# Exam project: Enumerate all neighborly 4-polytopes on 8 vertices

Alba Delgado Petar Hlad Rebecca Guardado
Simon Van den Eynde

January 15, 2017

## 1 Introduction

Our goal is to solve the following problem:

**Problem 1.** Enumerate all simplicial neighbourly 4-polytopes on 8 vertices up to combinatorial equivalence.

The idea of how we will solve this problem is the following. We start from all possible affine Gale diagrams of a 4-polytope on 8 vertices. These are point configurations of 8 points in the plane, each with an assigned positive or negative sign. We check which of these affine Gale diagrams correspond to neighbourly, simplicial polytopes, and consider only these configurations. Then, check whether they correspond to the same polytope (i.e. to a combinatorially equivalent polytope) and once we have all different Gale diagrams, retrieve the polytope they correspond to.

## 2 Results

In figure 1 we show all neighbourly Gale diagrams we found.

After filtering on isomorphisms, we found 3 isomorphism classes. For each of these classes we give the coordinates of 1 polytope in figure 2. These are 3 different neighbourly simplicial 4-polytopes on 8 vertices up to combinatorial equivalence.

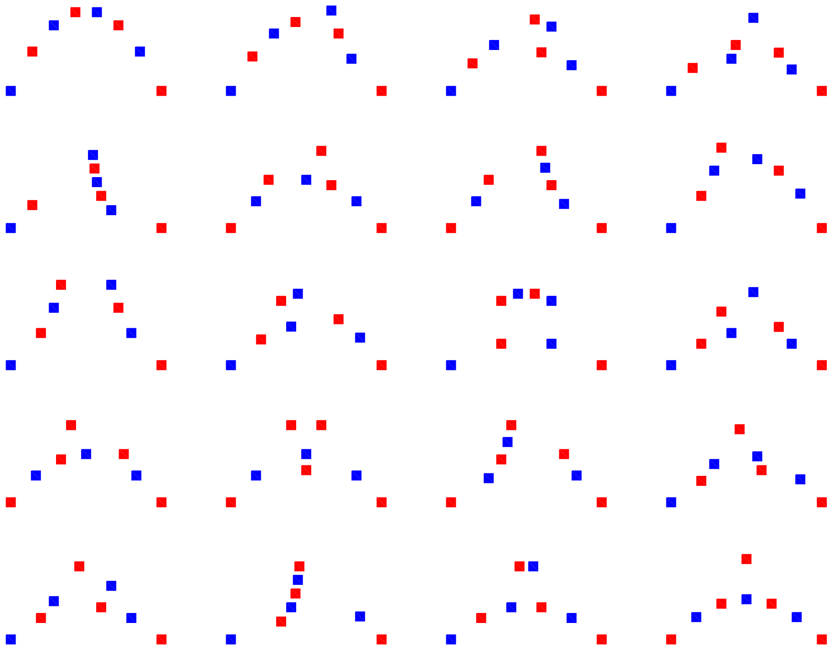

Figure 1: All neighbourly Gale Diagrams, some isomorphic.

## 3 Discussion

Since this is a studied problem (see [2]), we know that the 3 different simplicial neighbourly 4-polytopes on 8 vertices are $C_4(8)$, $N_8$ and $N_8^*$, where $C_4(8)$ is the cyclic polytope on 8 vertices, and an explicit description of $N_8$ and $N_8^*$ can be found in page 125 of [1].

## 4 Programming details

We start with a large number of Gale diagrams that we then filter through some tests to get to a low number of diagrams that we check for isomorphisms. Since the first computations take some time to run, we included an option in our code to use the hardcoded diagrams that we need to check for isomorphism.

The first thing the code does is find a finite number of points in the plane where our points of the different Gale diagrams will be. To do this, we use the fact that not the

```
Points of polytope 1
Point 0: 31/7 13/7 26/7 -10/7 1
Point 1: -45/91 8/91 -33/91 24/91 1
Point 2: -27/35 -54/35 -2/7 -3/7 1
Point 3: -1 0 0 0 1
Point 4: 0 -1 0 0 1
Point 5: 0 0 -1 0 1
Point 6: 0 0 0 -1 1
Point 7: 0 0 0 0 1


Points of polytope 2
Point 0: 101/27 38/27 86/27 -35/27 1
Point 1: -95/201 28/201 -71/201 56/201 1
Point 2: -131/165 -257/165 -10/33 -14/33 1
Point 3: -1 0 0 0 1
Point 4: 0 -1 0 0 1
Point 5: 0 0 -1 0 1
Point 6: 0 0 0 -1 1
Point 7: 0 0 0 0 1


Points of polytope 3
Point 0: 679/113 234/113 486/113 -189/113 1
Point 1: -959/1913 252/1913 -639/1913 504/1913 1
Point 2: -133/163 -167/163 -38/163 -42/163 1
Point 3: -1 0 0 0 1
Point 4: 0 -1 0 0 1
Point 5: 0 0 -1 0 1
Point 6: 0 0 0 -1 1
Point 7: 0 0 0 0 1
```

Figure 2: The coordinates for all 3 different neighbourly 4-polytopes on 8 vertices up to combinatorial equivalence

exact the location of the points is important, but the position of each point with respect to the lines defined by the other points. Therefore we can fix two of our points, 1 and 2, to be the lowermost, and choose the lines through these two points on which the other points will lie. To get more or less reasonable coordinates, we chose to fix the points $(0,0)$ and $(2,0)$, and the lines through them and $(1,i)$, with $i = 1,\dots,6$. Then, the points that we will consider to be on the Gale diagram are the intersections through the lines from 1 and from 2.

`enumerate_all_gale` uses backtracking to find all possible combinations of points on the grid of intersections previously defined. To do this, it takes into account that for a polytope to be simplicial, no three points can be aligned, so when it chooses a point to lie in a line through 1, it of course lies in a line through 2 and automatically bans any other point to go through the same line through 2. This procedure produces $6! = 720$ configurations of points in the plane. In addition, it has to give a sign to each point. The first sign is prescribed (since configurations with all signs interchanged are equivalent, we can prescribe one sign). Furthermore, only configurations with 3 or 4 negative signs are considered, since other configurations will not result in a neighbourly polytope. This gives for each of the Gale diagrams found $\binom{7}{5} + \binom{7}{4} = 91$ possible combinations of signs, so in total our program runs initially through $720 \cdot 91 = 65520$ possible Gale diagrams.

Since the computation of checking if a Gale diagram is neighbourly is a bit expensive, we try to reduce the number of Gale diagrams by eliminating the ones that are not simplicial. `isSimplicial` checks that in every set of four points of the diagram, no three points are aligned. After this step we are left with 51688 Gale diagrams.

`is_neighborly` loops through every pair of two points, and for each choice of pair, it removes the pair from the Gale diagram and then checks if the convex hulls of the negative and positive points intersect. After this step we are left with 20 Gale diagrams that can be seen in 1 .

`makeVertexFacetStructure` gives the vertex-facet lattice obtained from the Gale diagram, by creating a vertex $i$ for every point $i$ ($i = 0, \ldots, 7$) and making a vertex for every facet, so we have 28 vertices. The result is returned in an igraph_t object, from the igraph library[1].

To check whether two Gale diagrams are isomorphic, we use the in igraph built in graph isomorphism algorithm (uses BLISS). We find 3 different combinatorial types. Of the 20 Gale diagrams shown in 1, 7 represent one polytope, 8 another and 5 another.

`gale_to_polytope` takes an affine Gale diagram that we know corresponds to a simplicial and neighbourly polytope and returns projective coordinates for the vertices of the polytope. The procedure follows the reverse idea of how the vectors of the Gale transform are obtained. We initially have a matrix $B$ with the Gale points in the plane, and a sign assigned to each of the points. We project from the origin in $\mathbb{R}^3$ all positive points to the plane $Z = 1$, and all negative points to the plane $Z = -1$. Once we have this, we need the vector configuration to be balanced. Therefore we scale the vectors. We need coefficients $\lambda_i$ that multiply each vector $v_i$ and such that then all rows of the matrix $B$ sum to zero. To do this we can fix 5 of the $\lambda_i$ since any other set of three

[1] `http://igraph.org/c/`

points will have full rank because we are taking Gale diagrams where no three points are aligned, and then we only need to solve a $3x3$ linear system of equations. After the rescaling of the vectors of the Gale transform, we can get the kernel of their matrix and the vector of all ones is in its affine span, so we either move it or add it to the last row (which we can do by Steinitz Exchange Lemma) and we get 8 points of $\mathbb{R}^4$ in projective coordinates that are the vertices of a polytope to which the Gale diagram corresponds to.

# 5 Proof

There are some non-trivial results used to solve this problem that are worth discussing.

The theory of Gale diagrams and why they allow us to reconstruct polytopes can be found in [3] and [4]. We claim that only the position of a point in the diagram relative to the position of the lines defined by other points is important, as opposed to the exact coordinates of the point. That is quite clear: we select a point, and we can read from the diagram the faces of the polytope. This is done by looking at intersections of the convex hulls of negative and positive sets of points, and in a set where our selected point participates, it is clear that if the intersection is empty this does not change unless our point crosses a line of opposite sign, and if it is not empty this cannot change unless it crosses a line of opposite sign.

To check whether two polytopes are combinatorially equivalent, we check the isomorphism of the vertex-facet lattice, instead of the face lattice. The vertex-facet lattice is enough to determine the combinatorial type of a polytope because the face lattice is graded, atomic and coatomic ([3]).

To make sure our program doesn't make any rounding errors, we used rational numbers from the GMP-library.

Finally, our construction of a polytope from a Gale diagram is based on linear algebra. It is simply reversing the construction of a Gale diagram.

# References

[1] B. Grünbaum, Convex polytopes. *Graduate Texts in Mathematics* vol 221, (2003).

[2] B. Grünbaum, V.P. Sreedharan, An enumeration of simplicial 4-polytopes with 8 vertices. *Journal of Combinatiorial Theory 2*, 437-465, (1967).

[3] J. Matoušek, Lectures on Discrete Geometry. *Graduate Texts in Mathematics* vol 212, (2002).

[4] G. Ziegler, Lectures on Polytopes. *Graduate Texts in Mathematics* vol 212, (1993).