples and Counterexamples* 6 (2024/12/06 2024). DOI: 10.1016/j.exco.2024.100168. URL: https://doi.org/10.1016/j.exco.2024.100168.

[26] Alexei Lisitsa. "Towards computer-assisted proofs of parametric Andrews-Curtis simplifications, II". In: *LPAR 2024 Complementary Volume*. Ed. by Nikolaj Bjørner, Marijn Heule, and Andrei Voronkov. Vol. 18. Kalpa Publications in Computing. EasyChair, 2024, pp. 131–136. DOI: 10.29007/p2w1. URL: /publications/paper/tkQS.

[27] R.C. Lyndon and P.E. Schupp. *Combinatorial Group Theory*. Classics in Mathematics. Springer Berlin Heidelberg, 2015. ISBN: 9783642618963. URL: https://books.google.com/books?id=cOLrCAAAQBAJ.

[28] Wilhelm Magnus, Abraham Karrass, and Donald Solitar. *Combinatorial Group Theory: Presentations of Groups in Terms of Generators and Relations*. 2nd. Dover Publications, 1976.

[29] Bruno Martelli. "Four-manifolds with shadow-complexity zero". In: *International Mathematics Research Notices* 2011.6 (2011), pp. 1268–1351.

[30] Sergei Matveev. *Algorithmic topology and classification of 3-manifolds*. Vol. 9. Springer, 2007.

[31] Sergei Vladimirovich Matveev. "Special spines of piecewise linear manifolds". In: *Mathematics of the USSR-Sbornik* 21.2 (1973), p. 279.

[32] James McCool and Andreas Pietrowski. "On a conjecture of W. Magnus". In: *Word Problems: Decision Problems and the Burnside Problem in Group Theory*. Vol. 71. Studies in Logic and the Foundations of Mathematics. Amsterdam: North-Holland, 1973, pp. 453–456.

[33] Wolfgang Metzler. "Über den Homotopietyp zweidimensionaler CW-Komplexe und Elementartransformationen bei Darstellungen von Gruppen durch Erzeugende und definierende Relationen". In: *Journal für die Reine und Angewandte Mathematik* 285 (1976), pp. 7–23.

[34] Alexei D. Miasnikov. "Genetic algorithms and the Andrews–Curtis conjecture". In: *arXiv preprint math/0304306* (2003).

[35] Alexei D. Miasnikov and Alexei G. Myasnikov. *Balanced presentations of the trivial group on two generators and the Andrews-Curtis conjecture*. 2003. arXiv: `math/0304305 [math.GR]`. URL: `https://arxiv.org/abs/math/0304305`.

[36] Alexei D. Myasnikov, Alexei G. Myasnikov, and Vladimir Shpilrain. "On the Andrews–Curtis equivalence". In: *Contemporary Mathematics* 296 (2002), pp. 183–198.

[37] OEIS Foundation Inc. *Entry A373555 in The On-Line Encyclopedia of Integer Sequences*. `https://oeis.org/A373555`. 2024.

[38] Dmitry Panteleev and Alexander Ushakov. "Conjugacy search problem and the Andrews–Curtis conjecture". In: *Groups Complexity Cryptology* 11.1 (2019), pp. 43–60.

[39] Stanislas Polu and Ilya Sutskever. "Generative language modeling for automated theorem proving". In: *arXiv preprint arXiv:2009.03393* (2020).

[40] Jimmie Arnold Robertson. *Fake-Surfaces which are Spines of 3-Balls.* The Florida State University, 1973.

[41] C.P. Rourke and B.J. Sanderson. *Introduction to Piecewise-linear Topology.* Ergebnisse der Mathematik und ihrer Grenzgebiete. Springer-Verlag, 1982. ISBN: 9783540111023. URL: `https://books.google.com/books?id=AoEpAQAAMAAJ`.

[42] Ali Shehper et al. *What makes math problems hard for reinforcement learning: a case study.* 2025. arXiv: `2408.15332 [cs.LG]`. URL: `https://arxiv.org/abs/2408.15332`.

[43] David Silver et al. "Mastering the game of Go with deep neural networks and tree search". In: *Nature* 529.7587 (2016), pp. 484–489.

[44] Jean E Taylor. "The structure of singularities in soap-bubble-like and soap-film-like minimal surfaces". In: *Annals of Mathematics* 103.3 (1976), pp. 489–539.

[45] Trieu H Trinh et al. "Solving olympiad geometry without human demonstrations". In: *Nature* 625.7995 (2024), pp. 476–482.

[46] Vladimir G Turaev. *Quantum invariants of knots and 3-manifolds.* de Gruyter, 2010.

[47] Vladimir G Turaev and Oleg Ya Viro. "State sum invariants of 3-manifolds and quantum 6j-symbols". In: *Topology* 31.4 (1992), pp. 865–902.

[48] Masaaki Wada. "Twisted Alexander polynomial for finitely presentable groups". In: *Topology* 33.2 (1994), pp. 241–256. ISSN: 0040-9383. DOI: https://doi.org/10.1016/0040-9383(94)90013-2. URL: https://www.sciencedirect.com/science/article/pii/0040938394900132.

[49] C. T. C. Wall. "Formal Deformations". In: *Proceedings of the London Mathematical Society*. s3 16.1 (1966), pp. 342–352. DOI: 10.1112/plms/s3-16.1.342. URL: https://doi.org/10.1112/plms/s3-16.1.342.

[50] J. H. C. Whitehead. "On Incidence Matrices, Nuclei and Homotopy Types". In: *Annals of Mathematics* 42.5 (1941), pp. 1197–1239. ISSN: 0003486X, 19398980. URL: http://www.jstor.org/stable/1970465 (visited on 10/08/2025).

[51] J. H. C. Whitehead. "Simple homotopy types". In: *American Journal of Mathematics* 72.1 (1950), pp. 1–57.

[52] Perrin Wright. "Formal 3-deformations of 2-polyhedra". In: *Proceedings of the American Mathematical Society* 37.1 (1973), pp. 305–308.

[53] Perrin Wright. "Group presentations and formal deformations". In: *Transactions of the American Mathematical Society* 208 (1975), pp. 161–169.

[54] Kaiyu Yang and Jia Deng. "Learning to Prove Theorems via Interacting with Proof Assistants". In: *International Conference on Machine Learning (ICML)*. 2019.

[55] E.C. Zeeman. "On the dunce hat". In: *Topology* 2.4 (1963), pp. 341–358. ISSN: 0040-9383. DOI: https://doi.org/10.1016/0040-9383(63)90014-4. URL: https://www.sciencedirect.com/science/article/pii/0040938363900144.

[56] Eric Christopher Zeeman. "Seminar on combinatorial topology". In: *(No Title)* (1963).

[57] Heiner Zieschang. "Über die Nielsensche Kürzungsmethode in freien Produkten mit Amalgam". In: *Inventiones Mathematicae* 10 (1970), pp. 4–37. DOI: 10.1007/BF01402968.