Analog Computation

Presented by Mayuresh Kulkarni

Before the Digital Conquest

- Analog Computers

  Input variables are represented by one or more physical quantities. The laws of physics operate on these quantities to generate the required output.

- Early Analog Computers
  - Differential Analyzers, Planimeters, Integrators etc.

- Analog “computers” of a different kind
  - Folklore
  - Ingenuous devices constructed to solve one instance of a particular problem
  - Solve problems in one swoop

After the Digital Conquest

- Resurgence of Analog Computation
  - Can NP-Complete problems be *efficiently* solved using such devices?

Outline of the rest of the talk

- Concept of efficiency in analog computation.
- Description of devices for:
  - $Max(n_1, n_2)$
  - Primality detection
  - Minimal steiner tree construction
  - Sorting
  - Local optima.
- Strong Church thesis and limits on the efficiency of Analog computers.

Efficiency in analog computation

- The analog computer should use its resources *efficiently*.

  Examples of resources : physical size of the computer, energy used by the computer, time of operation till the solution is reached, maximum magnitude of the physical variables involved. Thus, for a Newtonian particle, its maximum displacement, velocity, acceleration and applied force are examples of resources. Values of these resources should be practically reasonable.

- Encoding of the inputs to the analog computer should be *efficient* :
  Consider that the same problem is to be solved using a digital computer and an analog computer. Then, the input needs to be encoded into suitable formats for the digital and analog computers. Let DIG and ANA represent the size of the same input value encoded into the two formats. Then, we say that the encoding for the analog computer is efficient if the growth of ANA is polynomially bounded by the growth of DIG

$Max(n_1, n_2)$

- Description

  Given two integers n_1 and n_2, create two particles of equal charge having masses m_1 and m_2. Place them in an uniform electric field. Then, the transit time from a fixed position to another fixed position is proportional to $\sqrt{m_1}$. ($t = \sqrt{\frac{2dm_1}{qE}}$). Thus, the smaller particle arrives at the desitination first.

- Issues
  - Does the analog computer use its resources efficiently? The resources, in this case, are the maximum velocity reached by the particles, the acceleration they are subject to, the size of the setup etc. For reasonable values of n, the above are all practically reasonable.
  - Is the encoding efficient? Let us have a function that maps a given number(n_i) into a mass-value(m_i) to be used as an input to the analog computer. i.e. we have a function $f(n_1) = m_1$. Then, the encoding is efficient if f is logarithmic. It is not efficient, say, if f is polynomial. (Why?)

Primality detection

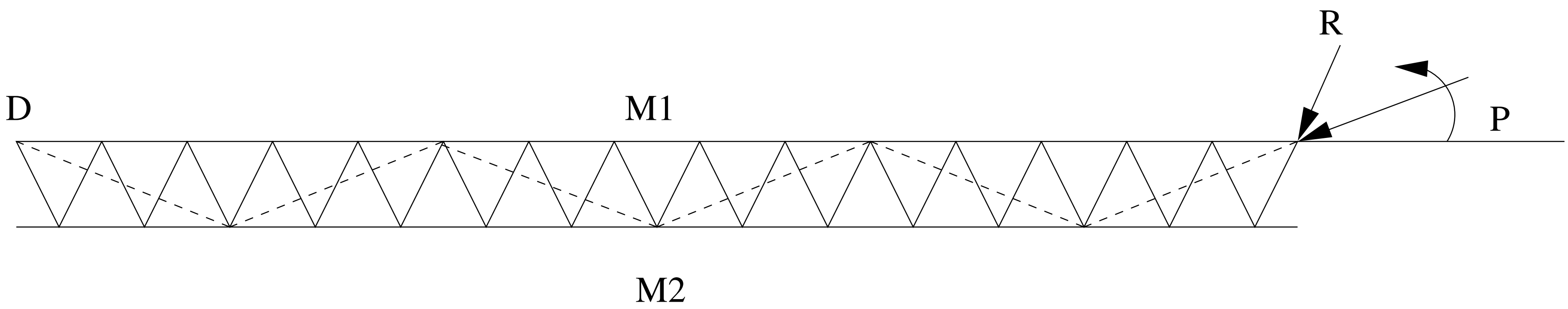

Figure 1: Lasers and Mirrors detect prime numbers

- Description:
  The mirrors M_1, M_2 and the laser R are initialized so that R bounces $n - 1$ times off M_1 and is absorbed at D. The laser P is then rotated from its initial horizontal position towards R. If, during the course of its rotation, it yields a reflection on M_1 at the same point(s) where the R beam is reflecting, and, at the same time, be absorbed at D, then and only then, n is prime. (Why?)

Construction of Minimal steiner trees

- Spanning Trees vs. Steiner Trees
- All Steiner points are functions of three lines forming $120°$ - Suggests using soap films. as shown in Figure 2. A, B, C and D are initial points. E and F are Steiner points.

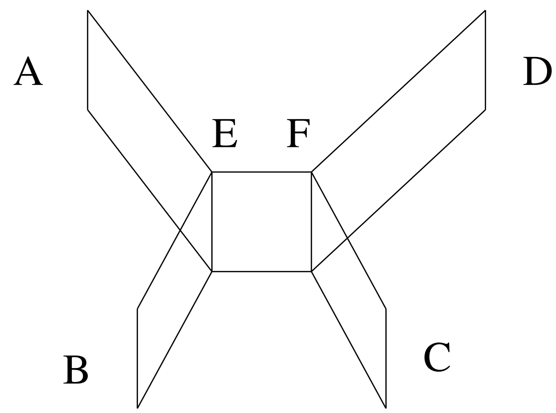

Figure 2: Soap films detect possibly minimal Steiner trees.

Sorting

- Comparison Networks and Zero-One Principle.

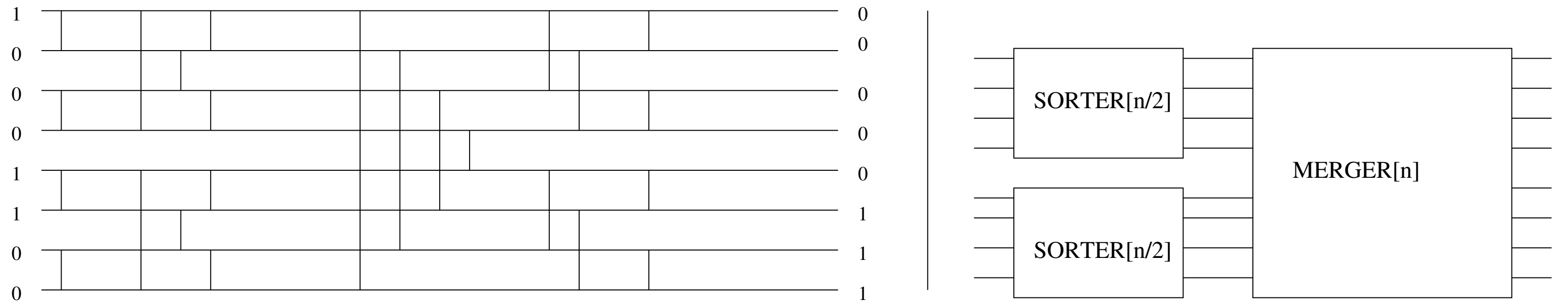

Figure 3: (a) Sorting network for $n = 8$ (b) Recursive structure of a general sorting network.

Local Optima

Let η denote the following linear programming problem:

$$\text{Maximize } z \text{ such that } \left\{ \begin{array}{rcl} z & = & 2x_1 + x_2 \\ x_1 + x_2 & \leq & 1 \\ x_1 & \geq & 0 \\ x_2 & \geq & 0 \end{array} \right\}$$

The following analog computer can be used to solve η: Each of the variables x_1, x_2 will be represented by the analog position of a shaft. Shaft positions can be negated, multiplied by a constant, added or subtracted using appropriate mechnical devices like coupling and differential gears. (Refer to Figure 4). The above primitives of multiplication and addition can be used to solve for η as shown in Figure 5.

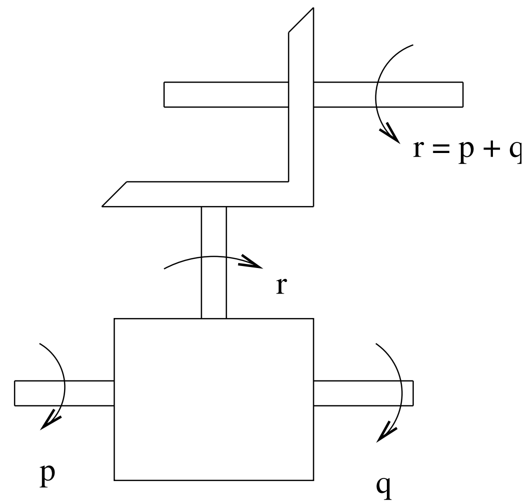

Figure 4: (a) $1 : c$ coupling gear (b) (Nonstandard) Differential gear to calculate $p + q$

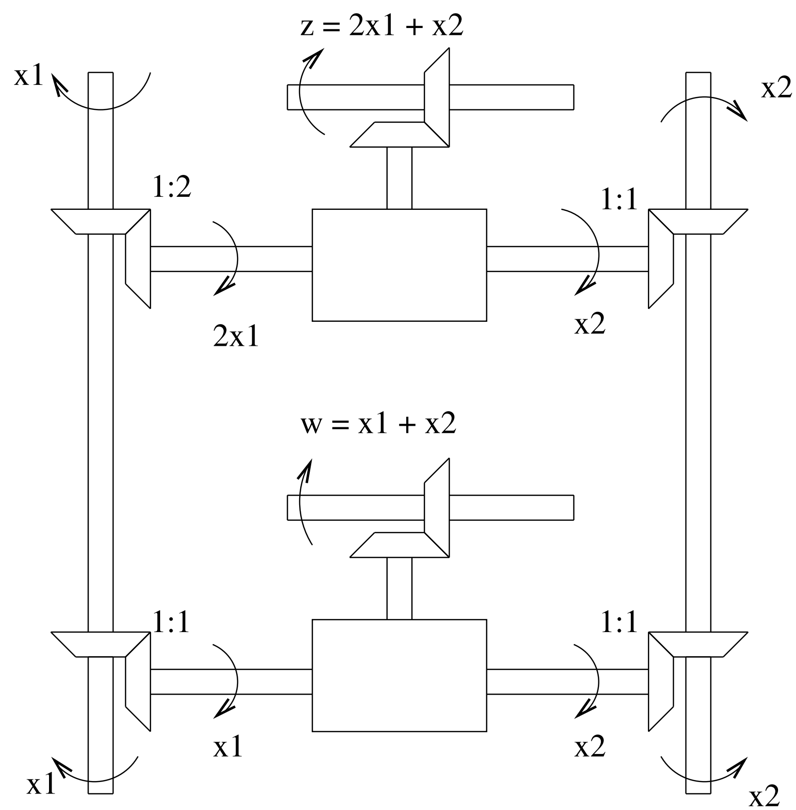

Figure 5: Local optima of system η

A lower bound on analog computation

The following is a description of the work done in [2]:

- Strong Church thesis :

  Any analog computer can be *efficiently* simulated by a digital computer.

- $3 - SAT$, which is NP-Complete, is reducible to a problem of checking whether a feasible point is a local optimum of an optimization problem.

- The mechanical device shown in Figure 5 is proposed for the solution to this problem.

- Assuming that the Strong Church thesis holds and $P \neq NP$, the authors conclude that the proposed device cannot operate successfully using only polynomial resources.

Conclusion

- It is likely that analog computers are no more efficient than digital computers.

References

[1] A. Dewdney, *The Turing Omnibus, 61 excursions in Computer Science*, Computer Science Press, Maryland 1989

[2] A. Vergis, D. Steiglitz and B. Dickinson, *The Complexity of analog computation*, Mathematics & Computers in Simulation, Volume 28, 1986, pages 91-113

[3] W. Paul, *A $2.5n$ lower bound on the combinational complexity of boolean functions*, Seventh annual ACM symposium on Theory of Computing, 1975

[4] T. Cormen, C. Leiserson, R. Rivest and C. Stein, *Introduction to Algorithms*, MIT Press, 2^{nd} edition, 2001