

polymake for integer linear programming
ISMP 2012

Michael Joswig
w/ Ewgenij Gawrilow and many others

TU Darmstadt

Berlin, August 21, 2012

- software for research (and education) in:
 - geometric combinatorics: convex polytopes
 - **linear/combinatorial optimization**
 - algebraic geometry
 - ...
- open source, GNU Public License
 - supported platforms: Linux, FreeBSD, MacOS X
 - more than 100,000 uloc (**Perl**, **C++**, C, Java)
- co-authored (since 1996) w/ Ewgenij Gawrilow [now TomTom]
 - contributions by many people

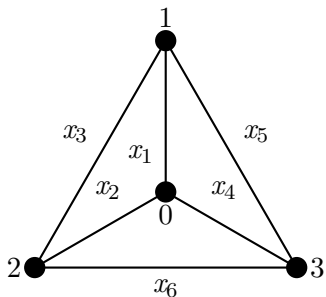
Example: Maximal Matching

Let $G = (V, E)$ be a finite graph.

$$\max \sum_{e \in E} x_e$$

$$\text{s.t.} \quad \sum_{e \ni v} x_e \leq 1 \quad \text{for all } v \in V$$

$$x_e \in \{0, 1\} \quad \text{for all } e \in E$$



switch to demo

Algorithm Overview (Selection)

- convex polytopes, polyhedra and fans
 - convex hulls: cdd, lrs, beneath-and-beyond
 - Voronoi diagrams, Delone decompositions
 - face lattices: Kaibel–Pfetsch (including variations)
 - \rightsquigarrow lattice polytopes/toric varieties
- optimization
 - Hilbert bases: normaliz, 4ti2
 - Gomory–Chvátal closures
 - counting integer points: LattE, bounding box/by projection
- graphs, matroids, ...
- simplicial complexes
- tropical geometry

Other polymake Resources for Optimization

- <http://polymake.org/tutorial/...>
 - Marc Pfetsch & Sebastian Pokutta: optimization tutorial
 - Michael Schmitt: implementation of branch-and-bound as proof-of-concept
- Matthias Walter: total unimodularity test [Wed 10:30, H 3005]
<http://github.com/xammy/unimodularity-test/wiki/Polymake-Extension>

- Hybrid design: Perl (interpreted) and C++ (compiled)
 - Perl: Server side (= organization/communication)
 - C++: Client side (= computation)
- Shell type user interface
 - (extension of) Perl as language
- Technical features include:
 - C++ template library
 - extends STL, based on template meta-programming
 - shared memory communication between client/server, transaction safe
 - whole system can be used as a C++ library (since 2.12)
- prototype: pypolymake [Burcin Erocal]
- interfaces to polymake in the making:
 - Singular, GAP, Sage

Objects and Properties

- hierarchy of **big object types** (modelling mathematical concepts)
 - e.g., polytopes, simplicial complexes, graphs, ...
 - under control of client/server system
 - with templates
- **properties** as class members (functions or data)
 - strongly typed
 - a type is a built-in Perl type, a C++ class type, or a big object type
 - immutable
- new big object types and properties to a given big object type *can be added at will*
- big object types grouped into **applications** (\approx name spaces)

New Features of `polymake` 2.13

- quadratic field extensions and exact representations of Platonic solids (and others)
- regularity for complete fans
- commutative algebra: new application ideal
- visualization: improved Sketch output
- fast loading of gzipped XML files
- further modularization via bundled extensions