Enumeration and Generation of Simple Unlabelled Graphs

UNIVERSITY OF WARWICK

Jonathan Glanfield   Supervisor: Levente Bodnár

Warwick Mathematics Institute

Graph Enumeration and Pólya's Enumeration Theorem

- Enumerating unlabelled graphs is a classical and challenging problem in combinatorics. Unlike labelled graphs, where each vertex has a distinct identity, unlabelled graphs consider graphs equivalent under relabelling of vertices, making their enumeration significantly more complex.
- This task is known to be #P-complete, meaning it is as computationally difficult as counting the solutions to NP problems, and no efficient algorithm is known for solving it in the general case [1].

Burnside's Lemma

Burnside's Lemma relates the number of distinct orbits under a group action to fixed points of group elements:

$$|X/G| = \frac{1}{|G|}\sum_{g\in G}|X^g|,$$

where $X^g = \{x \in X : g \cdot x = x\}$ is the set fixed by g.

This lemma allows counting objects up to symmetry by averaging fixed points across group elements.

Cycle Index Polynomial

The **cycle index polynomial** encodes the structure of a permutation group G acting on a set of size n:

$$Z(G, x_1, x_2, \dots, x_n) = \frac{1}{|G|}\sum_{g\in G}\prod_{j=1}^{n} x_j^{c_j(g)},$$

where $c_j(g)$ counts cycles of length j in the permutation g.

This polynomial compactly summarises how group elements permute cycles of different lengths.

Pólya's Enumeration Theorem

Pólya's Theorem [2] states that for a group G acting on a set X and a set of q colours A, the number of distinct colourings up to G-symmetry is:

$$\left|A^X/G\right| = Z(G, q, q, \dots, q).$$

For the purposes of graph enumeration, edges are coloured by two colours (either present or absent), and the group acts on vertex labels inducing permutations on edges.

Example: Counting Unlabelled Graphs on 4 Vertices

For $n = 4$, the symmetric group S_4 acts on the vertex set, inducing an action on the $\binom{4}{2} = 6$ edges. Using the conjugacy classes of S_4, we compute the induced cycle index polynomial for $S_4^{(2)}$:

Permutation in S_4	Class Size	Permutation in $S_4^{(2)}$	Monomial	Value for $s_k = 2$
$(1)(2)(3)(4)$	1	$(12)(13)(14)(23)(24)(34)$	s_1^6	$1 \cdot 2^6 = 64$
$(1)(2)(34)$	6	$(12)(34)(13\,14)(23\,24)$	$s_1^2 s_2^2$	$6 \cdot 2^4 = 96$
$(1)(234)$	8	$(12\,13\,14)(23\,34\,24)$	s_3^2	$8 \cdot 2^2 = 32$
$(12)(34)$	3	$(12)(34)(13\,14)(23\,24)$	$s_1^2 s_2^2$	$3 \cdot 2^4 = 48$
(1234)	6	$(13\,24)(12\,23\,34\,14)$	$s_2 s_4$	$6 \cdot 2^2 = 24$

Collecting terms gives the cycle index $Z\big(S_4^{(2)}\big) \;=\; \frac{1}{24}\Big(s_1^6 + 9\,s_1^2s_2^2 + 8\,s_3^2 + 6\,s_2s_4\Big)$

Then by substituting $s_k = 2$ (two colours: edge present/absent) we have: $\frac{1}{24}\big(2^6 + 9\cdot 2^4 + 8\cdot 2^2 + 6\cdot 2^2\big) \;=\; 11.$

Result: There are exactly **11** distinct simple unlabelled graphs on 4 vertices.

Recursive Generation of Simple Unlabelled Graphs

- We can generate all n-vertex graphs by extending the complete list $\mathcal{G}_{n-1}$ of non-isomorphic graphs on $n-1$ vertices.
- For each base graph $G \in \mathcal{G}_{n-1}$, we only consider new vertex neighbourhoods up to $\mathrm{Aut}(G)$ (one representative per orbit). This eliminates many redundant extensions before an computationally intensive isomorphism test.

Orbit Representatives & Graph Extension

Let $V = [n-1]$ be the vertex set of G and let $\mathrm{Aut}(G)$ act on the power set $\mathcal{P}(V)$ by

$$\gamma \cdot S = \{\gamma(v) \;:\; v \in S\}, \qquad \gamma \in \mathrm{Aut}(G).$$

This partitions $\mathcal{P}(V)$ into orbits; choose one representative set from each orbit:

$$\mathcal{S}_G = \{S_1, \dots, S_m\} \;=\; \mathcal{P}(V)/\mathrm{Aut}(G).$$

For each representative $S \in \mathcal{S}_G$ define the extension

$$G_{\{S\}} = (V \cup \{n\},\; E \cup \{\{u,n\} : u \in S\}).$$

Repeating this for all $G \in \mathcal{G}_{n-1}$ produces a candidate collection $\mathcal{E}_n$.

Removing Isomorphic Duplicates

Different base graphs may produce isomorphic extensions. To obtain the final list $\mathcal{G}_n$, we must check for graph isomorphisms and retain only unique graphs (e.g. using **McKay's Canonical Labelling Algorithm** [3]), keeping a single representative for each canonical label:

$$\mathcal{G}_n \;=\; \mathcal{E}_n \,/\, \cong .$$

This two-stage approach (orbit pruning then canonical labelling) can drastically reduce any redundant isomorphism checks in practice.

Graph Generation Algorithm

Input: List $\mathcal{G}_{n-1}$ of all graphs on $n-1$ vertices with their automorphism groups
Output: List $\mathcal{G}_n$ of all graphs on n vertices

$\mathcal{E}_n \leftarrow \emptyset$;
foreach $G \in \mathcal{G}_{n-1}$ **do**
  compute $\mathrm{Aut}(G)$;
  foreach *representative* $S \in \mathcal{P}(V) \,/\, \mathrm{Aut}(G)$ **do**
    declare $G_{\{S\}} \leftarrow G \cup \{n\}$ with $N(n) = S$;
    add $G_{\{S\}}$ to $\mathcal{E}_n$;
  end
end
$\mathcal{G}_n \leftarrow \mathcal{E}_n \,/\cong$ using McKay's canonical labelling algorithm;
return $\mathcal{G}_n$

Running Time and Complexity

- **Worst case:** If G is asymmetric then $\left|\mathcal{P}(V)/\mathrm{Aut}(G)\right| = 2^{n-1}$, so

$$|\mathcal{E}_n| \le |\mathcal{G}_{n-1}| \cdot 2^{n-1}.$$

  In addition, McKay's canonical labelling may require up to $O(n!)$ steps, giving a worst-case runtime of $O(g_{n-1}\, n!)$.
- **Average case:** Since almost all large graphs are asymmetric [4],

$$\mathbb{E}[|\mathcal{E}_n|] = (1 - o(1))\, g_{n-1}\, 2^{\,n-1}.$$

  McKay's exhibits a polynomial average number of labelling steps [3], hence average runtime of $O(g_{n-1}\, 2^{n-1}\, \mathrm{poly}(n))$.

Scan for Report

References

[1] L. G. Valiant. "The complexity of enumeration and reliability problems". In: *SIAM Journal on Computing* 8.3 (1979), pp. 410–421.

[2] G. Pólya. "Kombinatorische Anzahlbestimmungen für Gruppen, Graphen und chemische Verbindungen". In: *Acta Mathematica* 68 (1937), pp. 145–254.

[3] B. D. McKay. "Practical graph isomorphism". In: *Congressus Numerantium* 30 (1981), pp. 45–87.

[4] P. Erdős and A. Rényi. "Asymmetric graphs". In: *Acta Math. Acad. Sci. Hungarica* 14.3–4 (1963), pp. 295–315.