

Polyhedral Computation

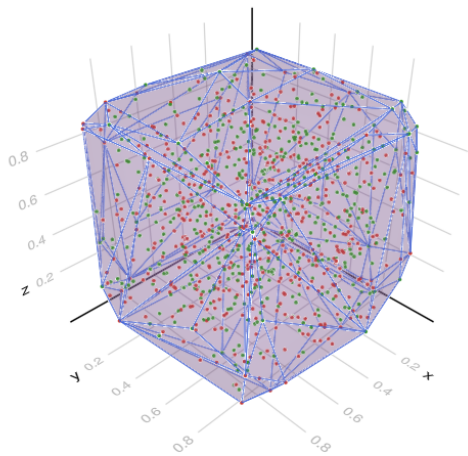
Benoît Legat

July 26th, 2023

ERC “Back to the Roots” with Prof. Bart De Moor, STADIUS, KU Leuven



What is Polyhedral Computation



What is it used for ?

Reachability analysis

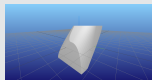


Dionysos

JuliaReach

Invariant sets

InvariantSets.jl



SetProg.jl, SwitchOnSafety.jl



EntropicCone.jl

Delaunay tessellation

HyperVoronoiDelaunay.jl,
VoronoiGraph.jl,
LaguerreVoronoi.jl, VoroX.jl

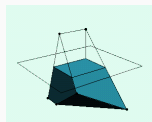
Game theory

GameTheory.jl

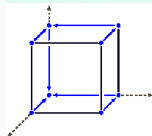
Optimization



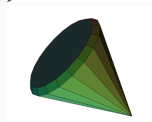
Which libraries exists



cddlib

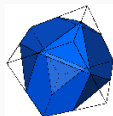


lrslib



qhull

pd



PORTA

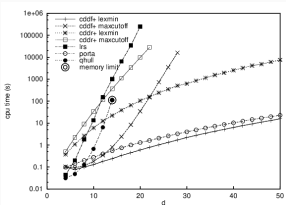


Parma Polyhedra
Library (PPL)

ConvexHull.jl

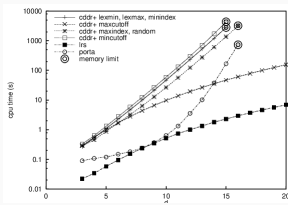
Polyhedra.jl

Fastest algorithm depend on the class of polyhedra



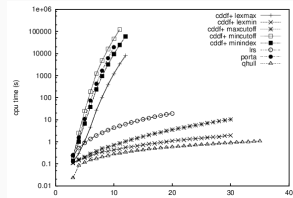
Double description
cddlib, porta

Rational{BigInt} cddr **faster** than
Float64 qhull



Reverse search
lrslib

Rational{BigInt} lrs **faster** than
Float64 qhull



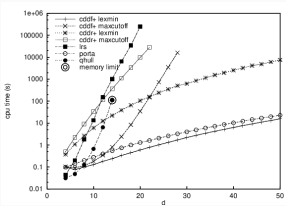
Quick hull
qhull

Rational{BigInt} lrs **faster** than
Rational{Int} PORTA

Avis D, Bremner D. (1995) *How good are convex hull algorithms?*

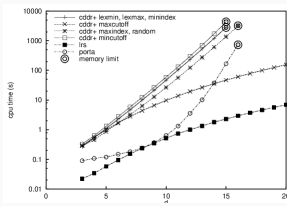
Proceedings of the eleventh annual symposium on Computational geometry

Fastest algorithm depend on the class of polyhedra



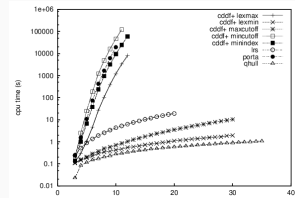
Double description
cddlib, porta

`Rational{BigInt} cddr` faster than
`Float64 qhull`



Reverse search
lrslib

`Rational{BigInt} lrs` faster than
`Float64 qhull`



Quick hull
qhull

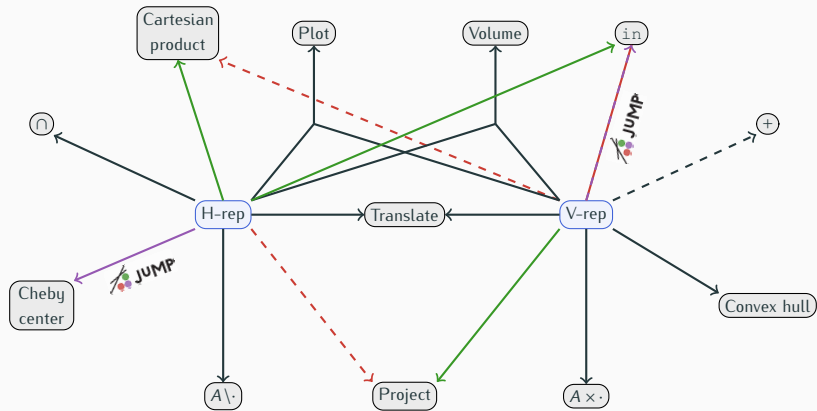
`Rational{BigInt} lrs` faster than
`Rational{Int} PORTA`

Avis D, Bremner D. (1995) *How good are convex hull algorithms?*

Proceedings of the eleventh annual symposium on Computational geometry

`InvariantSets.jl` built on `Polyhedra.jl` but alternatives have fixed library

Just In Time representation conversion





→ Preferred

→ Fallback if other rep not computed

→ Solve linear program using JUMP

- - -> Size of resulting representation may grow significantly

Polyhedra.jl interface

- V-representation type
- H-representation type (can use  JUMP model)
- Polyhedron type
 - Detect redundancy and linearity (e.g. affine hull) with  JUMP
 - **Transparently** H-convert or V-convert **only** if needed
- Interface implementation for libraries:
 - Defaults to pure julia double description implementation
 - CDDLib.jl (cddlib) : Thanks David van Leeuwen!
 - ConvexHull.jl by Joey Huchette
 - LRSLib.jl (lrslib) : Thanks Daisuke Oyama!
 - QHull.jl (qhull)
 - ParmaPolyhedra.jl (PPL)
 - XPORTA.jl (PORTA) : Thanks Brian Doolittle!