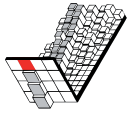


rPorta
An R Package for Analyzing Polytopes and Polyhedra

Robin Nunkesser[1,3] Silke Straatmann[2,3] Simone Wenzel[2]

[1]Department of Computer Science, TU Dortmund

[2]Department of Statistics, TU Dortmund

[3]Collaborative Research Center 475

Outline

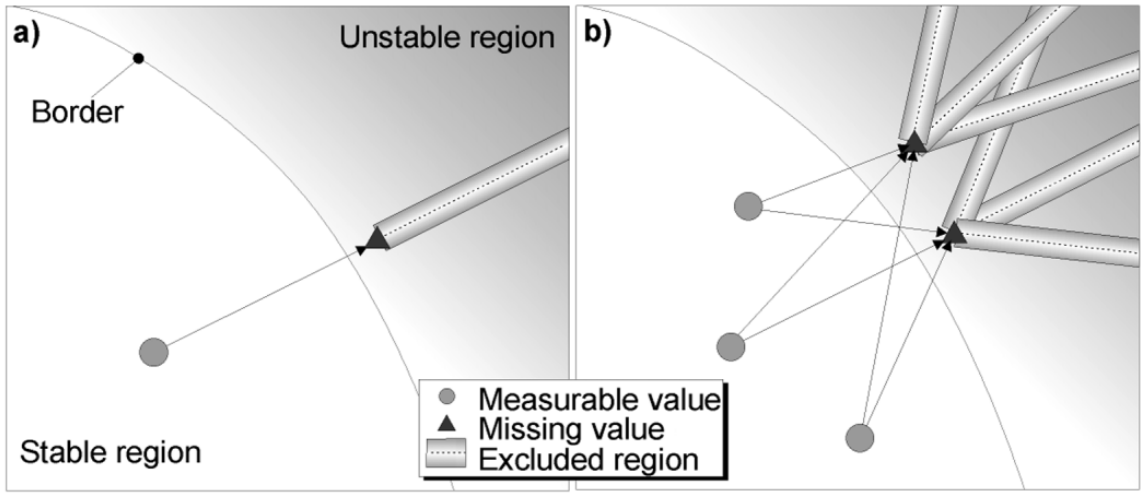
Motivation

Problem from design of experiments

Generate a space-filling design exploring the unknown feasible parameter space with a minimum of failures/missing values

Strategy (in the spirit of Henkenjohann et al., 2005)

- Assume feasible area is connected and convex
- Viewed from feasible point space behind failure points is failure region
- Examine and restrict parameter space sequentially



Motivation

Key aspects required for the strategy in $\mathbb{R}^d$

- Inefficient to construct a convex cone for each combination of one failure and d feasible points
- Find a fast way to check if a candidate-point is lying inside one of these cones and hence is a failure point

Solution

- Use Polyhedral Convex Cones (PCCs) with *extreme rays* to minimize number of convex cones
- Calculate PCCs with Double Description Method as introduced in Fukuda and Prodon (1996)

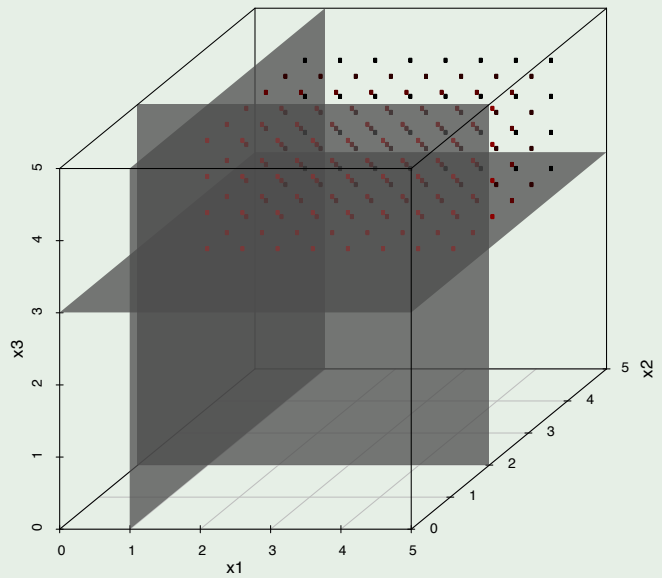
Double Description Method

Double Description Pair

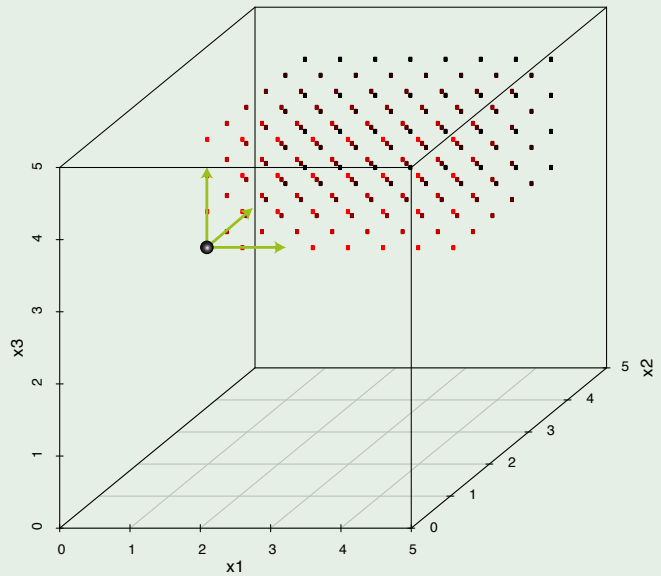
A pair (A, R) of real matrices A and R is called a *double description pair* (DD pair) if the following relationship holds:

$$Ax \geq 0 \text{ if and only if } x = R\lambda \text{ for some } \lambda \geq 0 \ .$$

$x_1 \geq 1$, $x_2 \geq 2$, and $x_3 \geq 3$



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



Double Description Method

Polyhedral Cone

A subset $P \in \mathbb{R}^d$ is called a *polyhedral cone* if

$$\exists\, A \in \mathbb{R}^{n \times d} : P = \{x \in \mathbb{R}^d : Ax \geq 0\} =: P(A)$$

Representation and Generation

Let $P \in \mathbb{R}^d$ be a polyhedral cone and $A \in \mathbb{R}^{n \times d}$ be the matrix with $P = P(A)$. Then there exists a matrix $R \in \mathbb{R}^{d \times m}$ such that (A, R) is a DD pair and it is:

$$\begin{aligned} P &= \{x \in \mathbb{R}^d : Ax \geq 0\} \\ &= \{x \in \mathbb{R}^d : x = R\lambda \text{ for some } \lambda \geq 0\} \ . \end{aligned}$$

A is called *representation matrix* of the polyhedral cone P,
R is called *generating matrix* for the polyhedral cone.

R Package rPorta

R Interface to PORTA (Polyhedron Representation Transformation Algorithm) by T. Christof (Universität Heidelberg) and A. Löbel (ZIB)

What is PORTA?

- Collection of routines for analyzing polytopes and polyhedra
- Supports both representations of the double description pair
- Transforms between the representations

Why PORTA? (and not polymake, cdd, PPL,...)

- Platform independence (gcc)
- Free availability (GPL license)
- Speed
- Fitting functionality for the intended application

Comparison to rcdd

rcdd is an R Package interfacing cdd(lib) (C implementation of the double description method) by K. Fukuda (Swiss Federal Institute of Technology)

What is cdd?

- Supports both representations of the double description pair
- Transforms between the representations
- Additionaly solves linear programming problems

Short comparison

Point of comparison	rPorta	rcdd
Platforms	Every platform with R	Every platform with gmp
Arithmetic	64 bit rational arithmetic	Exact rational arithmetic
Functions	Collection for transforming and analyzing polyhedra	Focuses on transformation and linear programming

PORTA's UI and its R Counterpart

PORTA reads all data from and to files $\leftrightarrow$ rPorta wraps files into S4 classes

Example of an ieq file ($\hat{=}$ representation matrix A)

```
DIM = 3

INEQUALITIES_SECTION
(1) x1              >= 1
(2)       x2        >= 2
(3)             x3  >= 3

END
```

S4 object `ieqExample` ($\hat{=}$ representation matrix A)

```
> ieqExample@inequalities@num
     [,1] [,2] [,3] [,4]
[1,]    1    0    0    1
[2,]    0    1    0    2
[3,]    0    0    1    3

>ieqExample@inequalities@sign
[1] 1 1 1
```

PORTA's UI and its R Counterpart

PORTA reads all data from and to files $\leftrightarrow$ rPorta wraps files into S4 classes

Example of a poi file ($\hat{=}$ generating matrix R)

```
DIM = 3

CONV_SECTION
 1 2 3

CONE_SECTION
 0 0 1
 0 1 0
 1 0 0

END
```

S4 object poiExample ($\hat{=}$ generating matrix R)

```
> poiExample=traf(ieqExample)

> poiExample@convex_hull@num
     [,1] [,2] [,3]
[1,]    1    2    3

> poiExample@convex_cone@num
     [,1] [,2] [,3]
[1,]    0    0    1
[2,]    0    1    0
[3,]    1    0    0
```

Method traf

Method to transform between the double description pair representations

S4 method

```
traf(object, opt_elim=FALSE, chernikov_rule_off=FALSE,
validity_table_out=FALSE, long_arithmetic=FALSE)
```

`object`	Object of class `ieqFile` or `poiFile`
`opt_elim`	Use a heuristic to eliminate that variable next, for which the number of new inequalities is minimal
`chernikov_rule_off`	Fourier-Motzkin elimination without or with rule of Chernikov
`validity_table_out`	Include a table which indicates strong validity
`long_arithmetic`	Use long integers for intermediate results

Example for traf

```
> poiExample=traf(ieqExample)
```

Method `fctp`

Checks the facet inducing property

S4 method

```
fctp(object, poiObject)
```

`object, poiObject` `ieqFile` object and `poiFile` object to check

Example `ieqFile`

```
DIM = 3
VALID
2 0 0
INEQUALITIES_SECTION
(1) x1 + x2 + x3 >= 2
(2) x1 + x2 + x3 <= 2
(3) x1           >= 0
(4)      x2      >= 0
(5)           x3 >= 0
```

Result for $(0,1,0)$, $(0,0,2)$, and $(0,0,3)$

```
[[1]] # not valid for (1)
0 1 0
[[2]] # satisfying (1) with equality
0 0 2
[[3]] # not valid for (2)
0 0 3
[[4]] # satisfying (2) with equality
0 0 2
...
```

Some Other Functions

Helper functions

`as.poi, as.ieq`	turns objects into `poi` or `ieq` objects
`read.portaFile`	converts PORTA files to corresponding S4 classes

PORTA functions

`vint`	enumerates integral points of a linear system
`portsort`	sorts and formats `poiFile` and `ieqFile` objects
`fmel`	projects a linear system to a subspace
`iespo`	enumerates valid inequalities for a given polyhedron
`posie`	enumerates valid points for given inequalities

Application specific functions

`failureRegions`	function specific for the application example

Application of rPorta

`failureRegions` determines unfeasible regions inside a parameter space (here: 3 steps with 10 points each to restrict parameter space $[-2, 2]^2$)

S4 method

```
failureRegions(experiments, parameterspace, fail)
```

`parameterspace` Represents parameter space grid (here: 1681 points)

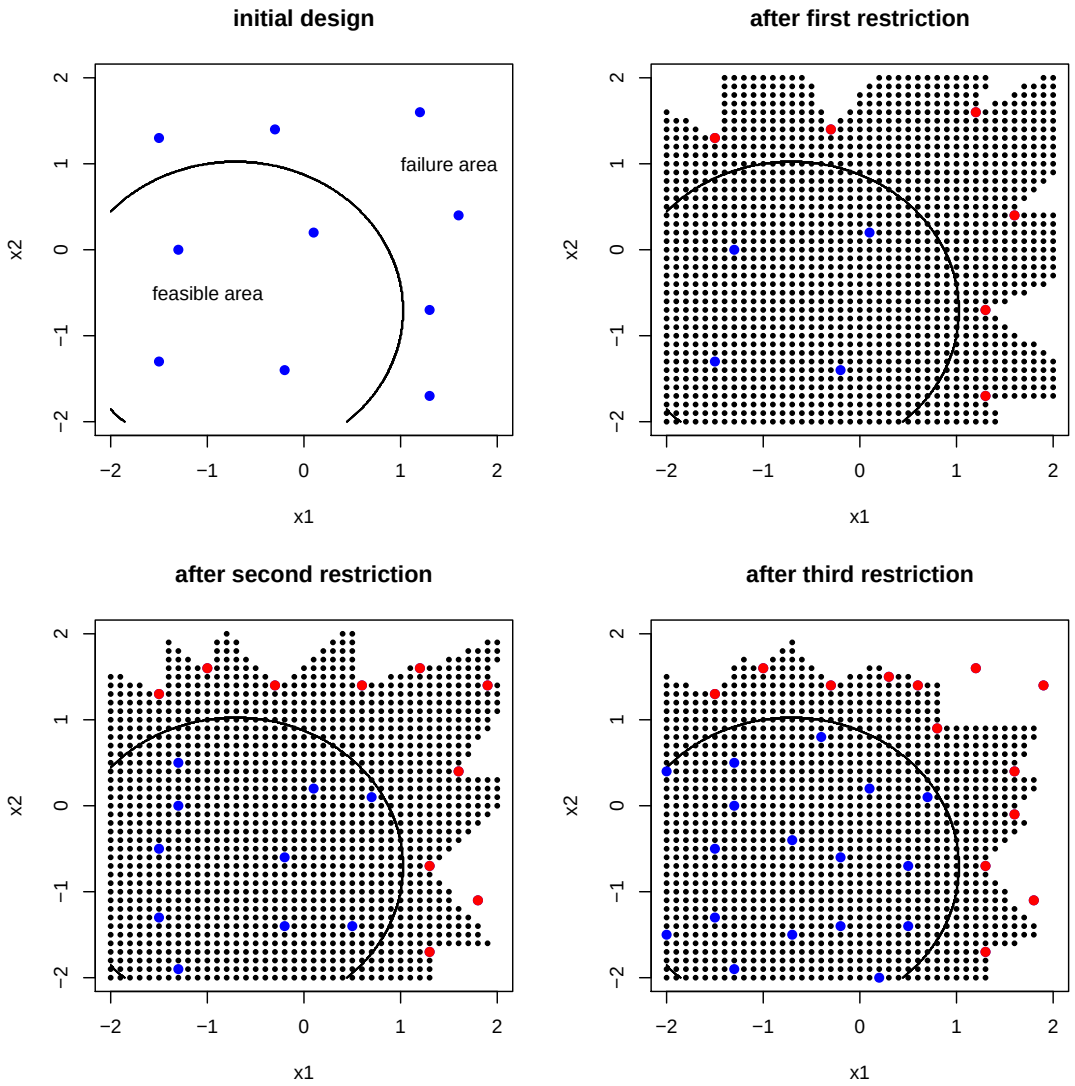
`experiments` Contains the points with known results (here: initial 10 point uniform coverage design)

`fail` A logical vector indicating which `experiments` failed

```
res <- failureRegions(as.poi(exper),as.poi(paramspace),fails)
restrictedSpace <- as.matrix(getFeasiblePoints(res))
```

- update with 10 new points from `restrictedSpace` regarding space-filling criterias
- restrict `restrictedSpace` again (repeat until 3 restrictions)

Result



rPorta

Each step < 1 second

Old Method

Step 1: 16.6 seconds
Step 2: 194.17 seconds
Step 3: 744.01 seconds

Summary

- Double Description Method speeds up handling of convex cones
- rPorta provides an interface to a double description implementation
- Easy analysis of polytopes and polyhedra in R

Bibliography


Fukuda, K., Prodon, A., 1996. Double description method revisited. In: Combinatorics and Computer Science. Vol. 1120 of LNCS. Springer-Verlag, London, pp. 91–111.


Geyer, C. J., Meeden, G. D., 2008. rcdd: rcdd (C Double Description for R). R package version 1.1.


Henkenjohann, N., Göbel, R., Kleiner, M., Kunert, J., 2005. An adaptive sequential procedure for efficient optimization of the sheet metal spinning process. Quality and Reliability Engineering International 21 (5), 439–455.


Nunkesser, R., Straatmann, S., Wenzel, S., 2008. rPorta: R/PORTA interface. R package version 0.1-6.