

IMA2017_LP_demo

August 24, 2017

1 Mixed Integer Linear Programming demo (IMA 2017)

We show how Sage can be used to do linear optimization with different kind of real numbers:
- floating point numbers (GLPK, CBC/Coin-OR, ...) - rational (PPL) - algebraic numbers (Sage generic implementation)

1.1 Example 1

Let us consider the following problem

maximize $f(x_0, x_1, x_2) = 3x_0 + 2x_1 + x_2$ under the constraints - $x_0 \geq 0, x_1 \geq 0, x_2 \geq 0$ -
 $7x_0 \leq x_1 + x_2 - 5x_1 \leq 3x_2 - 9x_0 + 8x_1 + 7x_2 = 27$

```
In [ ]: MixedIntegerLinearProgram?
```

```
In [ ]: M = MixedIntegerLinearProgram()
```

```
In [ ]: x = M.new_variable(nonnegative=True)
```

```
In [ ]: M.add_constraint(7*x[0] <= x[1] + x[2])
M.add_constraint(5*x[1] <= 3*x[2])
M.add_constraint(9*x[0] + 8 *x[1] + 7*x[2] == 27)
```

```
In [ ]: M.set_objective(3*x[0] + 2*x[1] + x[2])
```

```
In [ ]: M.solve()
```

```
In [ ]: M.get_values(x)
```

```
In [ ]:
```

```
In [ ]: M.get_backend()
```

```
In [ ]:
```

```
In [ ]: # to solve over rationals, just rerun the example by setting
#
# M = MixedIntegerLinearProgram(solver='PPL')
#
#####
```

```

#
# or to use CBC
# needs to install cbc first with
#
#   $ sage -i cbc
#   $ sage -b
#
# and then
#
# M = MixedIntegerLinearProgram(solver='CBC')

```

1.2 Solving over $\mathbb{Q}[\sqrt{5}]$

We consider the regular dodecahedron and maximize the linear functional $\sqrt{5}x_0 + x_1 + x_2$ on it.
(why is it not in the documentation!?)

```

In [ ]: poly = polytopes.dodecahedron()
In [ ]: poly
In [ ]: lp, x = poly.to_linear_program(solver='InteractiveLP', return_variable=True)
In [ ]: lp
In [ ]: lp.show()
In [ ]:
In [ ]: K = poly.base_ring()
        K
In [ ]: sqrt5 = K.gen()
In [ ]: sqrt5**2
In [ ]: lp.set_objective(sqrt5 * x[0] + x[1] + x[2])
In [ ]: lp.solve()
In [ ]:
In [ ]: vals = lp.get_values(x)
        opt = vector((vals[0], vals[1], vals[2]))
        obj = vector((sqrt5, 1, 1))
In [ ]: vals
In [ ]: print opt
In [ ]: poly.plot(alpha=0.3) + \
        point3d([opt], color='red', pointsize=50) + \
        arrow(opt, opt+obj/2, color='yellow')
In [ ]:
In [ ]:

```