MOPA - Master EUR - Sorbonne Paris Nord university

Practical session: Characterization of Combinatorial Polyedra with the PORTA Software

1 Characterization of Polyhedra

1.1 Polytopes and Polyhedra

In this document, we consider polyhedra embedded in $I\!R^n$ for a known n.

• A polyhedron P in $I\!R^n$ is of **dimension** d if there are at most $d+1$ affinely independent points in P.
We then write $dim(P) = d$.
A polyhedron P is said to be **full-dimensional** if $dim(P) = n$.

• Given $ax \leq \alpha$ a valid inequality for P.
The subset $F = \{x \in P \mid ax = \alpha\}$ is called a **face** of P.

• A face F is a **facet** of P if $\emptyset \neq F \neq P$ and $dim(F) = dim(P) - 1$.

• A full-dimensional polytope P can be **completely described** by:

- either the set of all its extreme points (of which it is the convex hull)
- or by a system containing at least one inequality for each of its facets.

A system describing a polytope can be **minimal** if it contains exactly one inequality per facet. Thus, a non-minimal system will contain valid inequalities that do not define facets, or several inequalities describing the same facet.

• More generally, a polytope P, which is not full-dimensional, can be described by:

- either the set of all its extreme points
- or by a system containing at least $n - dim(P)$ equalities and at least one inequality for each of its facets.

A minimal system will then contain exactly $n - dim(P)$ linearly independent equalities and exactly one inequality per facet.

- Even more generally, a polyhedron P can be unbounded and P is then described by:
  - either the set of all its extreme points plus a set of vectors R (called extreme rays). If we denote Q as the convex hull of the extreme points, then P is the convex cone obtained by intersecting all the cones with rays R emanating from the points of Q.
  - or by a system of equalities and inequalities.

Of course, there are unbounded polyhedra of various dimensions.

1.2 Conversion

The **"conversion"** from a description by points (or points and rays) to a description by a system (and vice versa) is a difficult problem. Note that the output of such an algorithm is often exponential...

The algebraic principle basically relies on
- from inequalities to points: computing intersections
- from points to inequalities: using the definition of the convex hull to obtain inequalities or equalities.
In both cases, the description involves a higher-dimensional space, then a Fourier-Motzkin projection is applied.

From dimension 10 or 12, it often takes several weeks of computation!

---

2 The PORTA Software

The name PORTA is an abbreviation for *POlyhedron Representation Transformation Algorithm*. It is a collection of programs for analyzing polyhedra.

This software was written by Thomas Christof. Subsequently, it was maintained by Andreas LÃ¶bel and then by Sebastian Schenker. The not-so-official but only website is `http://porta.zib.de` where the software is cited under the GNU General Public License.

Its installation is simple under Linux: just extract the archive, go to the gnu-make directory, and type the command "make".
The different programs are then compiled in the "gnu-make/bin" directory.

At PPTI, Porta has already been compiled in the directory "/Vrac/porta-1.4.1"... but for heavy use, it is recommended to install it locally on a machine.
To easily use the programs at PPTI, add in the bashrc (or bash_profile depending on the case) file:

```
export PATH=$PATH:/Vrac/porta-1.4.1/gnu-make/bin:.
```

Complete information is in the README files and especially INFO. The remainder of this document focuses on some uses of Porta in the context of the MAOA course.

2.1 Formats and Programs

Porta contains several programs including

dim returns the dimension of a polyhedron

traf converts from a point (or point-ray) description to a minimal system of equalities and inequalities... and vice versa!

vint enumerates all integer points that are solutions of a system of inequalities

Porta contains other programs that are not discussed here.

Everything is based on two formats:
- one with extension .ieq describing inequalities
- the other with extension .poi describing points
which are described in the exercise section.

For **traf**:
The command line `traf file.ieq` produces a file `file.ieq.poi` which is the result of transforming the inequalities into points.
And conversely, the command line `traf file.poi` produces a file `file.poi.ieq` which is the result of transforming the points into inequalities.

For **vint**:
The command line `vint file.ieq` produces a file `file.poi` which lists all integer points contained in the system.
Attention: vint requires LOWER_BOUND and UPPER_BOUND fields specific to this program (see later).

---

3 Quick Exercise on the Stable Set Polytope

3.1 Cycle on 5 Vertices

We consider the graph C_5 of a cycle on 5 vertices, numbered 1,. . . , 5.

Question 1 *Enumerate all trivial and edge inequalities of the ILP formulation of the stable set problem on* C_5.

In the PORTA "ieq" format, we get this

```
DIM = 5

VALID
0 0 0 0 0

INEQUALITIES_SECTION
x1+x2<=1
x2+x3<=1
x3+x4<=1
x4+x5<=1
x5+x1<=1
x1>=0
x2>=0
x3>=0
x4>=0
x5>=0
x1<=1
x2<=1
x3<=1
x4<=1
x5<=1
END
```

where

- the variables x_i correspond to including or not a vertex in a stable set
- DIM denotes the number of variables. **Attention:** this is not the dimension.
- VALID is a feasible point: here the zero vector which is a stable set
- and a section for edge and trivial inequalities.

Remark: variables in porta are always x_1, x_2,... which is great for the stable set polytope but really not very convenient for polyhedra where the numbering of objects (for example edges) is completely different.

Question 2 *Write the previous text in a file named* `cycle5.ieq`*. Use* `traf` *to obtain the corresponding vertex description.*

Normally you will get the file `cycle5.ieq.poi`:

```
DIM = 5

CONV_SECTION
(  1)   0   0   0   0   0
(  2) 1/2 1/2 1/2 1/2 1/2
(  3)   0   0   0   0   1
(  4)   0   0   0   1   0
(  5)   0   0   1   0   0
(  6)   0   1   0   0   0
(  7)   1   0   0   0   0
(  8)   0   0   1   0   1
(  9)   0   1   0   0   1
( 10)   0   1   0   1   0
( 11)   1   0   0   1   0
( 12)   1   0   1   0   0
END
```

These are all the extreme points of the linear formulation described in `cycle5.ieq`. Note that this formulation is not integral because it has a fractional extreme point!
We now want to know a minimal system of inequalities describing the stable set polytope associated with C_5. For this, we will proceed in 2 steps:
- first, we will enumerate all incidence vectors of a stable set of C_5
- then use **traf** to obtain the hull of these points.

Question 3 *Obtain all incidence vectors of stable sets of* C_5*.*

We can enumerate them by hand but we can also use `vint`: for this, we add to the file `cycle5.ieq` at the end

```
LOWER_BOUNDS
0 0 0 0 0

UPPER_BOUNDS
1 1 1 1 1
```

which specify the integer bounds of the variables
Normally, you should get the file `cycle5.poi`

```
DIM =  5

CONV_SECTION
(  1) 0 0 0 0 0
(  2) 0 0 0 0 1
(  3) 0 0 0 1 0
(  4) 0 0 1 0 0
(  5) 0 0 1 0 1
(  6) 0 1 0 0 0
(  7) 0 1 0 0 1
(  8) 0 1 0 1 0
(  9) 1 0 0 0 0
( 10) 1 0 0 1 0
( 11) 1 0 1 0 0

END
```

Question 4 *Using* `traf cycle5.poi`*, then obtain* `cycle5.poi.ieq`*, that is, the stable set polytope for* C_5*!*

Question 5 *What do you notice about this minimal description* `cycle5.poi.ieq`*?*

Question 6 *Without using Porta, what will* `cycle5.poi.ieq.poi` *be?*

3.2 Wheel on 5 Vertices

Question 7 *Repeat the same process for the wheel* W_5*, i.e., a cycle* C_5 *plus a vertex numbered 6 that is connected to all vertices of the cycle.*

Question 8 *Interpret each of the inequalities of the stable set polytope for* W_5*.*

3.3 Prism

Same process on this graph (which comes from the graph called "prism").