Solving LP problems using Analog Circuitry

Jack Justus | jackjustushome@gmail.com

St. Margaret's Episcopal School, Class of 2024

Ming Hsieh Department of Electrical and Computer Engineering, SHINE 2023

USC Viterbi School of Engineering

SHINE Summer High School Intensive in Next-Generation Engineering

Introduction

Digital components have reached their atomic size limit, prompting recent computing improvements by increasing component density on chips rather than further miniaturization. Professor Wu's team proposes an alternate solution by researching analog electronics to enhance digital performance. Analog computers can be utilized to speed up computations of certain problems. One such example are LP problems.

Linear Programming (LP) Problems consist of a set of equalities and inequalities called constraints, and a function called the objective. Solving the problem requires to you to find the minimum value possible for the objective function, providing that this solution is also a solution to all the constraint functions. The objective for my LP Problem was x+y, meaning that the point with the smallest x+y value that satisfied the constraints was the solution to my problem.

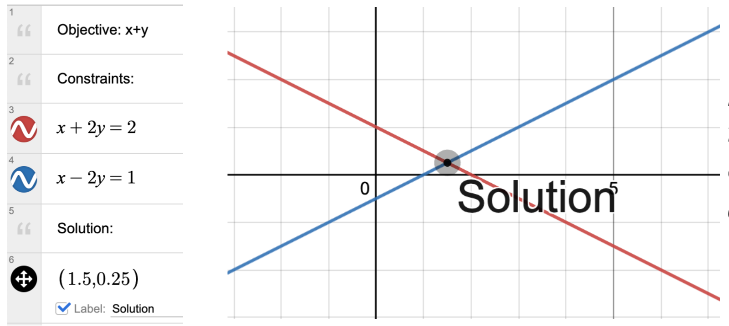

Figure 0: A graph showing the LP Problem that the circuit solves, with a solution of (1.5, 0.25)

It is possible to construct an analog circuit in order to compute this problem using various electrical components. I have constructed this circuit and compared it to the digital time to compute the same problem.

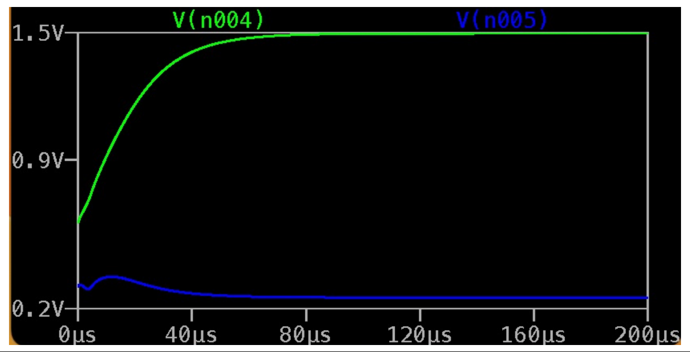

Figure: A graph showing the voltage output of a LP solver circuit with the solution (1.5, 0.25)

Objective & Impact of Professor's Research

Collision avoidance is an application where analog components are significantly faster than digital. Professor Wu's group is currently working on a drone which can calculate a route around an obstacle using an analog circuit.

This application significantly speeds up the pathfinding process, and in the future this technology may be applied to self-driving cars and other applications.

Acknowledgements

Thank you to Sushmit Hossain for helping me learn everything about analog circuitry, from Arduinos to memristors. I would like to thank Professor Wu for this incredible introduction to research. Additionally, thank you Marcus from the SHINE staff for being a wonderful center mentor.

Research & Learning Process

We started the research process by creating a list of tasks that would eventually lead up to building the complete LP solver circuit.

- **Learning how to build a circuit to solve an LP Problem**

Using Vichik's *Solving linear and quadratic programs with an analog* circuit, it is possible to create a grid of resistors, negative resistors, ideal diodes, and voltage sources that would work together to solve a linear programming problem. This would be the basis for an equality constraint, where R1 and R2 are given by the coefficients of the equalities, and the Ra value and Vb value could be found with coefficients. *(See Figure 1)* A similar circuit would be used to calculate an inequality constraint, using an ideal diode at point I.

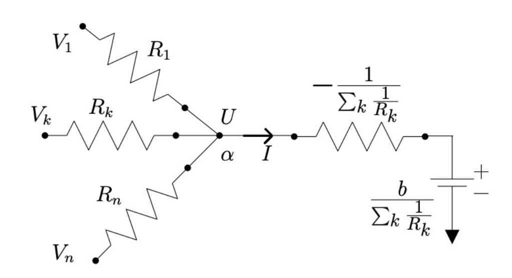

Figure 1: The basic component for an equality constraint [2]

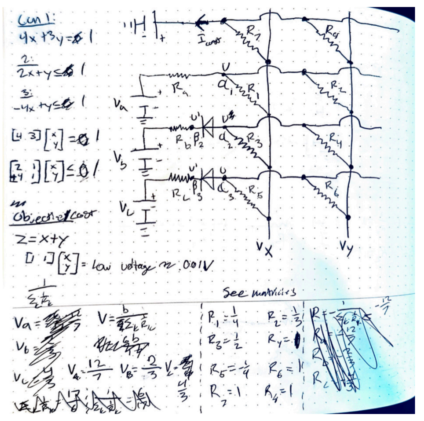

Figure 2: A drawn out LP solver circuit

- **Draw the circuit schematic**

This schematic shows what a completed LP solver would look like with one equality and two inequality constraints. The Vx and Vy voltages would be the solutions to the problem. *(See Figure 2)*

- **Model the circuit in LTSpice**

Modeling the circuit using a simulator such as LTSpice would allow me to test my circuit and verify my real-world output. Modeling the schematic was straightforward, as it was just bringing my drawing into a digital space. Figure 3 shows a complex circuit that I modeled to verify my solver could work if I added more equations. The third and fourth columns are to allow equations with –x and –y constraints.

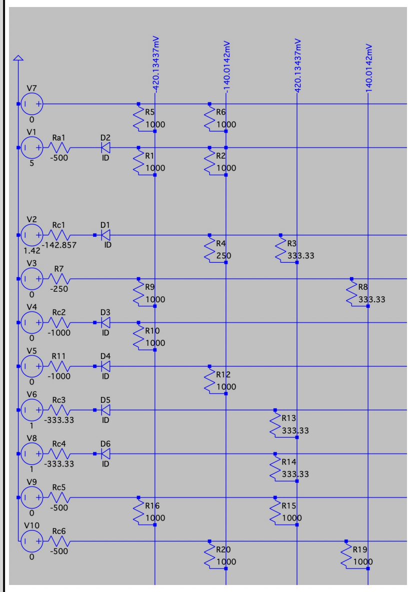

Figure 3: A complex LP solver circuit

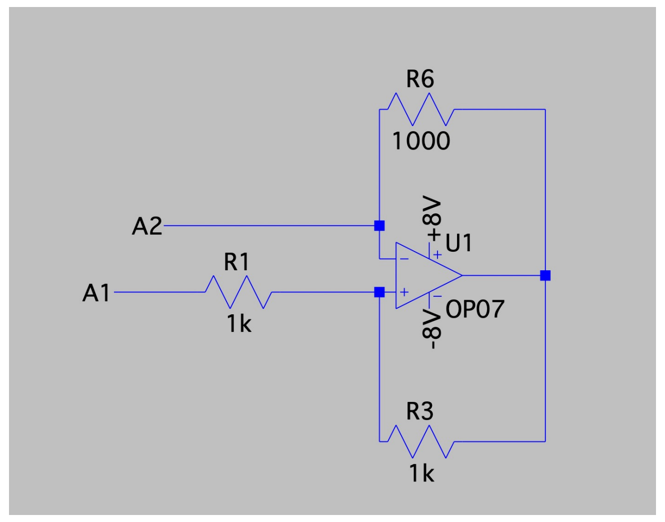

Figure 4: An experimental negative resistor

- **Making experimental negative resistors**

Building the experimental circuit was not straightforward. I first had to find an experimental circuit to model the output of a negative resistor. I ended up doing it using an operational amplifier, where R1 = R3, and R6 is the negative resistor value. This would replace all negative resistors in my experimental circuit.

Methods & Results

- **Build the experimental circuit & verify output**

Building the experimental circuit on the breadboard was simply laying out the components from my simulation. A simulation and a photo of the final circuit is shown below, where the experimental negative resistor is connected between A1-A2.

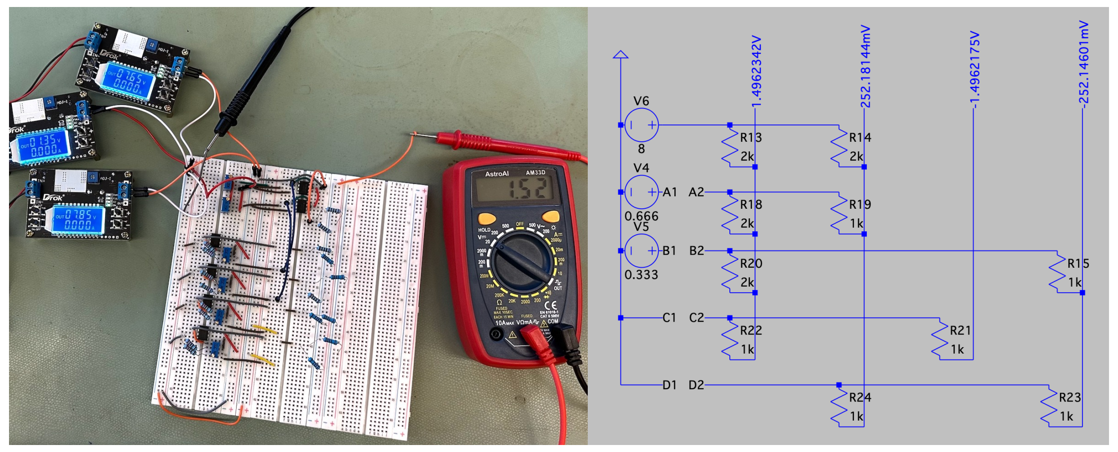

Figure 5: The finished LP solver circuit and schematic.

The solution to the LP problem was (1.5, 0.25). The simulation gave an output voltage of (1.48V, .252V). The real circuit gave an output of (1.52V, 0.21V). The deviation from the correct answer is due to resistor values not being exact.

- **Record the timing data for the experimental circuit**

After the output for the circuit was verified, the circuit was connected to a oscilloscope and was switched on. The time until the x value stabilized at the correct answer was recorded over six trials. *(See Figure 6)* The average time was about 20 microseconds from the time the first voltage increased happened to the stabilization of the results at ~1.5V.

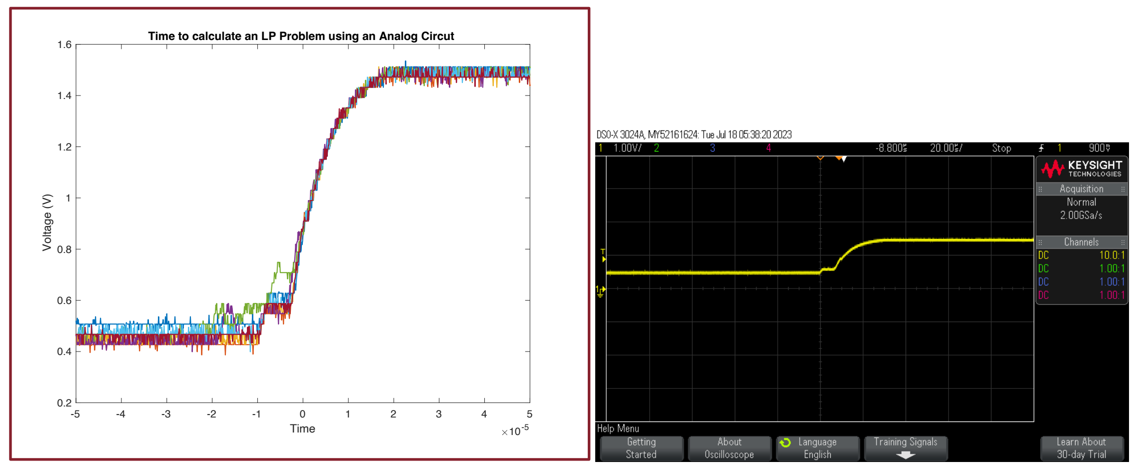

Figure 6: Experimental circuit timing results & Oscilloscope reading

- **Solve the LP Problem in MATLAB and record the data**

Solving the LP Problem in MATLAB took 160 milliseconds.

```
%Problem Setup
x = optimvar('x','LowerBound',-100,'UpperBound',100);
y = optimvar('y','LowerBound',-100,'UpperBound',100);
prob = optimproblem('Objective',x + y,'ObjectiveSense','min');
prob.Constraints.c1 = x/2 + y == 1;
prob.Constraints.c2 = x/2 - y == 1/2;
problem = prob2struct(prob);

tic %Timer start
[sol] = linprog(problem) %Solve Problem
timeElapsed = toc %Stop & Print timer
```

Command Window

```
Optimal solution found.

sol =

    1.5000
    0.2500

timeElapsed =

    0.1604
```

Results Analysis

Method	X	Y	Timing (s)
MATLAB	1.50	0.25	0.1629
Experimental Circuit	1.52	0.21	0.00002
Simulation	1.50	.252	0.000092

Figure: A table showing the solution and solve time for MATLAB, an analog circuit, and a circuit simulation for the same problem.

	MATLAB		ANALOG	
	Time (s)	Solution (+x)	Time (s)	Solution (+x)
Problem 0	0.1629	1.5	0.000092	1.5
Problem 1	0.1773	0.6154	0.000078	0.62
Problem 2	0.1618	1	0.000057	1
Problem 3	0.1721	-2.2222	0.000058	-2.2
Problem 4	0.1613	-0.8889	0.000055	-0.87

Figure: A table showing the x solution and the time taken between an analog circuit simulation and MATLAB.

After comparing the results of the analog and digital solvers, we can conclude that with the LP problems I tested, analog circuits are predicted to be 2,457x faster than their digital counterparts. Keep in mind this is a predicted performance with a simulator. The real analog circuit performed 8,145 than a digital computation of the same problem. These results are very promising and promote further exploration of this subject

Advice to a Future SHINE Student

I highly recommend taking advantage of your time in the lab as much as possible. You may think that seven weeks is a long time, but I still feel like I am just getting started with my project. I recommend finding ways to work on your project at home where you can, saving the lab time for tasks that can only be completed in the lab. Most importantly: Have fun and learn something!

Citations

Deems, Kristel. *High Speed Analog Circuit for Solving Optimization Problems*. 2015. [1]

[2] Vichik, S., & Borrelli, F. (2014). Solving linear and quadratic programs with an analog circuit. *Computers & Chemical Engineering*, *70*, 160-171. https://doi.org/10.1016/j.compchemeng.2014.01.011