AMSI VACATIONRESEARCH SCHOLARSHIPS 2021–22

Get a taste for Research this Summer

Towards a Uniform Sampling Procedure for Abstract Triangulations of Surfaces

Rajan Shankar

Supervised by Dr Jonathan Spreer

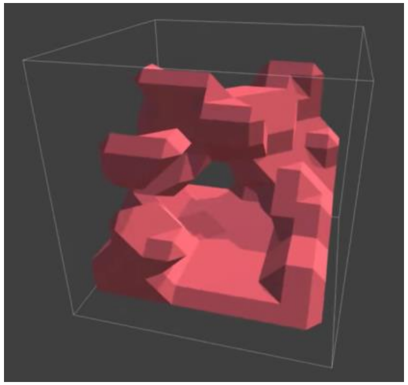
The University of Sydney

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE

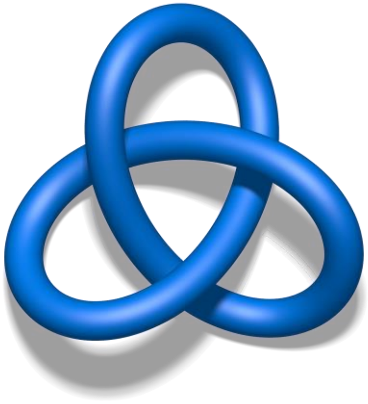
AMSI CONNECT

Why study surfaces?

- Marching cubes algorithm
  - MRI scans
  - Terrain rendering
  - Computer graphics

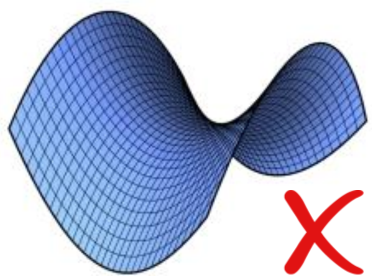
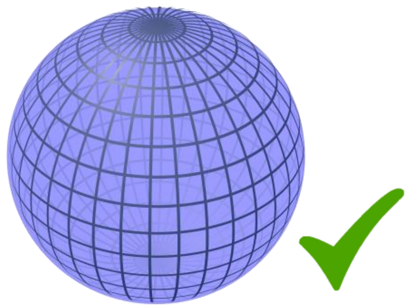
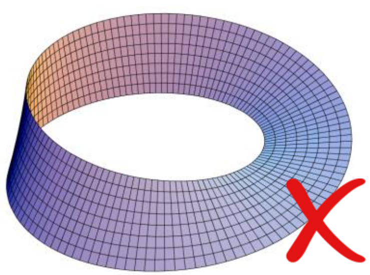
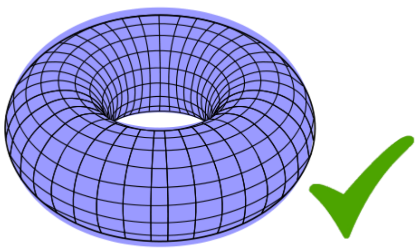


- Applications in pure math
  - Surface embeddings in 3-manifolds
  - Knot exteriors
  - Graphs on surfaces



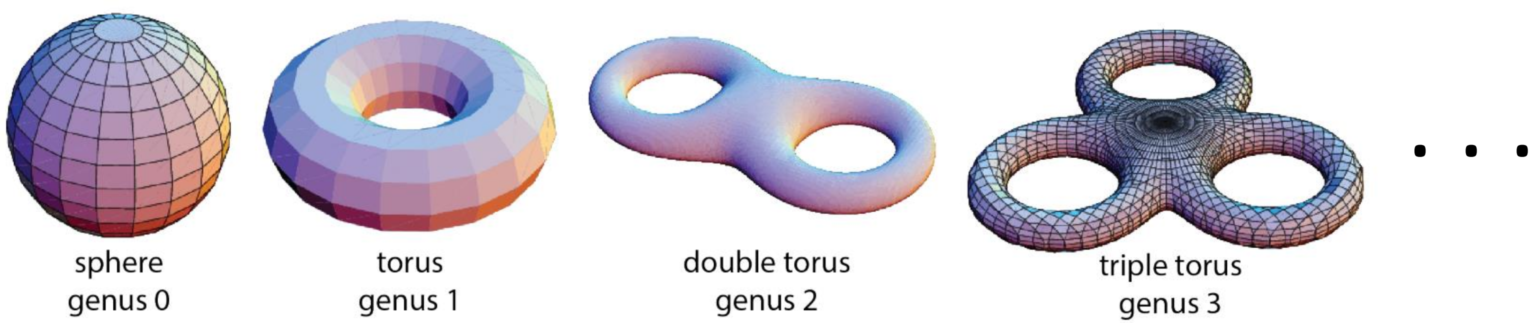
The surfaces we focus on

- Closed and orientable
- Results in a nicer, well-defined problem

Genus of a surface

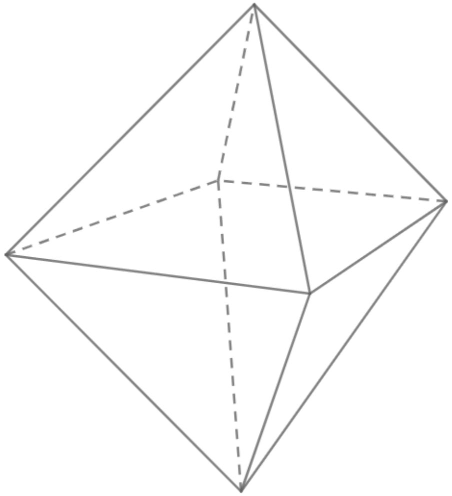
- Up to continuous deformation, every closed and orientable surface is one of these surfaces:



Abstract triangulation of a surface

- Any closed, orientable surface can be decomposed into $2n$ triangles
- *Abstract* refers to the lack of a coordinate system
- Triangulations allow us to solve problems on surfaces using a computer

The octahedron is one of the triangulations of the sphere when $n = 4$



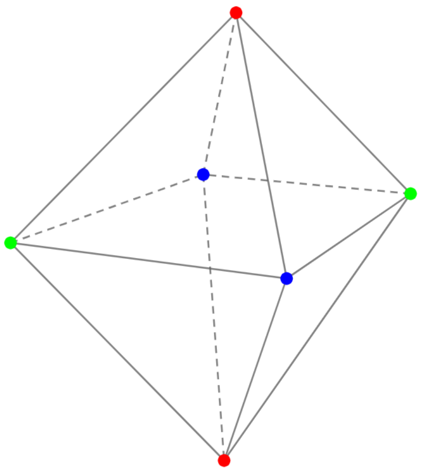
Graph-encoded manifold (GEM)

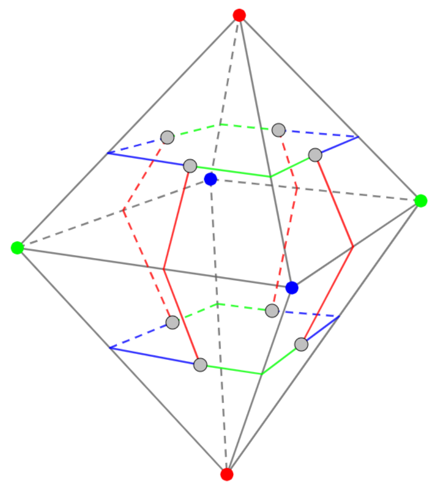
- A triangulation can be uniquely encoded using a coloured graph

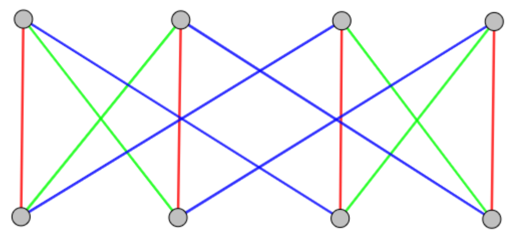
1. Colour the vertices of the triangulation using three colours

2. Thinking of each triangle as a "node", connect the triangles via coloured arcs

3. Rearrange the coloured arcs into standard form

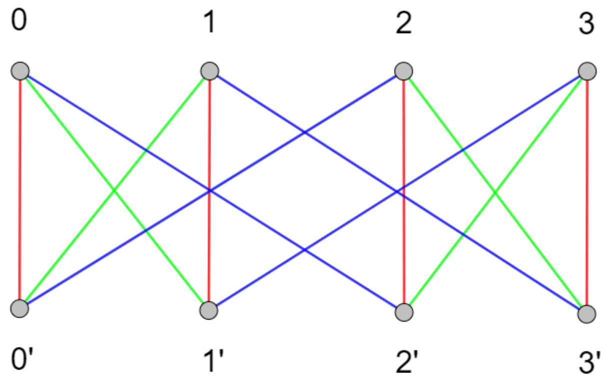






Permutations

- We can exploit the bipartite property of a GEM to represent it using a pair of permutations

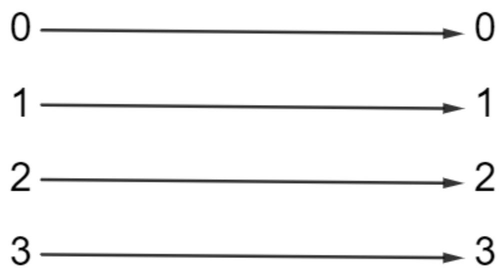


$$\mu = (0\ 1)(2\ 3)$$
$$\sigma = (0\ 2)(1\ 3)$$

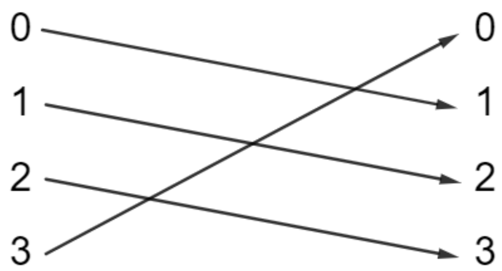
- This enables us to sample triangulations by sampling permutations!

Aside: cycle notation

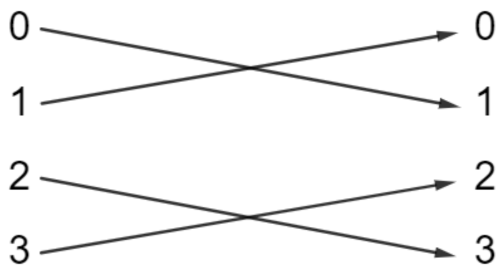
- Write out permutations in compact ways
- Easily identify cycles and the lengths of these cycles



$(0)(1)(2)(3)$



$(0\ 1\ 2\ 3)$



$(0\ 1)(2\ 3)$

Why uniform sampling?

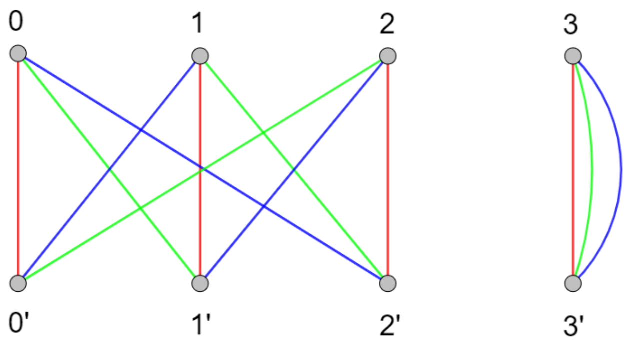
- Recall: uniformly sample from space of combinatorial types
- There exist algorithms that solve problems on triangulations
- However, we do not know what an "average" triangulation looks like
- So, we cannot conduct runtime analyses on these algorithms
- A uniform sampling procedure would enable us to understand the "average" triangulation

Uniform sampling procedure #1

1. Uniformly sample a permutation μ from S_n
2. Uniformly sample a permutation σ from S_n
3. Compute GEM
4. Add triangulation $T_{\mu,\sigma}$ to the sample

Disconnected GEMs

- Not every μ, σ pair corresponds to a valid GEM



$$\mu = (0\ 1\ 2)(3)$$
$$\sigma = (0\ 2\ 1)(3)$$

- Fortunately, it is easy to identify this by checking if the subgroup generated by μ, σ is transitive

Isomorphisms and symmetries

- Multiple μ, σ pairs can correspond to the same GEM
- Isomorphisms: number of "disguises" for a GEM
- Symmetries: number of copies of each unique disguise
- We want to know the number of <u>unique</u> disguises
  - Call this the *weight*

Uniform sampling procedure #2

1. Uniformly sample a permutation μ from S_n
2. Uniformly sample a permutation σ from S_n
3. Check if the subgroup generated by μ, σ is transitive
   - If not transitive, go back to step 1
4. Compute GEM
5. Compute symmetries and then the weight
6. Add triangulation $T_{\mu,\sigma}$ to the sample with the computed weight

Sampling partitions of an integer

- There is a one-to-one correspondence between an integer partition and the cycle structure of a permutation (i.e. the conjugacy classes of the symmetric group)

Partition	$1+1+1$	$2+1$	3
Permutation	$(0)(1)(2)$	$(0\ 1)(2)$	$(0\ 1\ 2)$

- Instead of sampling μ from the whole set of permutations, it is enough to sample μ from the <u>set of cycle structures</u>
- Benefits
  - Greatly reduces space of μ, σ pairs: $n! \times n! \rightarrow e^{\sqrt{n}} \times n!$
  - Makes it easier to calculate the weights

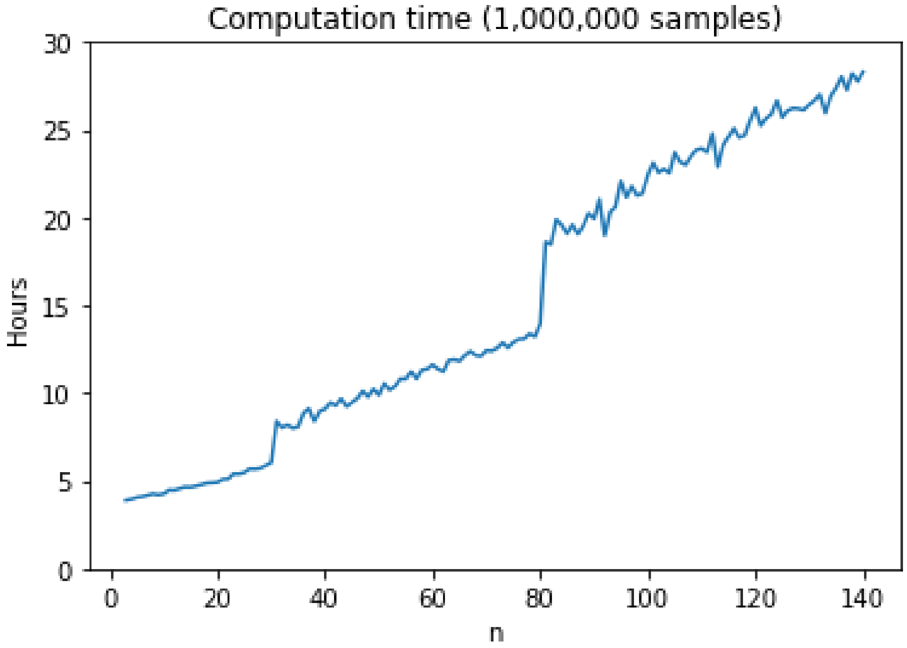
Uniform sampling procedure #3

1. Uniformly sample a partition λ of n
   - Then, convert λ into its corresponding permutation μ
2. Uniformly sample a permutation σ from S_n
3. Check if the subgroup generated by μ, σ is transitive
   - If not transitive, go back to step 1
4. Compute GEM
5. Compute symmetries and then the weight
6. Add triangulation $T_{\mu,\sigma}$ to the sample with the computed weight

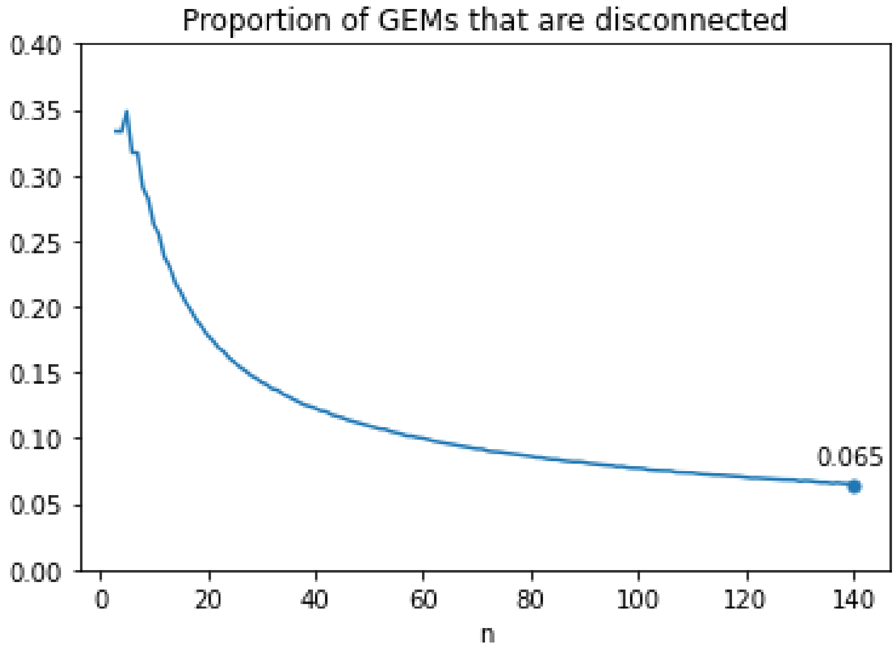
Python implementation

- Coded a triangulation class in Python
  - Input parameters are permutations μ and σ
  - Methods for computing genus, gem, symmetries and weight
- Wrote a script to perform uniform sampling
  - Ran it on a powerful computer
  - 140,000,000 samples
- Added in extra code to collect data of interest
- Will apply hypothesis tests to the data to form conjectures

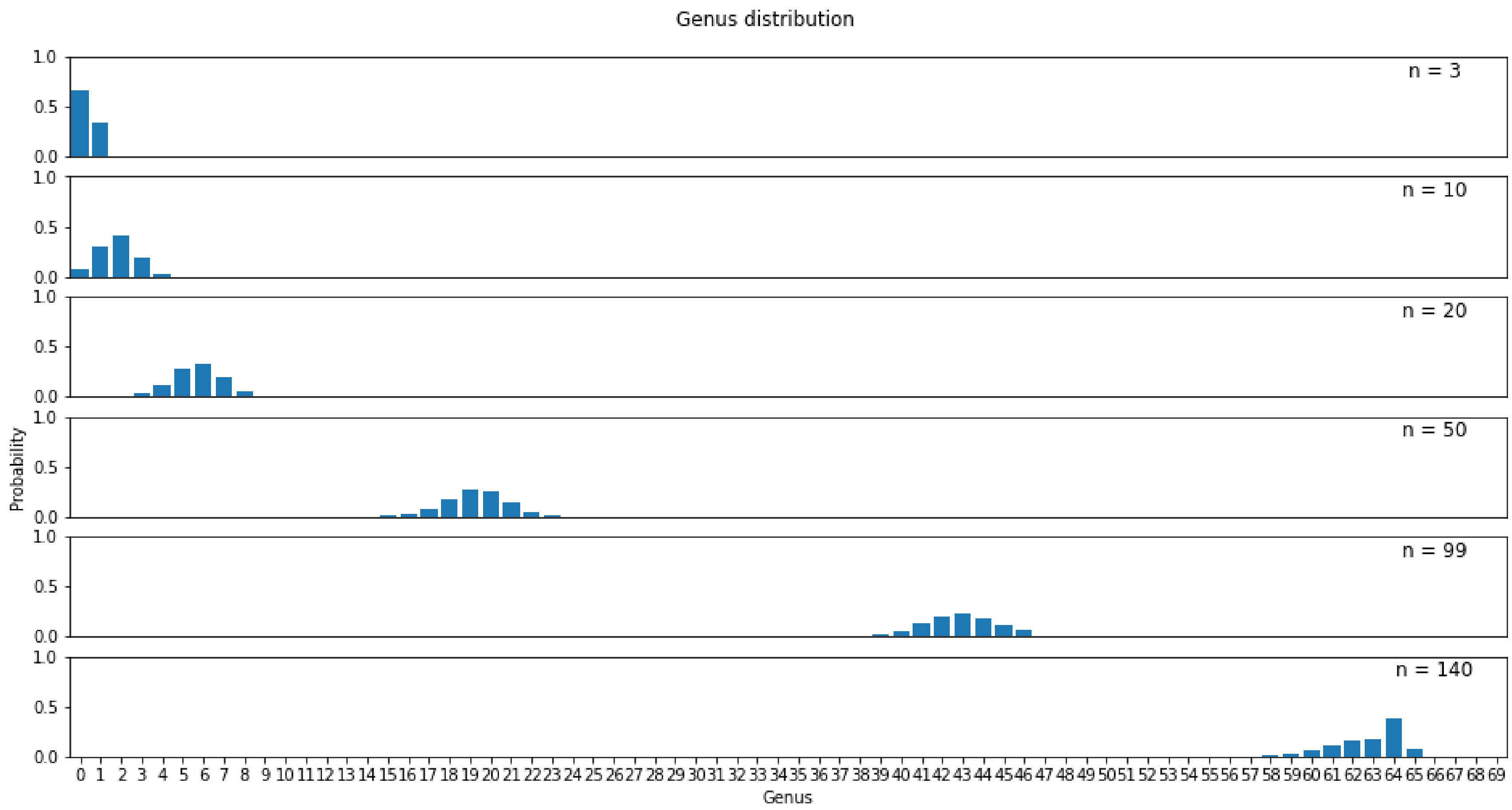
Computation times



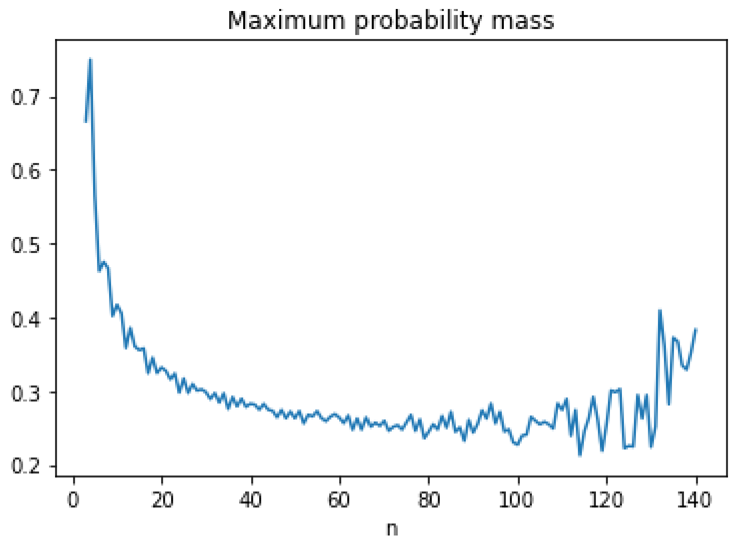
Experimental results

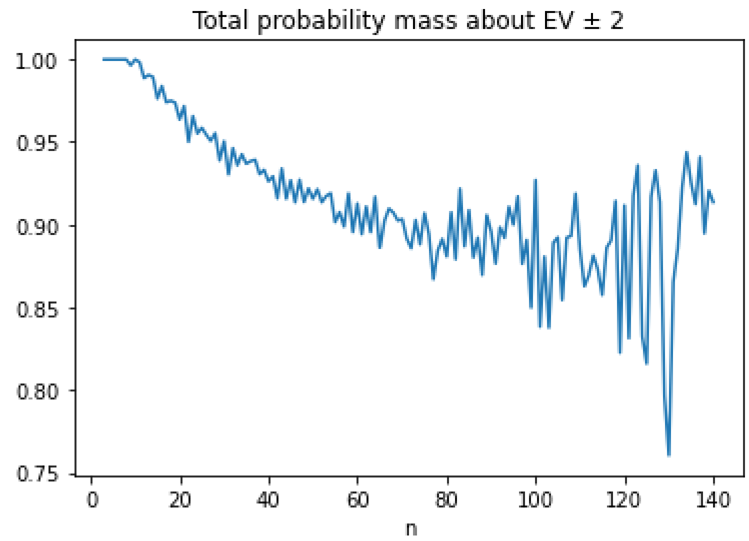


Experimental results

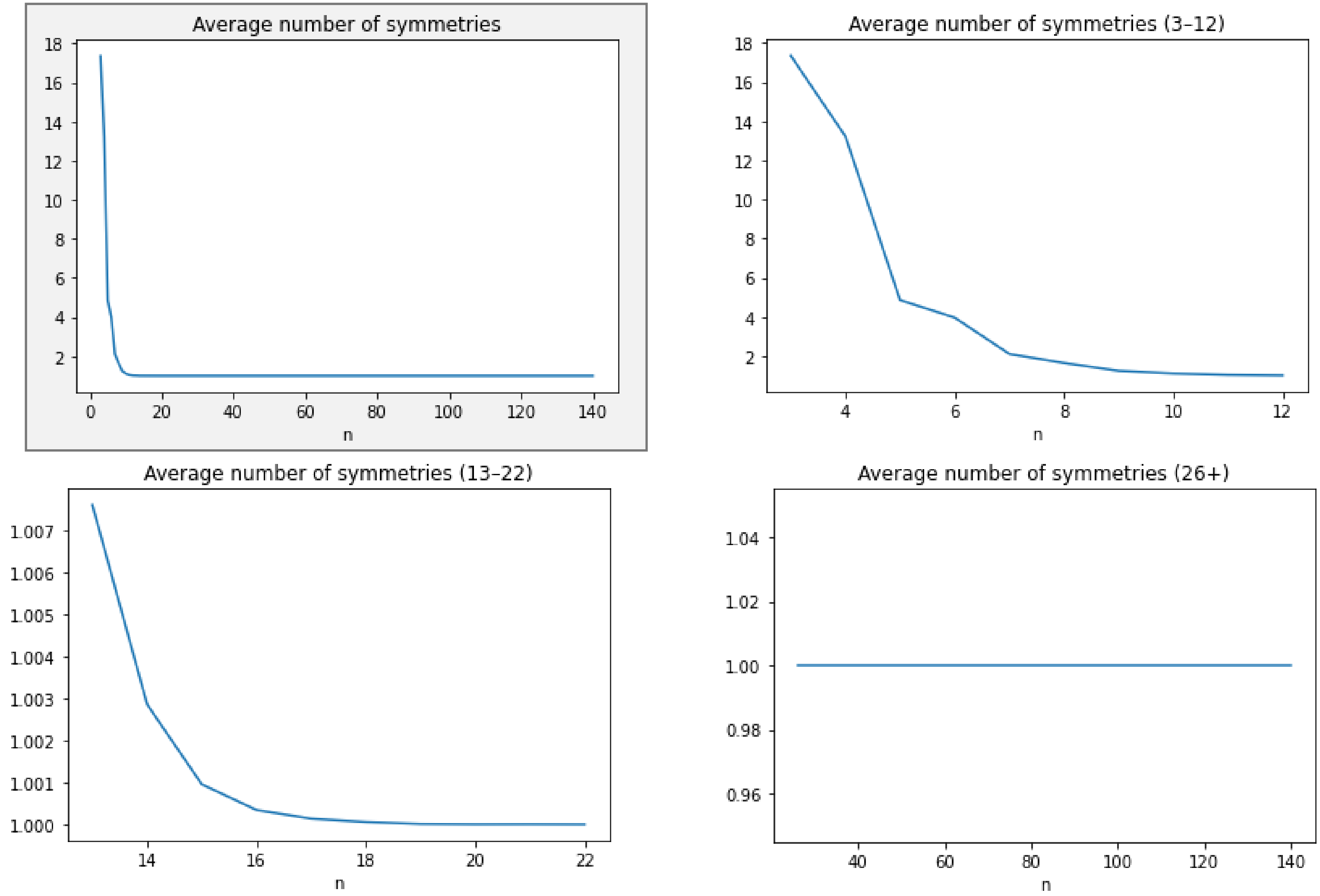


Experimental results





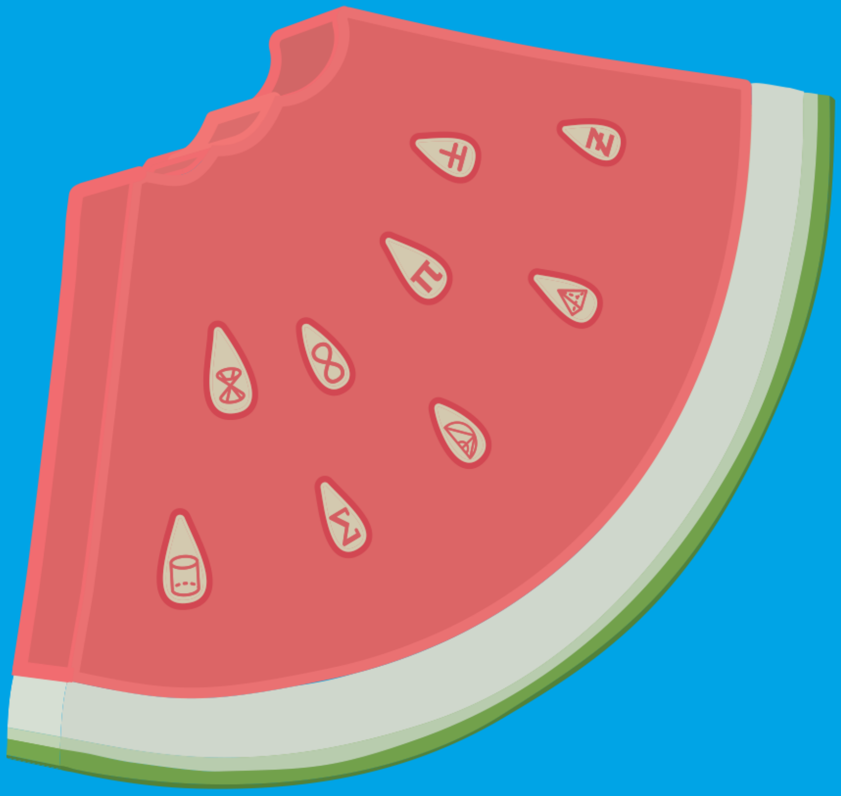
Experimental results



AMSI

VACATIONRESEARCH SCHOLARSHIPS 2021–22

Get a taste for Research this Summer



AMSI
CONNECT

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE