The optimization problem of triviality search in group presentations

Yifeng Wei[1,a,*]

[1]College of Science, China Jiliang University, Hangzhou, China
[a]S22080701009@cjlu.edu.cn
**Corresponding author*

Abstract: *This paper transforms the triviality search problem in group presentations into an optimization problem by defining decision variables, objective functions, and constraints, and conducts an in-depth analysis of the scale and complexity of the proposed model, highlighting its computational challenges. A genetic algorithm-based solution framework is designed to efficiently search for Andrews-Curtis transformation sequences, and its effectiveness for specific group presentations is validated, successfully solving the Andrews-Curtis transformation sequence search problem. This study not only provides a novel methodology for addressing the triviality search problem but also demonstrates the practical application of optimization techniques in group theory, offering new insights for future research in this field.*

Keywords: *Andrews-Curtis Conjecture; Triviality Search; Optimization Model; Genetic Algorithm*

1. Introduction

In the field of group theory, the theory of group presentations occupies a central position, and the problem of group isomorphism is an important and widely studied topic within this domain.

The origin of the group isomorphism problem can be traced back to 1903, when Tietze[1] first pointed out that groups defined by different sets of generators and relators may exhibit isomorphic relationships. Specifically, determining whether two given group presentations are isomorphic and exploring the transformation paths between isomorphic group presentations constitute the core content of the group isomorphism problem.

In 1965, Andrews and Curtis[2] proposed the famous Andrews-Curtis Conjecture, which provides an important theoretical foundation for the transformation between balanced presentations of trivial groups. The Andrews-Curtis Conjecture states that any balanced presentation of a trivial group can be transformed into a standard presentation through a series of transformations. Although this conjecture has been extensively studied over the past few decades, its validity has not yet been fully proven, making it one of the important unsolved problems in group theory.

In recent years, numerous scholars have devoted efforts to verifying this conjecture through methods such as computer simulations, exploring its possibilities. In this process, algorithms such as genetic algorithms[3-5], breadth-first search algorithms[6], blind search algorithms[7], and distance metric ensemble learning[8, 9] have been successively applied to search for Andrews-Curtis transformation sequences of balanced presentations of trivial groups. However, the primary goal of these methods has been to find transformation sequences for specific group presentations, lacking a systematic optimization framework.

Given the aforementioned research background, this paper, for the first time, defines the triviality search problem in group presentations as an optimization problem. It systematically elaborates on how to apply the ideas and methods of optimization problems to transform a given group presentation into the standard presentation of a trivial group through Andrews-Curtis transformations. Based on the optimization model, this paper designs a genetic algorithm-based Andrews-Curtis transformation sequence search algorithm for solving the triviality search problem in group presentations.

The main contributions of this paper are as follows:

(1) An optimization model for the triviality search problem in group presentations is proposed.

(2) A genetic algorithm-based framework for searching Andrews-Curtis transformation sequences is

constructed.

2. Preliminaries

In this section, we review some concepts related to group presentations and Andrews-Curtis transformations.

Definition 1.1[10]

Let X be a set, and R be a set of relators on X. If there exists a group G defined by generators $x \in X$ and relators $r = e$(where $r \in R$), such that $G \cong F/N$, where F is the free group on X and N is the normal closure of R in F, then $\langle X \mid R \rangle$ is called a **presentation** of the group G.

Let $X = \{x_1, x_2, \ldots, x_n\}$ and $R = \{r_1, r_2, \ldots, r_m\}$.

If $n = m$, then

$$\langle x_1, x_2, \ldots, x_n \mid r_1, r_2, \ldots, r_n \rangle \tag{1}$$

is called a **balanced presentation**.

If $X = R$, then

$$\langle x_1, x_2, \ldots, x_n \mid x_1, x_2, \ldots, x_n \rangle \tag{2}$$

is called the **standard presentation** of the trivial group.

Definition 1.2[2]

Let the group presentations

$$G_n = \langle x_1, x_2, \ldots, x_n \mid r_1, r_2, \ldots, r_n \rangle \tag{3}$$

and

$$H = \langle x_1, x_2, \ldots, x_n \mid r_1', r_2', \ldots, r_n' \rangle \tag{4}$$

be given. If there exists a sequence of elementary transformations, including:

(1) **(AC1)** Replacing r_i with r_i^{-1}, i.e., $r_i = r_i^{-1}$;

(2) **(AC2)** Replacing r_i with $r_i r_j$, i.e., $r_i = r_i r_j$, where $i \neq j$;

(3) **(AC3)** Replacing r_i with $x_j r_i x_j^{-1}$ or $x_j^{-1} r_i x_j$, i.e., $r_i = x_j r_i x_j^{-1}$ or $r_i = x_j^{-1} r_i x_j$, such that G is transformed into H, then G and H are said to be **Andrews-Curtis equivalent**, and these elementary transformations are called **Andrews-Curtis transformations**.

Conjecture 1.3[2]

If the group presentation

$$G_n = \langle x_1, x_2, \ldots, x_n \mid r_1, r_2, \ldots, r_n \rangle \tag{5}$$

is a balanced presentation of the trivial group, then G_n is Andrews-Curtis equivalent to the standard presentation of the trivial group,

$$\langle x_1, x_2, \ldots, x_n \mid x_1, x_2, \ldots, x_n \rangle. \tag{6}$$

3. The Triviality Search Problem in Group Presentations

The core of the triviality search problem in group presentations lies in finding a sequence of Andrews-Curtis transformations that gradually transforms a given group presentation into the standard presentation of the trivial group. This process essentially involves searching for an appropriate path in a complex transformation space, where the group presentation is successfully simplified to the standard presentation of the trivial group while satisfying Andrews-Curtis equivalence. Therefore, the triviality search problem in group presentations can be transformed into a typical path optimization problem.

In the general triviality search path optimization problem, we consider the most basic form of a balanced group presentation:

ISSN 2616-5805 Vol. 6, Issue 2: 25-32, DOI: 10.25236/AJMS.2025.060204

$$G = \langle x_1, x_2, \dots, x_n \mid r_1, r_2, \dots, r_n \rangle \quad (7)$$

3.1. Decision Variables

The decision variables are the transformation operations in the Andrews-Curtis transformation sequence, including the type of transformation (AC1, AC2, AC3), the relators involved in the transformation, and the order of the transformations.

Let i denote the Andrews-Curtis transformation acting on the i-th relator $r_i (i = 1,2, \dots, n)$. The specific Andrews-Curtis transformations are as follows:

(1) **AC1 Transformation**: $T_{AC1}(i,j)$ denotes replacing the relator r_i with $r_i r_j (i \neq j)$.

(2) **AC2 Transformation**: $T_{AC2}(i)$ denotes replacing the relator $r_i$with r_i^{-1}.

(3) **AC3 Transformation**: $T_{AC3}(i,l,k)$ denotes replacing the relator r_i with $x_l r_i x_l^{-1}$ or $x_l^{-1} r_i x_l$, where $k = 0$ indicates replacing r_i with $x_l r_i x_l^{-1}$, and $k = 1$ indicates replacing r_i with $x_l^{-1} r_i x_l$.

Considering the order in the transformation sequence, let $T = T_m \circ \cdots \circ T_2 \circ T_1$ denote the Andrews-Curtis transformation sequence. The decision variable $T(G)$ is defined as:

$$T(G) = T_m(\cdots (T_2(T_1(G))) \cdots) \quad (8)$$

where m represents the length of the transformation sequence.

3.2. Objective Function

The design of the objective function is one of the core aspects of this paper. The goal of the search problem is to find an Andrews-Curtis transformation sequence T that transforms the given group presentation G into the standard presentation of the trivial group. Therefore, we define the objective function $F\big(T(G)\big)$ as the sum of the occurrences of generators x_i and their inverses x_i^{-1} in the relators $r_1, r_2, \dots, r_n$, i.e.:

$$F\big(T(G)\big) = \sum_{i=1}^{n} |r_i| \quad (9)$$

where $|r_i|$ denotes the number of occurrences of generators and their inverses in the relator r_i.

The minimum value of the objective function is n. When $F\big(T(G)\big) = n$, it can be proven that the group presentation G has been successfully trivialized.

3.3. Constraints

The constraints include the conditions of the Andrews-Curtis Conjecture, namely the types of Andrews-Curtis transformations and the balancedness constraints of the group presentation. Additionally, when searching for transformation sequences, it is necessary to ensure that their lengths are within a certain range to prevent unlimited searching. The specific constraints are as follows:

(1) **Transformation Type Constraint**: Each transformation T_o must be one of the following forms:

$$T_o \in \{T_{AC1}(i,j), T_{AC2}(i), T_{AC3}(i,l,k)\} \quad (10)$$

where $i, j, l = 1,2, \dots, n$, $k = 0,1$, and $i \neq j$.

(2) **Balancedness Constraint**: To ensure that the group presentation $T(G)$ remains balanced after applying the Andrews-Curtis transformation sequence T, the transformed relators must satisfy the following conditions: the relators are pairwise distinct, and none of the n relators equals the identity element e. Let

$$T(G) = \langle x_1, x_2, \dots, x_n \mid r_1{}', r_2{}', \dots, r_n{}' \rangle \quad (11)$$

Then,

$$r_i' \neq r_j', r_i' \neq e \quad (12)$$

(3) **Transformation Sequence Length Constraint**: The length m of the Andrews-Curtis transformation sequence must be constrained by a threshold to avoid unlimited growth. Therefore, m is constrained as follows:

ISSN 2616-5805 Vol. 6, Issue 2: 25-32, DOI: 10.25236/AJMS.2025.060204

$$0 \le m \le M \tag{13}$$

where $M \in N^{+}$.

The triviality search path optimization problem in group presentations can be formulated as:

$$\min F(T(G)) \tag{14}$$

subject to

$$\begin{cases} T(G) = T_m(\cdots(T_2(T_1(G)))\cdots) \\ T_o \in \{T_{AC1}(i,j), T_{AC2}(i), T_{AC3}(i,l,k)\}, \\ 0 \le m \le M, M \in N^{+}, \\ r_i' \ne r_j', \\ r_i' \ne e \end{cases} \tag{15}$$

3.4. Analysis of the Scale and Complexity of the Optimization Problem

3.4.1. Analysis of Problem Scale

The scale of the triviality search path optimization problem in group presentations is primarily influenced by two factors: the number of generators and relators, and the length of the Andrews-Curtis transformation sequence.

Let the number of generators and relators be n. For any relator r_i, three types of Andrews-Curtis transformations can be applied, and each transformation can be combined with other elements. Table 1 shows the number of possible Andrews-Curtis transformations for different values of n.

As the number of generators and relators nn in the group presentation G increases, the number of possible Andrews-Curtis transformations that can be applied to G grows at a quadratic rate. From the perspective of the number of generators and relators alone, the scale of the optimization problem is $O(3n^2)$.

As the length m of the Andrews-Curtis transformation sequence increases, the number of possible transformation sequences grows exponentially. For a single Andrews-Curtis transformation, there are $3n^2$ possible transformations. For a transformation sequence of length m, there are $(3n^2)^m$ possible sequences.

(1) When $n = 2$ and $m = 5$, the number of possible transformation sequences is $(3 \times 5^2)^5 \approx 2.37 \times 10^9$.

(2) When $n = 2$ and $m = 10$, the number of possible transformation sequences is $(3 \times 5^2)^{10} \approx 5.63 \times 10^{18}$.

As n and m increase, the scale of the problem expands rapidly, far exceeding the computational capabilities of modern computers.

3.4.2. Analysis of Problem Complexity

Although for a given group presentation G and transformation sequence T, it is possible to verify in polynomial time whether T can successfully transform G into a balanced presentation of the trivial group, both the objective function $F(T(G))$ and the Andrews-Curtis transformation sequence T are discrete. If one attempts to exhaustively enumerate all possible Andrews-Curtis transformation sequences T, the required algorithm runtime would be as high as $O((3n^2)^m)$.

The triviality search path optimization problem in group presentations is, in fact, an **NP-hard problem**, making it extremely difficult to solve. When the problem scale is large, direct exhaustive search methods become infeasible. In dealing with such problems, it is usually necessary to resort to heuristic algorithms to find approximate optimal solutions within an acceptable time frame.

4. Solution Methods and Steps

4.1. Group Transformation Problem Under a Given Transformation Sequence

Due to the inherent characteristics of computer programs, which are well-suited for string operations,

ISSN 2616-5805 Vol. 6, Issue 2: 25-32, DOI: 10.25236/AJMS.2025.060204

we define string operation rules to replace the power operations in polynomials. Since most triviality search problems in group presentations involve groups with two generators, this subsection defines the arrangement and operation rules for strings using group presentations with two generators.

4.1.1. String Arrangement Rules

Let the group presentation be

$$G = \langle x, y \mid r_1, r_2 \rangle \tag{16}$$

where x and y are generators, and r_1, r_2 are reduced words in x and y, representing the relators of G.

The specific rules are as follows:

(1) Lowercase letters represent generators, and uppercase letters represent the inverses of generators. For example, X represents x^{-1}, and Y representsy^{-1}.

(2) Consecutive letters represent powers of generators. For example, xx represents x^2, and YYY represents y^{-3}.

Table 1 shows the comparison between the mathematical definitions of relators and the string arrangement rules.

Table 1 Comparison Between Mathematical Definitions and Program Rules.

Comparison Between Mathematical Definitions and Program Rules							
x	x^{-1}	y	y^{-1}	x^2	y^2	x^{-2}	y^{-2}
x	X	y	Y	xx	yy	XX	YY

4.1.2. String Operation Rules

In Andrews-Curtis transformations, to ensure that the transformed strings remain reduced, the rules for reduced word operations must be followed. Both the pre- and post-transformation strings must satisfy the reduced word requirement, meaning that a generator and its inverse cannot be adjacent. Therefore, after each transformation, the string must be checked and substrings such as xX, Xx, yY, and Yy must be eliminated. This operation must be repeated until the string no longer contains such substrings, ensuring that the final result remains a reduced word.

4.2. Genetic Algorithm-Based Solution Framework for Trivial Group Presentations

In this paper, we employ the genetic algorithm to solve the optimization problem of triviality search in group presentations. The genetic algorithm can effectively explore the potential solution space of Andrews-Curtis transformation sequences through random search and evolutionary strategies, overcoming the limitations of traditional methods in terms of search efficiency and computational complexity.

4.2.1. Genetic Encoding

The encoding method directly affects the crossover and mutation operations in the genetic algorithm, thereby influencing the algorithm's performance and efficiency. The decision variable is the Andrews-Curtis transformation sequence $T(G)$, where the length m of the transformation sequence ranges from 0 to M, and mm is variable. However, crossover operations between transformation sequences of different lengths are not practical in real-world scenarios, as the optimization goal is to find a suitable transformation sequence rather than minimizing the sequence length. Therefore, we fix the length of individual encoding to M.

This paper adopts a character-based encoding method, where each character $T_i \in \{T_{AC1}(i,j), T_{AC2}(i), T_{AC3}(i,l,k)\}$ represents a specific transformation operation. Character encoding is suitable for decision sequences with ordered relationships and can flexibly represent various types of transformation operations.

4.2.2. Population Size

The population size directly affects the efficiency and convergence speed of the genetic algorithm. A small population size may prevent individuals from fully exploring the diversity of the solution space, while a large population size helps improve the algorithm's global search capability at the cost of increased computational resources.

ISSN 2616-5805 Vol. 6, Issue 2: 25-32, DOI: 10.25236/AJMS.2025.060204

4.2.3. Fitness Function

The fitness function is a key metric in the genetic algorithm for evaluating the quality of individuals, guiding the algorithm's search direction and evolutionary process. It is used to measure the effectiveness of a given Andrews-Curtis transformation sequence in transforming the group presentation. The fitness function is defined as the sum of the occurrences of generators and their inverses in the relators:

$$F(G) = \sum_{i=1}^{n} |r_i| \tag{17}$$

where $G = \langle x_1, x_2, \ldots, x_n \mid r_1, r_2, \ldots, r_n \rangle$, and $|r_i'|$ represents the number of occurrences of generators and their inverses in the relator r_i'. A higher fitness value indicates that the group presentation G is closer to the standard presentation of the trivial group. When the fitness function $F(G) = n$, the group presentation G is the standard presentation of the trivial group.

While this fitness function is reasonable, it does not guarantee that the group presentation is balanced. For example, consider the group presentationG_e:

$$G_e = \langle x, y \mid xy, e \rangle \tag{18}$$

The fitness function value for $G_e$is $F(G_e) = 2$. Based on the definition of the fitness function, G_e appears to be the standard presentation of the trivial group. However, G_e is neither balanced nor a trivial group. This issue is addressed by the termination conditions.

4.2.4. Population Evolution Strategy

The population evolution strategy is the core of the genetic algorithm, simulating the gene exchange process in natural selection. Through selection, crossover, and mutation operations, the population evolves toward higher fitness.

(1) Selection Operation: A fitness-based ranking selection method is used. The fitness values of each individual in the population are calculated, and the individuals are ranked accordingly. A proportion of parent individuals is selected from those with higher fitness values.

(2) Crossover Operation: A multi-point crossover strategy is adopted. Two individuals are randomly selected from the parent population as crossover candidates. Several crossover points are randomly chosen in their sequences, and the gene segments are exchanged to produce two new offspring individuals.

(3) Mutation Operation: For each newly generated offspring individual, there is a certain probability of selecting one or more positions in the sequence and randomly replacing the transformations at those positions with new transformations.

4.2.5. Termination Conditions

(1) **Maximum Iteration Generations**: The algorithm stops automatically when it reaches the maximum number of generations.

(2) **Fitness Threshold**: If the fitness of an individual in the population reaches this threshold, it indicates that the group presentation has been successfully transformed into the standard presentation of the trivial group.

(3) **Balancedness Condition**: If any relator equals ee or two relators are equal, the group presentation is considered not to be the standard presentation.

5. Experiments and Solution Results

5.1. Experimental Setup

This section uses a class of group presentations constructed by Miller and Schupp as an example to demonstrate the parameters of the genetic algorithm and the process of Andrews-Curtis transformations.

The group presentation is defined as:

$$G_{MS} = \langle a, b | a^{-1} b^n a = b^{n+1}, a = w \rangle \tag{19}$$

where the exponent sum of a in w is 0.

Let $n = 2$ and $a = a^{-2} b^3 a^2$. The group presentation G is equivalent to:

$$G_{MS}' = \langle a, b | a^{-1} b^2 a = b^3, a = a^{-2} b^3 a^2 \rangle$$

$$= \langle a,b | a^{-1}b^2ab^{-3} = e, a^{-3}b^3a^2 = e \rangle \quad (20)$$

Under the string arrangement rules, G' is equivalent to H:

$$H_{MS} = \langle x,y | XyyxYYY, XXXyyyxx \rangle \quad (21)$$

5.2. Comparative Analysis of Genetic Algorithm Parameters

The genetic algorithm was implemented on the Windows 11 operating system using the Python programming language. To verify the effectiveness of the algorithm, four sets of experiments were designed, focusing on key parameters such as population size (P), gene length (L), number of crossover points (N), and mutation rate (M). In each set of experiments, only one parameter was adjusted while the others remained fixed. The parameter values were gradually increased to systematically evaluate the impact of each key parameter on the algorithm's performance. During the genetic algorithm experiments, the running time of the code blocks was precisely measured using Python's time module.

To comprehensively evaluate the performance of the genetic algorithm, the maximum number of iterations was set to 5000. For each parameter configuration, the algorithm was independently run 10 times. The success rate was calculated as the percentage of runs where the fitness function value reached 2 out of the 10 runs. The maximum, minimum, and average running times were also recorded. These data served as the core evaluation metrics for analyzing the impact of parameters on the algorithm's performance.

(1) **Population Size (P)**: When the population size is small, the success rate of the genetic algorithm in searching for transformation sequences is low.

(2) **Gene Length (L)**: When the gene length is 8, the success rate of the genetic algorithm is 0%, indicating that the length of the transformation sequence may need to be greater than 8. However, excessively long transformation sequences lead to a significant increase in algorithm complexity and search space, thereby increasing the difficulty of the search and the running time. The experimental results show that when the gene length is 10 or 12, the algorithm achieves a good balance between success rate and running efficiency.

(3) **Number of Crossover Points (N)**: The number of crossover points has a relatively small impact on the search efficiency of the genetic algorithm, indicating that this parameter contributes less to the algorithm's performance within a reasonable range.

(4) **Mutation Rate (M)**: A lower mutation rate generally leads to higher running efficiency but may cause the algorithm to fall into local optima. On the other hand, a higher mutation rate may increase search diversity but also significantly increases running time.

Table 2 presents the parameter settings of the genetic algorithm.

Table 2 Genetic Algorithm Parameter Ranges.

Population Size	Gene Length	Number of Crossover Points	Mutation Rate	Maximum Iterations
250-1000	8-14	3-6	0.1-0.9	5000

5.3. Result Analysis

Table 3 present a set of solutions obtained by the genetic algorithm and the process of Andrews-Curtis transformations when the transformation sequence length (L) is 10, respectively.

Table 3 Andrews-Curtis Transformation Process (L=10).

Transformation Operation	Transformed Relators
Initial Group Presentation	$\langle x,y \mid XyyxYYY, XXXyyyxx \rangle$
1.$r_1 \to Xr_1x$	$\langle x,y \mid XyyxYYY, XXyyyx \rangle$
2.$r_1 \to Xr_1x$	$\langle x,y \mid XyyxYYY, Xyyy \rangle$
3.$r_0 \to Xr_0x$	$\langle x,y \mid XXyyxYYYx, Xyyy \rangle$
4.$r_0 \to r_0r_1$	$\langle x,y \mid XXyyx, Xyyy \rangle$
5.$r_0 \to xr_0X$	$\langle x,y \mid Xyy, Xyyy \rangle$
6.$r_1 \to r_1^{-1}$	$\langle x,y \mid Xyy, YYYx \rangle$
7.$r_1 \to r_1r_0$	$\langle x,y \mid Xyy, Y \rangle$
8.$r_0 \to r_0r_1$	$\langle x,y \mid \mathrm{X}y, Y \rangle$
9.$r_0 \to r_0r_1$	$\langle x,y \mid \mathrm{X}, Y \rangle$
10.$r_0 \to Xr_0x$	$\langle x,y \mid \mathrm{X}, Y \rangle$

ISSN 2616-5805 Vol. 6, Issue 2: 25-32, DOI: 10.25236/AJMS.2025.060204

Through the genetic algorithm's search, the group presentation H reduces the number of generators and their inverses in the relators to a minimum, proving that G'_{MS} can be transformed into the standard form of the trivial group via Andrews-Curtis transformations. This result demonstrates the effectiveness of the genetic algorithm-based triviality search algorithm in finding Andrews-Curtis transformation sequences.

6. Conclusion

This paper constructs an optimization model for the triviality search problem in group presentations. By defining decision variables, objective functions, and constraints, the complex problem of triviality in group presentations is transformed into an optimization problem. The paper also analyzes the scale and complexity of the problem, confirming its NP-hard nature, and proposes a genetic algorithm-based solution framework. Through experiments, the paper examines the impact of different parameter settings on the search efficiency of the genetic algorithm and verifies the effectiveness of the optimization model for the triviality search problem in group presentations. Future research can further explore the algorithmic solution steps to solve the optimization model of the triviality search problem in group presentations with higher efficiency.

Acknowledgements

The first author was supported by the NSF in China (Nos. 192162).

References

[1] Tietze H. Über die topologischen Invarianten mehrdimensionaler Mannigfaltigkeiten[J]. Monatshefte für Mathematik und Physik, 1908, 19: 1-118.

[2] Andrews J J, Curtis M L. Free groups and handlebodies[J]. Proceedings of the American Mathematical Society, 1965, 16(2): 192-195.

[3] Miasnikov A D, Myasnikov A G. Balanced presentations of the trivial group on two generators and the Andrews-Curtis conjecture[J]. Groups and Computation III,(W Kantor and A Seress, eds), de Gruyter, Berlin, 2001: 257-264.

[4] Myasnikov A D, Myasnikov A G, Shpilrain V. On the Andrews-Curtis equivalence[J]. Contemporary Mathematics, 2002, 296: 183-198.

[5] Miasnikov A D. Genetic algorithms and the Andrews-Curtis conjecture[EB/OL] (2003-04-30) [2024-05-23]. https://arxiv.org/abs/math/0304306.

[6] Bowman R S, Mccaul S B. Fast searching for Andrews-Curtis trivializations[J]. Experimental Mathematics, 2006, 15(2): 193-197.

[7] Swan J, Ochoa G, Kendall G, et al. Fitness Landscapes and the Andrews-Curtis Conjecture[J]. International Journal of Algebra and Computation, 2012, 22(2): 1-13.

[8] Krawiec K, Swan J. AC-trivialization proofs eliminating some potential counterexamples to the Andrews-Curtis conjecture[EB/OL]. (2015-9-1)[2024-05-23]. https://www.cs.put.poznan.pl/kkrawiec/-wiki/uploads/Site/ACsequences.pdf.

[9] Myropolska A. Andrews-Curtis and Nielsen equivalence relations on some infinite groups[J]. Journal of Group Theory, 2016, 19(1): 161-178.

[10] Hungerford T W. Algebra[M]. Berlin: Springer Science & Business Media, 2012: 1-157.