CayleyPy Growth: Efficient growth computations and hundreds of new conjectures on Cayley graphs

Chervov A. , Fedoriaka D., Obozov M.*, Konstantinova E., Naumov A., Kiselev I., Sheveleva A., Koltsov I., Lytkin S., Smolensky A., Soibelman A., Levkovich-Maslyuk F., Grimov R., Volovich D., Isakov A., Kostin A., Litvinov M., Vilkin-Krom N., Bidzhiev A., Krasnyi A., Evseev M., Geraseva E., Grunwald L., Galkin S., Koldunov E., Diner S., Chevychelov A., Kudasheva E., Sychev A., Kravchenko A., Kogan Z., Natyrova A., Shishina L., Cheldieva L., Zamkovoy V., Kovalenko D., Papulov O., Kudashev S., Shiltsov D., Turtayev R., Nikitina O., Mamayeva D., Nikolenko S., Titarenko A., Dolgorukova A., Aparnev A., Debeaupuis O., Alami C. S., Isambert H., Trusova E.

Abstract

We present the first public release of CayleyPy, an open-source Python library for working with Cayley and Schreier graphs. Compared to classical systems such as GAP and Sage, CayleyPy scales to much larger graphs and achieves speedups of several orders of magnitude.

Using CayleyPy we obtained about 200 new conjectures on diameters and growth of Cayley and Schreier graphs. For symmetric groups S_n we observe quasi-polynomial diameter formulas depending on $n \bmod s$, and conjecture this is a general phenomenon. This leads to efficient diameter computation despite NP-hardness in general. We refine Babai-type bounds for S_n, proposing $\frac{1}{2}n^2 + 4n$ as an upper bound in the standard case, and identify explicit generator families likely maximizing diameters, confirmed for $n \leq 15$. We also conjecture a closed formula for the diameter of the directed Cayley graph generated by the left cyclic shift and $(1, 2)$, answering a 1968 question of V.M. Glushkov.

For nilpotent groups we conjecture linear dependence of diameters on p in $\mathrm{UT}_n(\mathbb{Z}/p\mathbb{Z})$, improving results of Ellenberg, and observe Gaussian-type growth distributions akin to Diaconis' results for S_n.

Several conjectures are LLM-friendly, reducible to sorting problems verifiable via Python code. To foster benchmarking, we release 10+ Kaggle datasets for path-finding on Cayley graphs. CayleyPy supports arbitrary permutation and matrix groups with 100+ predefined generators, including puzzle groups. Its growth computation routines outperform GAP/Sage by up to 1000× in both speed and capacity.

*Research Center of the Artificial Intelligence Institute Innopolis University Innopolis, Russia

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: MATH-AI.

1 Introduction

Cayley graphs are fundamental in group theory Gromov (1993),Tao (2015), and have various applications: bioinformatics Hannenhalli & Pevzner (1995, 1999); Bulteau & Weller (2019); processor interconnection networks Akers & Krishnamurthy (1989); Cooperman et al. (1991); Heydemann (1997); coding theory and cryptography Hoory et al. (2006); Zémor (1994); Petit & Quisquater (2011); quantum computing Ruiz et al. (2024); Sarkar & Adhikari (2024); Dinur et al. (2023); Acevedo et al. (2006); Gromada (2022), etc.

There are many open conjectures in the subject and making progress in their understanding is a fundamental challenge in the field. Two of these that are quite well-known, easy to formulate, wide open and most relevant to us are:

- **Babai-like conjecture:** for any choices of generators the diameter of S_n is $O(n^2)$ (see, e.g., Helfgott & Seress (2014), Helfgott (2019), Helfgott et al. (2015));
- **Diaconis conjecture Diaconis (2013):** the mixing time for random walks is $O(n^3 \log n)$ (again for any choices of generators).

An important characteristic of a Cayley graph G is its *growth* — the vector of sizes of *spheres* (or *layers*) containing all elements at the same distance (length of the shortest path) from some fixed element $g_0 \in G$. It is easy to see that for Cayley graphs (in strict sense) growth does not depend on the choice of g_0. The growth of a graph G contains a lot of information on it: for example, the diameter is just the length of the growth vector minus one. It is suggestive to view growth as an unnormalized probability distribution over $\mathbb{N}$, as in P. Diaconis' works. Then the diameter is just the maximum of a random variable. And it is important to understand its other characteristics: its mean, mode, moments, etc. Ideally, the goal is to understand from what family of distributions it comes, and, hopefully, to observe some universality phenomena, such as the distribution approaching something known for large values of n. For example, the Gaussian normal distribution trivially arises in the case of abelian groups, while in the case of S_n with generators close to commutative — like Coxeter's neighbor transpositions $(i, i+1)$ — the appearance of the Gaussian normal approximation has been demonstrated by P. Diaconis.

So, having some elements in, say, the permutation group S_n (or in some other group), one constructs the Cayley graph, and there is a set of natural questions and lines of investigation:

- What group is obtained?
- What is its diameter?
- Growth statistical characteristics: mean, mode, moments, what distribution does it follow (or at least asymptotically as $n \to \infty$)?
- Algorithm: is there an effective/polynomial algorithm which decomposes a given element into a product of generators (optimally/sub-optimally)?
- Antipodes ("super-flips"): is there an explicit description of the longest elements?
- What can be said about the graph's spectrum?
- What is the mixing time?

2 Main contribution

The aim of the present paper is to make progress on of fundamental problems described above (i.e. understanding various properties of Cayley graphs) with the help of the new tool which we are developing: AI-based Python open-source library, CayleyPy, which allows to make computational experiments orders of magnitude more effectively than standard computer algebra systems GAP/SAGE. We show that the pipelines which were introduced previously for some specific S_n sub-groups are scaling quite well for Cayley graph tasks Furthermore, due to the permutation-like structure of most problems they can be formulated as sorting problems, which are easy to formulate for LLM, and their solutions can be given as an algorithm or Python code, are easy to verify, so they can be used to test LLM's abilities to solve research problems. Meanwhile, our code for direct growth computation outperforms similar functions on the standard computer algebra system GAP/SAGE up to 1000 times both in speed and in maximum sizes of the graphs that it can handle.

- We generate around 200 conjectures on various properties of Cayley graphs, that is achieved by extensive computational experiments with around 50 Cayley graphs. The conjectures are summarized in tables 1,**??**.
- In particular we propose the following:
  - We conjecture that diameters of many S_n-Cayley graphs are quasi-polynomials (quadratic/linear) in n (i.e. several polynomials depending on n modulo some s) allowing to find them rather efficiently, which is surprising since it is NP-hard in general.
  - The improvement of the L.Babai-like conjecture for S_n - diameters are bounded by $n^2/2 + 4n$, by $3n^2/4 + O(n)$ (directed cases), $n^2/4 + O(n)$ for some Schreier graphs, comparing to prior $O(n^2)$ conjectural bounds. Moreover we present explicit families of generators for S_n which conjecturally provide largest (or near) diameters. They are related to involutions and follow rather simple pattern ("square-with-whiskers"). They were found by an extensive (partly exhaustive) search for $n <= 15$ of the generators with maximum diameter.
  - For nilpotent groups we conjecture improvement of J.S. Ellenberg's results on diameter of upper-triangular matrices over Z/p presenting phenomena of linear dependence of diameter on p. Moreover growth for nilpotent groups conjectured to follow Gaussian distributions (a central limit phenomena - similar to results of P.Diaconis for S_n).
  - We present a conjectural answer on the open question: diameter of the directed Cayley graph generated by left cyclic shift and transposition $(1, 2)$ is equal to $(3n^2 + 8n + 9)/4$ for odd n, else $(3n^2 - 8n + 12)/4$.
- To benchmark various methods of path-finding on Cayley graphs and LLMs we create 11 benchmark datasets in the form of Kaggle challenges, making benchmarking easy and public to community.

2.1 CayleyPy features

CayleyPy is an AI-based open-source Python library which can work with googol size graphs, with the current main focus on mathematical tasks for Cayley graphs of finite groups. The key current goals are: using the AI approach (Chervov et al. (2025a,b)) to solve path-finding tasks on Cayley graphs, or in other words to decompose a given group element into a product of generators (solution of Rubik's cube is an illustrative example).

More generally one is interested in understanding various properties of Cayley graphs: their diameters, growth, spectrum, random walks mixing time, etc. CayleyPy provides a framework to handle all these tasks in a simple manner accessible to non-experts in programming, but utilizing the power of efficient code, algorithms and GPU accelerators. For graphs of not too large size (up to trillions) one can compute the diameter and growth effectively, for smaller sizes one can also compute the spectrum and generate visualizations, and so on.

CayleyPy can work with arbitrary permutation and matrix groups, which the user defines as an input. Moreover, it also supports a large collection (more than a hundred) of predefined generators of permutation groups (including various puzzles) and matrix groups.

The main outcome of that stage of the project is that we were able to generate hundreds of new mathematical conjectures on Cayley/Schreier graphs via extensive computational experiments with CayleyPy. Thus, we demonstrate that effective computational tools can advance discoveries in pure mathematics.

2.2 Diameter quasi-polynomiality conjecture

Having discussed the main features of our library CayleyPy, let us now turn to our main mathematical results. We start with the following definition. A function $f\colon \mathbb{N} \to \mathbb{N}$ is a quasi-polynomial if there exist polynomials $p_0, \dots, p_{s-1}$ such that

$$f(n) = p_i(n) \quad \text{when } i \equiv n \pmod{s}.$$

The polynomials p_i are called the *constituents* of f.

The following conjecture generalizes the results of extensive computations we performed (see the next section):

(Extremely optimistic). For any generators of S_n (or A_n) which can be constructed by an algorithm with say polynomial complexity in n (e.g. a Python function which takes as input n and outputs generators in time polynomial in n) the diameter of the Cayley graph will be given by some quadratic or linear quasi-polynomial in n (at least for n large enough). Even more optimistically, the leading terms of all constituents coincide.

Experimentally we see an even more general phenomenon: not only the distance to the longest element (i.e. the diameter) is quasi-polynomial, but also the distance to many other elements. Thus, it is tempting to propose the following:

(Extremely optimistic). For any generators of S_n (or A_n) and additionally elements $g_n \in S_n$ (or A_n) which can be constructed by an algorithm with say polynomial complexity in n (e.g. a Python function which takes as input n and outputs generators jointly with elements g_n in time polynomial in n) the distance from identity to g_n (i.e. "word metric" of g_n) will be given by some quadratic or linear quasi-polynomial in n, at least for n large enough. Even more optimistically, the leading terms of all constituents coincide.

The two conjectures do not seem to imply each other in general, however in practice we often see that the longest elements ("antipodes", "superflips") can be often described quite effectively. In such cases the second conjecture implies the first one.

Generalizations to other groups are also plausible, e.g. to Coxeter's groups and the generalized symmetric group (examples will be given below). Even for matrix groups one may hope to have a similar quadratic/linear quasi-polynomial dependence on n (i.e. on the "rank"). And similarly for Schreier coset graphs, for example $S_n/(S_{\lfloor n/\ell \rfloor} \times S_{n-\lfloor n/\ell \rfloor})$ (i.e. "Grassmanians over the field with one element"), and other similar cosets like flags over F_1. However, in that case we typically choose some node and compute the most distant node to it (not exactly the diameter), and in contrast to a Cayley graph that distance ("God's number" in puzzle's terminology) can depend on the choice of the node. We expect quasi-polynomiality for all choices of the starting node. Another direction of generalization is weighted Cayley graphs, in particular circular Cayley graphs, i.e. permutations factorized by cyclic shifts (see e.g. Adin et al. (2025)), which represent weighted Cayley graphs with weights of cyclic shifts set to zero.

All known to us examples of the explicitly computed diameters are indeed quasi-polynomial, although they may be written in a slightly different way in the literature, e.g. in terms of rounding functions floor or ceil which are just particular simple cases of quasi-polynomials. In our analysis we observe more complicated examples with the modulo s equal to e.g. 4, 6, or 8.

Examples (known). For Coxeter generators ($(i, i+1), 1 \leq i < n$) of S_n the diameter is $n(n-1)/2$ – just polynomial (and similarly for all Coxeter's groups). As a small modification one can add the transposition $(1, n)$ to the Coxeter generators – this gives the diameter $\left\lfloor \frac{n^2}{4} \right\rfloor$ van Zuylen et al. (2016), which is $n^2/4$ for even n, and $(n^2-1)/4$ for odd n, i.e. quasi-polynomial with modulo 2. (A similar (but very recent Adin et al. (2025)) circular version gives $\left\lfloor \frac{(n-1)^2}{4} \right\rfloor$). For all transpositions (i, j) the diameter is $n-1$, while for transpositions of the form $(1, i)$ (star graph) it is $\left\lfloor \frac{3(n-1)}{2} \right\rfloor$, which is $3(n-1)/2$ for odd n, and $(3n-2)/2$ for even – a linear quasi-polynomial.

Examples (conjectural). Consider the LRX generators: L – left cyclic shift, R – right cyclic shift, $X = (1, 2)$. The diameter is conjectured to be $n(n-1)/2$ (OEIS-A186783), strong support for this is presented in our previous work Chervov et al. (2025b). The LX case with only two of these generators (L and X) has been first considered by V.M.Glushkov Glushkov (1968) and studied by his school (survey: Glukhov & Zubov (1999) pages 18-21). Our conjecture is that the diameter is $(3n^2 - 8n + 9)/4$ for n odd, and $(3n^2 - 8n + 12)/4$ for n even, discussed below in detail.

Consider consecutive 4-cycles: $(i, i+1, i+2, i+3), i \leq 1 \leq n-3$, and their action on the coset $S_n/(S_{\lfloor n/2 \rfloor} \times S_{n-\lfloor n/2 \rfloor})$ which is just the action on binary vectors with 0 and 1 having $\lfloor n/2 \rfloor$ zeros. Choose the vector with first $\lfloor n/2 \rfloor$ zeros as the starting vector and compute the most distant element with respect to that initial vector (it is not exactly the diameter in general, but its analog depending on choice of initial node, and can be called "God's number" like in puzzles). We expect that for

$n \geq 12$ the God's number is given by quasi-polynomials modulo 6: $n^2/12+1, n \equiv 0 \mod 6$, $(n^2+4n-5)/12, n \equiv 1 \mod 6$, etc.

3 Conclusion

In this paper, we present the CayleyPy library and propose approximately 200 conjectures on Cayley graphs using it. We provide a comprehensive comparison with the GAP method in terms of the computational time required for the growth of different groups. We also emphasize that the conjectures we obtained, together with our Kaggle challenges, may constitute an effective benchmark for both reinforcement learning and large language model algorithms.

4 Acknowledgements

The study was supported by the Ministry of Economic Development of the Russian Federation (agreement No. 139-10-2025-034 dd. 19.06.2025, IGK 000000C313925P4D0002)

A Experimental results

We conducted extensive computations computing growth for large number of Cayley and certain Schreier coset graphs for up to $n \leq 15$ and $n \leq 42$ respectively. Obtained results and conjectures are summarized in the tables discussed below, which also include results known in the literature.

We analyzed not only diameters but other growth characteristics as a probability distribution - mean, mode, variance, skewness, kurtosis, tried to fit the distribution by some known like Gaussian or Gumbel, antipodes (longest elements, or super-flips), spectrum of the Cayley graph. In some cases we observe that growth by itself might have close analytical formula - given by Stirling numbers or related to Fibonacci numbers or e.g. coincide with some known sequence e.g. OEIS-A367270 ("Growth/F-la" column of the table). For diameters in most (but not all) cases we able to fit by quasi-polynomials, for some cases, apparently, available data is not enough. But for mean diameters and other characteristics of growth there are not quasi-polynomials in general, for example literature contains results $n - log(n)$ for mean diameters, Nevertheless we expect that our numerical fits for the data provides approximations to the leading terms of these characteristics. They are obtained as fit on for small values of n and can be considered as conjectures for large values.

Notations used in the table:

1. ◆ - conjecture obtained by CayleyPy project, ◆ - proved by CayleyPy
2. $+$ information is known/conjectured - can be found in the main text (too big to fit into table)
3. $*$ - conjecture from the literature
4. ? - no information, neither in literature, nor our experiments suggest clear pattern
5. notations like $1|2$ indicates 1 for even n and 2 for odd (or vice versa)
6. notations $+I$ - some quasi-polynomial typically of zero degree
7. "Group" information on generated group, if just $+$ information is known, but not fits into the table
8. "Growth/PDF" - what continuous distribution fits growth for large n (Gaussian, or Gumbel, etc)
9. "Growth/F-la" - explicit formulas for the growth
10. "Antipode" - information on longest elements - i.e. if there explicit description, if the number is known (and simple to fit into table) we indicate it, or simple write $+$
11. "Algorithm" - indicates is there known algorithm to decompose element into product of these generators, the upper-script O indicates that optimal algorithm is known, notations like $NP/2$ means that optimal decomposition was proved to be NP-hard, but there polynomial approximations by factor of 2.
12. "Metric" - is there explicit expression for the word metric for given generators, for example for Coxeter generators it is a number of inversions

13. "Spectrum" - information on spectrum, "Int" integer spectrum, "Wig" - approaches Wigner semi-circle law for large n, "Uni" - almost uniform

Table 1: Summary of properties of Cayley graphs

Generators	**Group**	**Diameter**	**Growth: PDF**	**F-la**	**Mean**	**Mode**	**Var**	**Skew**	**Kurt**	**Antipodes**	**Algorithm**	**Metric**	**Spectrum**	**Mixing Time**
Coxeter	S_n	$\frac{n(n-1)}{2}$	Gauss	+	n(n-1)/4	n(n-1)/4	$+$	$\to 0$	$\to 0$	1	Bubble	$+$	?	?
Cyclic Coxeter	S_n	$\left\lfloor \frac{n^2}{4} \right\rfloor$	Gauss◆	?	$0.17(n^2-n+1)$◆	≈Mean◆	?	$\to 0$◆	$\to 0$◆	1|2◆	?	+	Wig◆	?
LRX	S_n	$\frac{n(n-1)}{2}*$	Gumbel◆	?	$\approx 0.38n^2-n$◆	$\approx 0.39n^2-n$◆	?	$\to -0.7$◆	$\to 3.3$◆	◆	◆	?	Uni◆	$> n^3$◆
LX-Glushkov	S_n	$\frac{3n^2-8n+9\|12}{4}$◆	Gumbel	◆Fib /?	$\approx 0.57n^2-2n$◆	$\approx 0.57n^2-1.6n$◆	?	$\to -0.7$◆	$\to 0.5$◆	*	◆	?	?	?
LARX	S_n	$\frac{n^2-2\|5}{2}$◆	?	?	$0.4n^2-0.7n$◆	≈Mean◆	?	?	?	+◆|?	?	?	?	?
LARX+I	S_n	$\frac{n(n+6)-12\|19}{4}$◆	?	?	$\approx \frac{n(n+1)}{4}$◆	≈Mean◆	?	?	?	+◆	?	?	?	?
LSL	S_n	$\frac{n(n-3)}{2}+3$◆	?	?	$\approx 0.4n^2-1.5n$◆	≈Mean◆	?	?	?	+◆	?	?	?	?
LSL+I	S_n	$\frac{n(n+4)}{4}-3\|4.25$◆	?	?	$\approx 0.2n^2$	$\approx 0.2n^2$	?	?	?	?	?	?	?	?
3-cyc	A_n	$\left\lfloor \frac{n}{2} \right\rfloor$	?	+−	$\approx D-0.5\|1$◆	$D-0\|1$◆	?	?	?	+	+	+	Int	?
(0ij)	A_n	$\left\lfloor \frac{3(n-1)}{4} \right\rfloor$	?	?	$\approx 0.55n$◆	$\approx Mean$◆	?	?	?	?	?	?	?	?
(01i)	A_n	$\frac{3n-5}{2}+\frac{i^n+(-i)^n}{4}$◆	?	?	$\approx n-2$◆	$n-1$◆	?	?	?	?	+◆	?	?	?
(01i)I	A_n	$\left\lfloor \frac{3n-6}{2} \right\rfloor$◆	?	?	$\approx n+1.25ln(n)+\ldots$◆	≈Mean◆	◆	◆	◆	◆	◆^{O}	◆	Int◆	?
(i,i+1,i+2)	A_n	$\left\lfloor \frac{n^2+1}{4} \right\rfloor$◆	Gauss◆	?	$\approx D/2$◆	$\approx D/2$◆	$\approx \frac{n^3}{100}$	$\to 0$◆	$\to 0$◆	?	?	?	?	?
(i,i+1,i+2)I	A_n	$\left\lfloor \frac{n(n-1)}{4} \right\rfloor$◆	Gauss◆	?	$\approx D/2$◆	$\approx D/2$◆	$\approx \frac{n^3}{100}$◆	$\to 0$◆	$\to 0$◆	+◆	?	?	?	?
(i...i+3)	S_n	$\approx 0.3n^2$◆	?	?	$\approx 0.16n^2$◆	$\approx 0.15n^2$◆	?	≈ 0◆	≈ 0◆	?	?	?	?	?
(i...i+3)I	S_n	$\approx 0.16n^2$◆	?	?	$\approx 0.036n^2$◆	$\approx 0.045n^2$◆	?	?	?	?	?	?	?	?
(i,i+1,i+2)C	A_n	$\approx \frac{n(n+2)}{8}+I$◆		?	$\approx 0.085n^2$◆	$\approx 0.065n^2$◆	?	?	?		?	?	?	?
(i,i+1,i+2)CI	A_n	$\left\lfloor \frac{n^2}{8} \right\rfloor$◆	?	?	$\approx 0.08n^2$◆	$\approx 0.086n^2$◆	?	?	?	+◆	?	?	?	?
(i...i+3)C	S_n	?	?	?	?	?	?	?	?	?	?	?	?	?
(i...i+3)CI	S_n	?	?	?	?	?	?	?	?	?	?	?	?	?
Pref.cyc	S_n	$n-1$												
Pref.cyc+I	S_n	$n-1$◆	?	?							?	?	?	?
Down.cyc	S_n	$n-1$◆	?	Stirling	$\approx 0.86n$◆	≈Mean◆	?	?	?	+◆	?	?	?	?

Continued on next page

Generators	Group	Diameter	Growth: PDF	F-la	Mean	Mode	Var	Skew	Kurt	Antipodes	Algorithm	Metric	Spectrum	Mixing Time
Down.cyc+I	S_n	$n-1$◆	?	?	$\approx 0.75n$◆	≈Mean◆	?	?	?	+◆	?	?	?	?
Inc.3cyc	A_n	$n-1\|2$◆	?	?	$\approx 0.5n$◆	≈Mean◆	?	?	?	$+-$◆	?	?	?	?
Inc.4cyc	S_n	$n-1$◆	?	?	$\approx 0.5n$◆	≈Mean◆	?	?	?	?	?	?	?	?
RapaportM1	S_n	$\lceil \frac{3n}{2} \rceil$◆	?	?	$\approx 1.4n$◆	≈Mean◆	?	?	?	?	?	?	?	?
RapaportM2	S_n	$\approx \frac{n^2+n}{2}$◆	?	?	$\approx \frac{n^2-3n}{2}$◆	≈Mean◆	?	$\to -0.6$◆	$\to 0.5$◆	?	?	?	?	?
Globes n/1	+	$\lceil \frac{5n+12}{4} \rceil$◆	?	?	$\approx n+1$◆	$\approx n+1$◆	?	?	?	?	?	?	?	?
3PancakeS_1	S_n	$\frac{3n(n+2)-48+I}{8}$◆	?	?	◆		?	?	?		?	?	?	?
3PancakeS_2	S_n	◆	?	?	◆		?	?	?		?	?	?	?
3PancakeS_3	S_n	◆	?	?	◆		?	?	?		?	?	?	?
3PancakeS_4	S_n	◆	?	?	◆		?	?	?		?	?	?	?
3PancakeS_5	S_n	◆	?	?	◆		?	?	?		?	?	?	?
3PancakeS_6	S_n	$\frac{3n(n+2)-48}{8}$◆	?	?	◆		?	?	?		?	?	?	?
3PancakeS_7	S_n	◆	?	?	◆		?	?	?		?	?	?	?
Pancake	S_n	$\approx 1.2n?$	?	?	$< \frac{17n}{12}$	$\approx n/2*$	$\approx 0.2n?$	?	?	$+-$	$NP/2$?	$-$	?	?
Reversals	S_n	$n-1$	?	?	$\approx n/2$	$\approx Mean?$	$\approx 0.05n$?	?	?	+	$NP/1.5$	$-$	?	?
sReversals	B_n	$n-1$	?	?	$\approx n-\frac{nlogn}{2}$	$\approx Mean?$	$\approx 0.05n$	?	?	$+-$	+	$+-$	?	?
Transposons	S_n	$\lceil \frac{n+1}{2} \rceil *$	?	?	$\Theta(n)$	$\Theta(n)$		?	?	$-$	$NP/1.375$		?	?
(i,j)	S_n	$n-1$	?	Stirling	$\approx n-lnn$	$\approx n-lnn$	$\approx lnn$	$\approx \frac{1}{\sqrt{lnn}}$	$\approx \frac{1}{lnn}$	+	+	+	Int	nlnn
$(1,i)$(Star)	S_n	$\lfloor \frac{3(n-1)}{2} \rfloor$	?	?	$\approx n-lnn$	$\approx n-lnn$	$\approx lnn$			+	+	+	Int	?
G-star transp.	S_n	+	?	?	?	?	?	?	?	?	?	?	?	?

End of table

A.1 Distribution of diameter over conjugacy classes pairs - involutions strikes

Figure 1 represents diameters dependence on the conjugacy classes of generators obtained by exhaustive search: one generator from one class, another from another. Presented figure is for S_7. The values in each cell represent all values of the diameters found for corresponding pair of classes and the heatmap coloring is done with respect to the maximal found diameter for corresponding pair of classes. Color is white if S_n is not generated by any pair of elements from the above classes. Due to symmetry we are showing only relevant part of data, not duplicating $g1, g2$ with $g2, g1$.

One can see that large diameters appears when one of the classes is involution, and another has rather short cycles in decomposition. Which is simple to expect, because the longer cycles are present - means individual generators would have higher degree and allow to create more words like: $XX...XX$. More words of fixed length k one has - the less diameter can be, since words exhausts finite space of group elements earlier. So naively it is easy to expect that generators related to as small degree as possible are good candidates. Exhaustive search confirms it for many cases. So:

The generating set with largest diameter for S_n contains an involution (at least for infinite number of n). Both for directed and undirected cases.

We performed similar exhaustive search up to $n \leq 9$ and randomized search up to 12 – for the pairs of generators and for both undirected and directed cases, results are consistent with the conjecture. Similar heatmap plots for other n can be found in full version the manuscript; or up to $n \leq 7$ in the notebook.

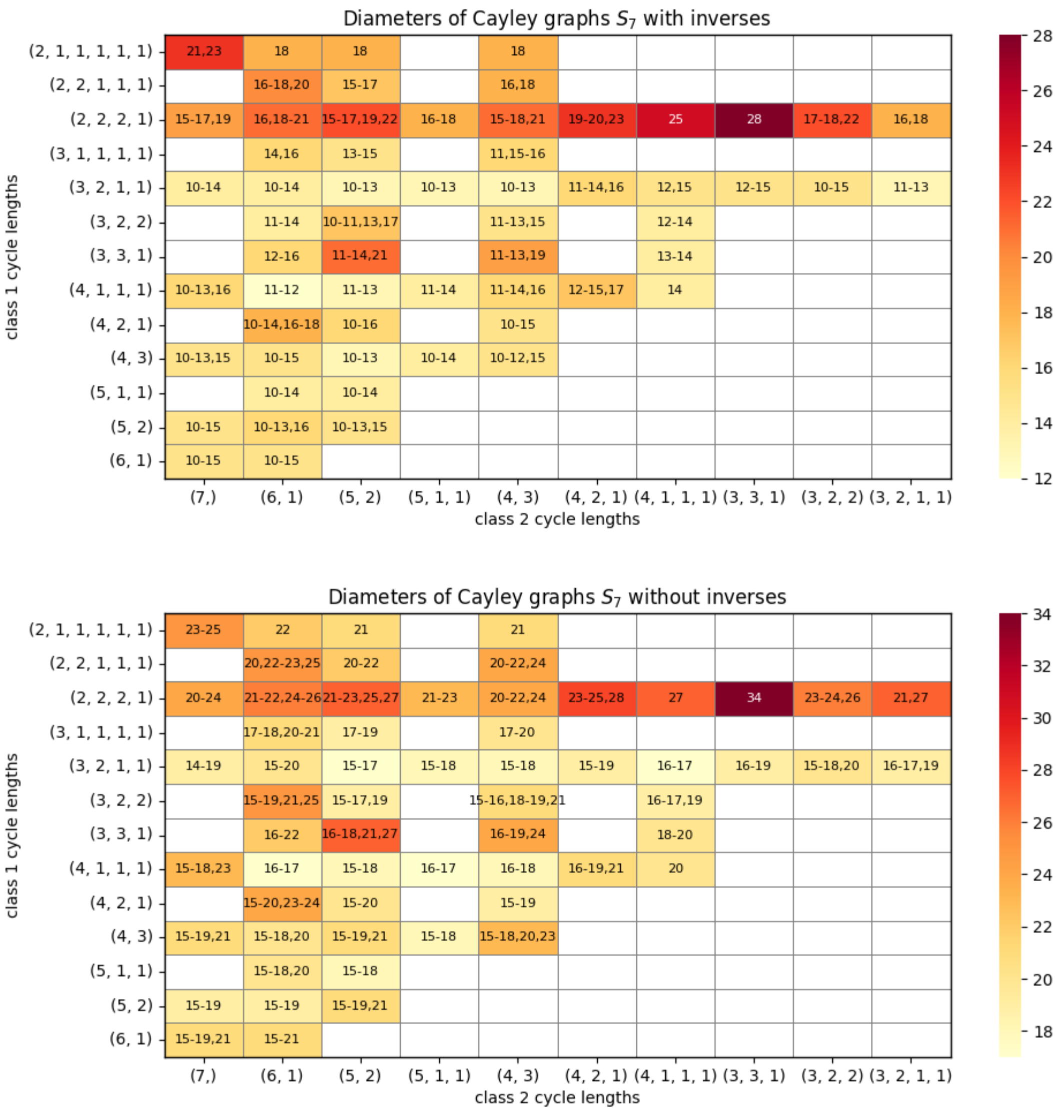

Figure 1: Diameters of all possible Cayley graphs for S_7 generated by two permutations with/without their inverses.

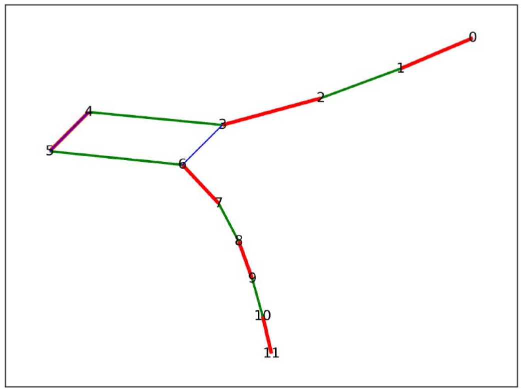

Figure 2: Three generators of S_{12} represented graphically. Pattern - "square with whiskers". Three involutions - edges orientations are not necessary.

A.2 Graphical visual representation for any generators - pattern "square with whiskers"

Here we describe a very simple graphical visual representation of elements (e.g. generators) of permutation groups, and present a pattern "square with whiskers" which corresponds to most of the largest diameters found for $n \leq 15$.

Step 1. Single permutation. Each permutation defines a directed graph on n nodes in a natural and obvious way – if $p(i) = j$ let us connect $i \to j$. (That can be said in the other words - take a permutation matrix and consider it as adjacency matrix of a directed graph). Clearly if permutation is involution - then orientation of edges is unnecessary - if $i \to j$, then $j \to i$ (in matrix language - permutation matrix is symmetric).

Step 2. Many permutations - use colors. Consider several permutations and just use the same construction but use different colors to represent edges coming from different permutations.

Thus for any sequence of permutations (generators) we constructed a directed multi-colored graph on n nodes.

The code for the visualization can be found e.g. in the notebook.

The figure 2 presents an example of such visualization and also presents an example of the pattern which we call "square with whiskers" - there is one 4-cycle (square) and two branches going out of its corners.

Let us call the generators to follow "square with whiskers" pattern if underlying undirected graph (forgetting colors and multi-edges) is of that type - one 4-cycle (square) and two branches.

The generators with maximal (or nearly) diameter for S_n/A_n follow "square with whiskers" pattern (at least for infinite number of n). We expect that to be true for the both for undirected and directed case of Cayley graphs.

B Largest diameters found (and known) for $n <= 15$

We conducted extensive search for generators producing large diameters for small n, remarkable patterns showed up - that will be discussed in the next section, here we just present the diameters and the corresponding generators, and organization of the experiments. But already here it is worth to highlight that all found generators are related to involutions.

To the best of our knowledge these are the largest diameters known so far. We consider both standard undirected and directed cases (meaning that generators are not necessarily inverse closed). For the latter case the same diameters up to $n \leq 7$ were found in Egri-Nagy & Gebhardt (2016) (table 4), but our result extends to much larger n.

The diameters provided below are largest possible or at least within say 5% from them. It is impossible to make exhaustive search even for such values of n, so we cannot exclude the chance that large diameters exists, though it seems unlikely to us that they will be significantly larger than presented here. Anyway we hope our results may stimulate that research.

Maximal diameter for Cayley graph of the group S_n (undirected graph)		
n	Maximal diameter	Example of a set of generators
3	3	$[0,2,1],[2,1,0]$
4	6	$[2,1,0,3],[3,0,2,1],[1,3,2,0]$
5	10	$[4,0,1,2,3],[1,2,3,4,0],[0,1,3,2,4]$
6	15 (16)	$[4,5,3,2,0,1],[2,5,0,3,1,4],[2,4,0,3,5,1]$ $([[0,1,2,3,5,4],[0,2,1,4,3,5],[1,0,3,2,5,4]])$
7	28 (30)	$[1,0,3,2,5,4,6],[2,6,5,3,1,0,4],[5,4,0,3,6,2,1]$ $([0,1,3,2,4,6,5],[0,4,6,5,1,3,2],[6,1,3,2,5,4,0])$
8	33 (39)	$[3,7,5,6,0,2,4,1],[4,7,5,0,6,2,3,1],[1,0,3,2,5,4,6,7]$ $([0,1,2,3,5,4,7,6],[0,1,3,2,6,7,4,5],[7,3,6,1,4,5,2,0])$
9	(52)	$([8,5,2,7,4,1,6,3,0],[1,2,0,5,3,4,8,6,7],[2,0,1,4,5,3,7,8,6])$
10	(77)	$([0,1,2,3,5,4,7,6,9,8],[1,0,3,2,5,4,8,9,6,7],$ $[0,6,4,8,2,5,1,7,3,9])$
11	(85)	$([1,0,3,2,5,4,7,6,9,8,10],[0,3,4,1,2,6,5,8,7,10,9],$ $[0,4,3,2,1,5,6,7,8,9,10])$
12	(95)	$([1,0,3,2,5,4,7,6,9,8,11,10],[0,2,1,5,6,3,4,8,7,10,9,11],$ $[0,1,2,6,5,4,3,7,8,9,10,11])$
13	(111)	$([1,0,2,5,4,3,7,6,9,8,11,10,12],[0,2,3,4,1,6,5,8,7,10,9,12,11],$ $[0,4,1,2,3,6,5,8,7,10,9,12,11])$
14	(132)	$([1,0,3,2,5,4,7,6,9,8,11,10,13,12],$ $[0,2,1,4,3,6,5,8,7,10,9,12,11,13],$ $[0,1,2,3,5,4,6,7,8,9,10,11,12,13])$
15	(148)	$([1,0,3,2,5,4,7,6,9,8,11,10,13,12,14]$, $[0,2,1,4,3,6,5,8,7,10,9,12,11,14,13]$, $[0,1,2,3,4,8,7,6,5,9,10,11,12,13,14])$

Maximal diameter for Cayley graph of the group S_n (oriented graph)		
n	Maximal diameter	Example of a set of generators
3	3	$(01),(02)$
4	7	$(01),(123)$
5	14	$(01)(23),(0314)$
6	18	$(01)(23)(45),(012)(34)$
7	34	$(01)(23)(45),(052)(146)$
8	44	$(01)(23)(45),(1736)(25)$
9	61	$(01)(23)(45),(3647)(12)(58)$
10	83	$(01)(23)(45)(67)(89),(185)(237)(469)$
11	93	$(01)(23)(45)(67)(89),(1528)(47)(6,10)$
12	106	$(01)(23)(45)(67)(89),(1,11,9,10)(04)(28)(36)$
13	147	$(01)(23)(45)(67)(89),(1,11,2,12)(34)(56)(78)(9,10)$

For the directed case (i.e. not inverse closed generators) the search has been organized as follows - for small $n = 3, 4$ we considered several possible numbers of generators - from 2 to 5, it was observed that largest diameter is observed for 2 generators - which is not surprising, since less generators - less words can one generate and large diameter can potentially be. We continue search with 2 generators for n up to 13. The search has been - exhaustive up to $n \leq 9$, and randomized search after. We did not search all possible pairs, but relied on a simple fact that conjugacy of all generators produces an isomorphic graph, so we say first generator can be taken as a unique representative of conjugacy class, with the loop over conjugacy classes - that of course significantly reduce the search space. For the randomized search we sampled the second generator from each conjugacy class uniformly. To reduce search further for $n \geq 12$ we mostly considered conjugacy classes for the first generator to be involutions, and used guesses from previously observed patterns.

C Benchmarks for growth computations

CayleyPy utilizes GPU out of the box, for example, even on old GPU P100 it can compute growth for S_{11} with Coxeter generators in 0.6 seconds, while GAP takes 2352 seconds, so CayleyPy is more than 1000 times faster in that example. Performance depends on group size and number of generators. The tables below demonstrate that for large groups starting from S_8 CayleyPy is typically 10-100 times faster than GAP, even when using CPU. Moreover, it can support larger groups. Below are the results obtained and easily reproducible on the Kaggle cloud where we can perform computations up to S_{13}. For S_{14}, S_{15} we use more powerful machines. The group S_{14} requires around $40 - 100$ GB RAM and $4 - 20$ hours of computations. The group S_{15} requires $500 - 1000$ GB RAM, and computations take several days. (For S_{15} we mainly worked with a small number of generators like 3, while a larger number of generators (e.g. 15) may take months.)

Tomas Rokicki and Lucas Garron's program "Twsearch" (example on Kaggle) apparently is faster than our growth computations for CPU. However our code supports GPU out of the box, and apparently can achieve better timing using modern GPU. Also our framework seems to be more user-friendly, supports directed Cayley graphs and achieves computations for large groups like S_{15}, which, apparently, is not yet achieved by Twsearch.

CayleyPy supports several algorithms for growth computation all based on BFS (breadth first search), but different in internal data representation. The bfs bitmask uses bit-wise encoding with 3 bits per any state, and it is more memory efficient. It allows one to work with S_{13} requiring only $3 - 8$ GB RAM. We use it for computations for S_{13}, S_{14}, S_{15}. See also the notebooks: algorithm, benchmarks.

In the tables below, the label "(bm)" identifies uses of the "bfs-bitmask" algorithm.

Table 2: Growth computations. Time in seconds for CayleyPy and GAP using CPU on Kaggle cloud, 32 GB RAM. Different types of generators.

	LRX		Coxeter		Transpositions		Pancake		Reversals	
n	GAP	CayleyPy	GAP	CayleyPy	GAP	CayleyPy	GAP	CayleyPy	GAP	CayleyPy
6	1.97	0.02	0.01	0.038	0.03	0.02	0.01	0.02	0.04	0.02
7	2.04	0.03	0.12	0.052	0.34	0.03	0.08	0.02	0.41	0.05
8	2.66	0.05	1.32	0.079	3.92	0.09	0.80	0.04	4.24	0.1
9	8.75	0.16	13.99	0.293	52.98	1.03	8.90	0.26	54.45	1.089
10	75.72	1.24	172.38	3.54	737.42	14.14	112.37	3.77	762.54	14.8
11	884	22.5	2352	41	10891	296	1559	72	11138	354
12	12024	658	34470	706	-	3331(bm)	35387	4923	-	26657
13	-	2670(bm)	-	7308(bm)	-	$>$ 12h	-	17694(bm)	-	>12h

Table 3: CayleyPy growth computations on different hardware. Time in seconds: CPU (32G), GPU (16G), and advanced CPU (330G), on Kaggle cloud. Coxeter generators.

Coxeter CayleyPy				
n	CPU	GPU T4	GPU P100	CPU (at TPU v3-8)
4	0.013	0.013	0.014	0.010
5	0.029	0.026	0.026	0.022
6	0.038	0.043	0.043	0.037
7	0.052	0.068	0.067	0.067
8	0.079	0.151	0.138	0.087
9	0.293	0.110	0.110	0.140
10	3.539	0.199	0.154	0.641
11	41/41(bm)	1.085	0.601	10.8/12(bm)
12	705/512(bm)	214(bm)	207(bm)	187/151(bm)
13	7308(bm)	3004(bm)	2867(bm)	2613/2180(bm)

D Details

D.1 LLM statement

LLMs were used for:

- generating potential solutions to benchmark current state-of-art systems against human results;
- code assistance in human submissions.

D.2 Reproducibility statement

The present article was written using Overleaf.

The code was executed using Kaggle, Colab or GCP (TPU Research Cloud, v4_32 instances) environments, using the following packages:

- pandas
- matplotlib
- numpy
- cayleypy
- torch (torch_xla for TPU)
- numba
- scipy

The code is available in an anonymous GitHub repository.

References

O. L. Acevedo, J. Roland, and N. J. Cerf. Exploring scalar quantum walks on cayley graphs, 2006.

R. M. Adin, N. Alon, and Y. Roichman. Circular sorting, 2025.

S. B. Akers and B. Krishnamurthy. A group-theoretic model for symmetric interconnection networks. *IEEE transactions on Computers*, 38(4):555–566, 1989.

L. Bulteau and M. Weller. Parameterized algorithms in bioinformatics: an overview. *Algorithms*, 12 (12):256, 2019.

A. Chervov, K. Khoruzhii, N. Bukhal, J. Naghiyev, V. Zamkovoy, I. Koltsov, and A. Romanov. A machine learning approach that beats large rubik's cubes, 2025a.

A. Chervov, A. Soibelman, S. Lytkin, I. Kiselev, S. Fironov, A. Lukyanenko, A. Dolgorukova, A. Ogurtsov, F. Petrov, S. Krymskii, M. Evseev, L. Grunvald, D. Gorodkov, G. Antiufeev, G. Verbii, V. Zamkovoy, L. Cheldieva, I. Koltsov, A. Sychev, M. Obozov, A. Eliseev, S. Nikolenko, N. Narynbaev, R. Turtayev, N. Rokotyan, S. Kovalev, V. Rozanov, S. Nelin, L. Ermilov, D. Shishina, A. Mamayeva, K. Korolkova, A. Khoruzhii, and A. Romanov. Cayleypy rl: Pathfinding and reinforcement learning on cayley graphs. `https://arxiv.org/abs/2502.18663`, 2025b.

G. Cooperman, L. Finkelstein, and N. Sarawagi. Applications of cayley graphs. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes: 8th International Conference, AAECC-8 Tokyo, Japan, August 20–24, 1990 Proceedings 8*, pp. 367–378. Springer Berlin Heidelberg, 1991.

P. Diaconis. Some things we've learned (about markov chain monte carlo), 2013.

I. Dinur, M. H. Hsieh, T. C. Lin, and T. Vidick. Good quantum ldpc codes with linear time decoders. In *Proceedings of the 55th annual ACM symposium on theory of computing*, pp. 905–918, 2023.

A. Egri-Nagy and V. Gebhardt. Computational enumeration of independent generating sets of finite symmetric groups, 2016.

M. M. Glukhov and A.Yu. Zubov. On the lengths of symmetric and alternating permutation groups in various generating systems (review). *Mathematical questions on cybernetics*, 8:5–32, 1999. (In Russian). `https://keldysh.ru/papers/1999/mvk/mvk1999_5.pdf`, `https://www.kaggle.com/datasets/alexandervc/cayleypy-development-3-growth-computations/data?select=Glukhov_Zubov1999review.pdf`.

V. M. Glushkov. Completeness of a system of operations in digital computers. *Cybernetics and Systems Analysis*, 4(2):1–5, 1968. `https://link.springer.com/article/10.1007/BF01073731`.

D. Gromada. Some examples of quantum graphs. *Letters in Mathematical Physics*, 112(6):122, 2022.

M. Gromov. *Geometric Group Theory: Asymptotic invariants of infinite groups*, volume 2. Cambridge University Press, 1993.

S. Hannenhalli and P. A. Pevzner. Transforming men into mice (polynomial algorithm for genomic distance problem). In *Proceedings of IEEE 36th annual foundations of computer science*, pp. 581–592. IEEE, 1995.

S. Hannenhalli and P. A. Pevzner. Transforming cabbage into turnip: polynomial algorithm for sorting signed permutations by reversals. *Journal of the ACM (JACM)*, 46(1):1–27, 1999.

H. A. Helfgott. Growth in linear algebraic groups and permutation groups: towards a unified perspective. In *Groups St Andrews 2017 in Birmingham*, volume 455, pp. 300, 2019.

H. A. Helfgott and Á. Seress. On the diameter of permutation groups. *Annals of mathematics*, pp. 611–658, 2014.

H. A. Helfgott, Á. Seress, and A. Zuk. Random generators of the symmetric group: diameter, mixing time and spectral gap. *Journal of Algebra*, 421:349–368, 2015.

M. C. Heydemann. Cayley graphs and interconnection networks. In *Graph symmetry: algebraic methods and applications*, pp. 167–224. Springer Netherlands, 1997.

S. Hoory, N. Linial, and A. Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006.

C. Petit and J. J. Quisquater. Rubik's for cryptographers. Cryptology ePrint Archive, 2011.

F. J. Ruiz, T. Laakkonen, J. Bausch, M. Balog, M. Barekatain, F. J. Heras, and P. Kohli. Quantum circuit optimization with alphatensor. `https://arxiv.org/abs/2402.14396`, 2024.

R. S. Sarkar and B. Adhikari. Quantum circuit model for discrete-time three-state quantum walks on cayley graphs. *Physical Review A*, 110(1):012617, 2024.

T. Tao. *Expansion in finite simple groups of Lie type*, volume 164. American Mathematical Soc., 2015.

A. van Zuylen, J. Bieron, F. Schalekamp, and G. Yu. A tight upper bound on the number of cyclically adjacent transpositions to sort a permutation. *Information Processing Letters*, 116:718–722, 2016.

G. Zémor. Hash functions and cayley graphs. *Designs, Codes and Cryptography*, 4(3):381–394, 1994.