Nicolas Primeau
6352145
SYS5130
Project 1 Report

Summary

The following is a brief report on the work done for the project. It’s composed of a description of the 2 major components of the software including their limitations, the backend and the GUI, how to run it, and the validation question proposed in the project description solved both manually and with the software.

Components

The implementation is mostly made up of two components. The first is a backend that runs the simplex algorithm, and the other is the GUI.

The backend is independently and can be used as a library. It is used to solve optimization problem composed of variables, objectives, and constraints. The input can be given in any form. At the moment, it only supports the Simplex linear programming algorithm. When providing a system, all variables must be made positive, either manually or by marking them as negated. The software will take care of transforming it into standard form, then canonical form with slack variables. It will then iterate per the simplex algorithm until either an optimal solution is found, or loops with distance of less than 3 are detected. It returns the optimal values of the variables.

The GUI is web-based. It allows for the input of variables, objectives, and constraints. It accepts the input of maximization or minimization objective functions, as well as the input of expressions composed of simple terms. Variables can also be inverted in order to fulfill the conditions of the simplex algorithm. Once the system is suitable, you can choose the solver method (only simplex for now) and run it to solve the system. It also allows to reset the system so that another may be specified.

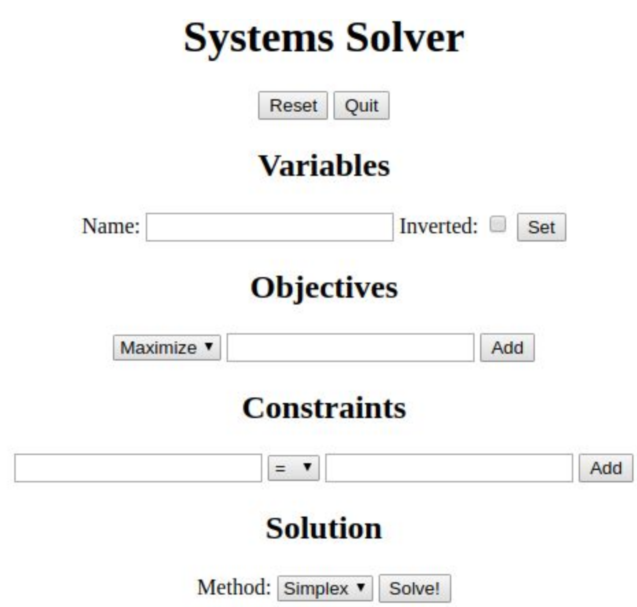

Figure 1 Screenshot of blank GUI

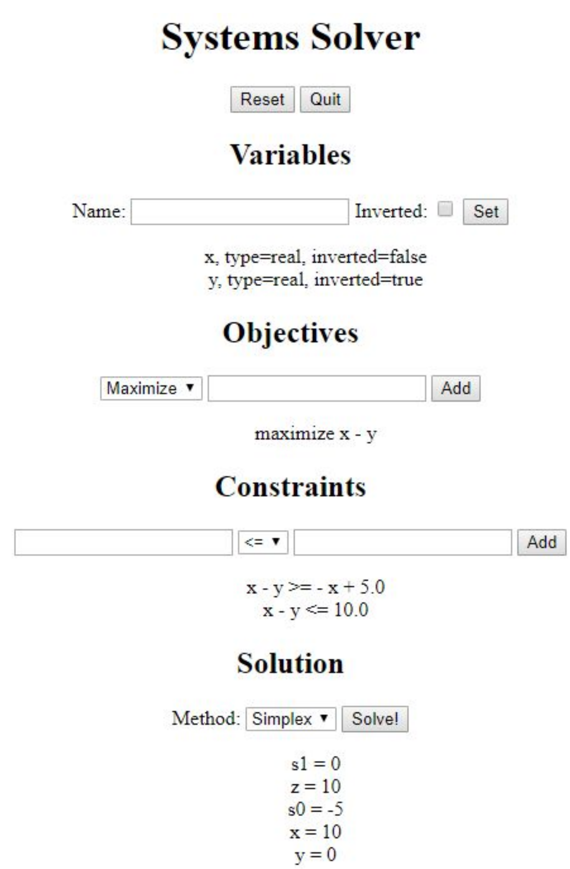

Figure 2 Screenshot of non standard solved system

Execution

The program can be executed via an executable that is provided under bin/. Command line or file explorer is fine. There is one for linux, bin/app.linux and one for windows, bin/app.exe It will launch a server that listens on port 50000, which can be connected to with the browser of your choice by navigating to http://localhost:50000/. Upon termination, click on quit and close the tab.

Validation

The validation question is the following system:

Maximize $z=x-1.2y$
$2x-y \le k$
$x-3y \le 120$
$x \ge 0, y \le 0$
$k=6 + 3 + 5 + 2 + 1+ 4 + 5 = 26$

First we need to change to standard form:
Maximize $z=x + 1.2\hat{y}$
$2x + \hat{y} \le 26$
$x + 3\hat{y} \le 120$
$x \ge 0, \hat{y} \ge 0$
$\hat{y} = -y$

and then the canonical form:

Maximize $z - x - 1.2\hat{y} = 0$
$2x + \hat{y} + s0 = 26$
$x + 3\hat{y} + s1 = 120$
$x \ge 0, \hat{y} \ge 0$
$\hat{y} = -y$

Solving by hand leads to this table:

x	ŷ	s0	s1	z	
2	1	1	0	0	26
-5	0	-3	1	0	42
7	0	6	0	5	156

Which means y = -26, s2=42, z=31.2, x=0, s1=0

Building this system in the code as a test case (available to run on-demand in test/test_simplex.py:test_simplex_project1), we get the following solution:

x=0, s0=0, y=-26.0, z=31.2, s1=42.0

Similar results are found when using the GUI:

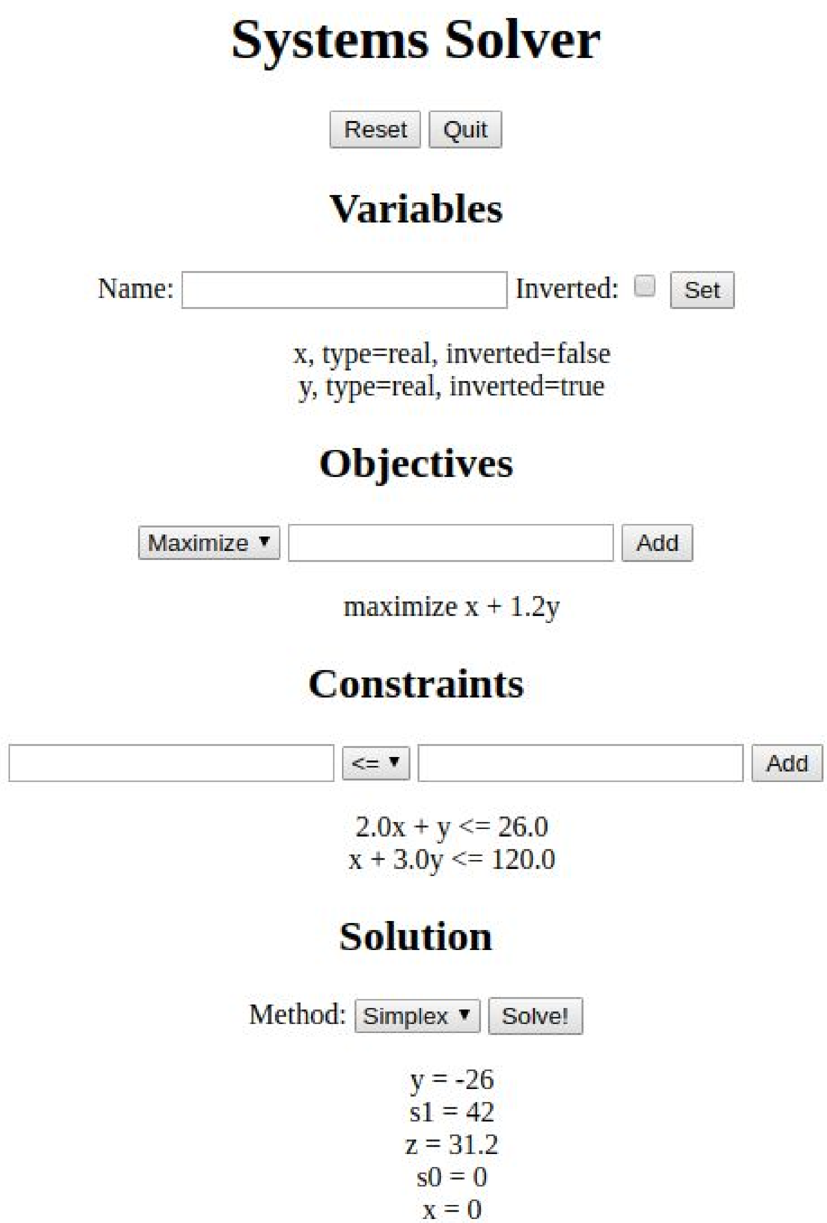

Figure 3 Solved Example System