Solving NP-complete problems with delayed signals: an overview

Mihai Oltean, Oana Muntean

Babes-Bolyai University,

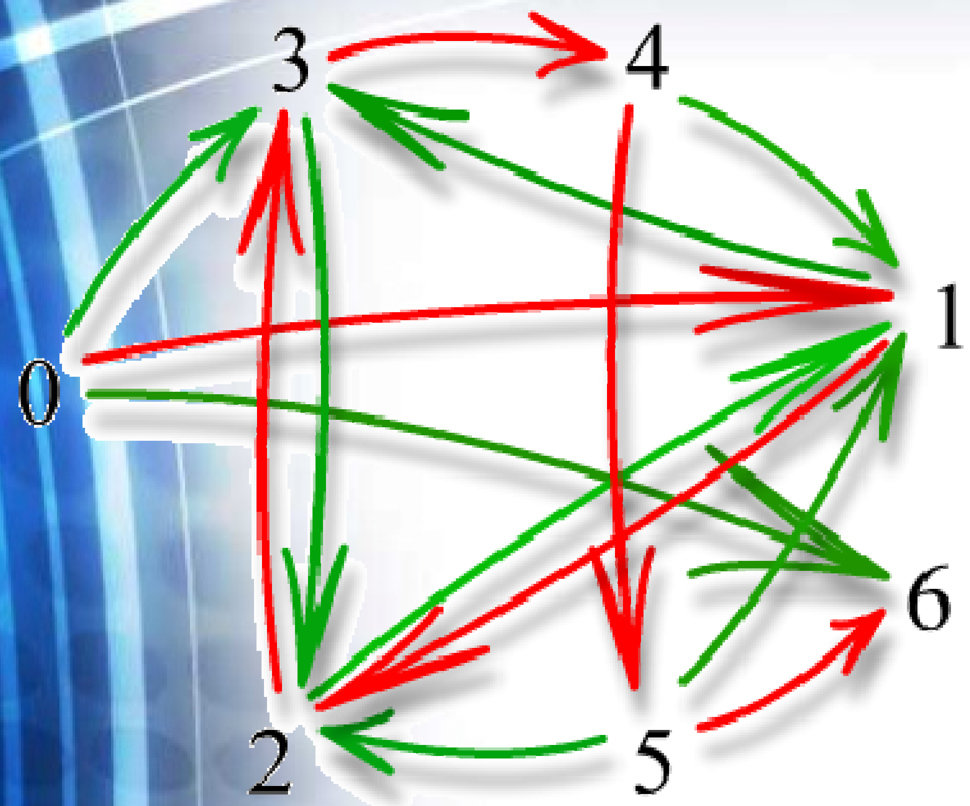
Cluj-Napoca, Romania

moltean@cs.ubbcluj.ro

Outline

- Problems that we want solve
- Properties of the signal that we can count on
- Basic idea
- Examples – Subset sum problem
- Possible implementations
  - Electric-based simulation
- Energy consumption
- What is Next ?

Problems that we solve



NP-complete problems.

- ✓Hamiltonian Path
- ✓Travelling Salesman
- ✓Subset Sum
- ✓Unbounded subset Sum
- ✓Exact Cover
- ✓Diophantine Equations

We solve the YES / NO decision problem.

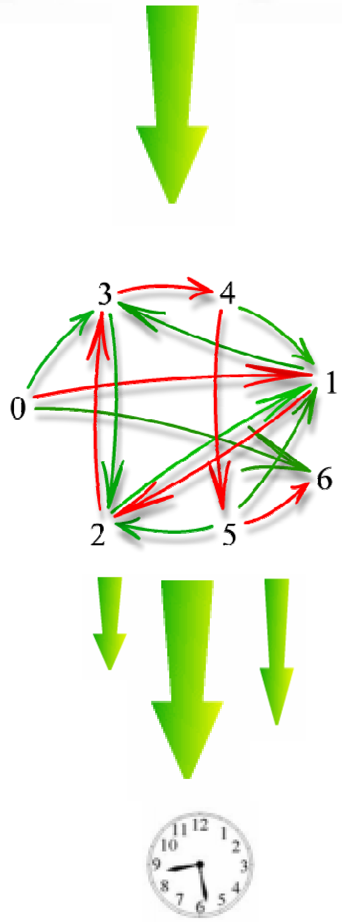
Tobias Haist did more.

Properties of the signal that we can count on

- The signal has a limited speed.
  - It can be delayed by forcing it to pass through a cable of a certain length.
- The signal can be divided into multiple (sub)signals of smaller intensity/power.
  - Massive parallelism.

Basic ideas

- Underlying mechanism: **Brute-force** approach.
  - All possible solutions are generated.
- The device has a directed graph-like structure.
  - There is a *Start* and a *Destination* node.
- Initially a signal is sent to the *Start* node.
- Operations performed with the signals:
  - Delayed by arcs.
  - Divided within nodes and sent to the nodes connected to the current node.
- Each solution follows a particular path in the graph.
  - It is delayed a particular quantity of time.
- At the *Destination* node we will search only for particular signals (arriving at some special moment – denoted by *M*).



Did we solve the problem ?

Different solutions arrive at different moments in the *Destination* node.

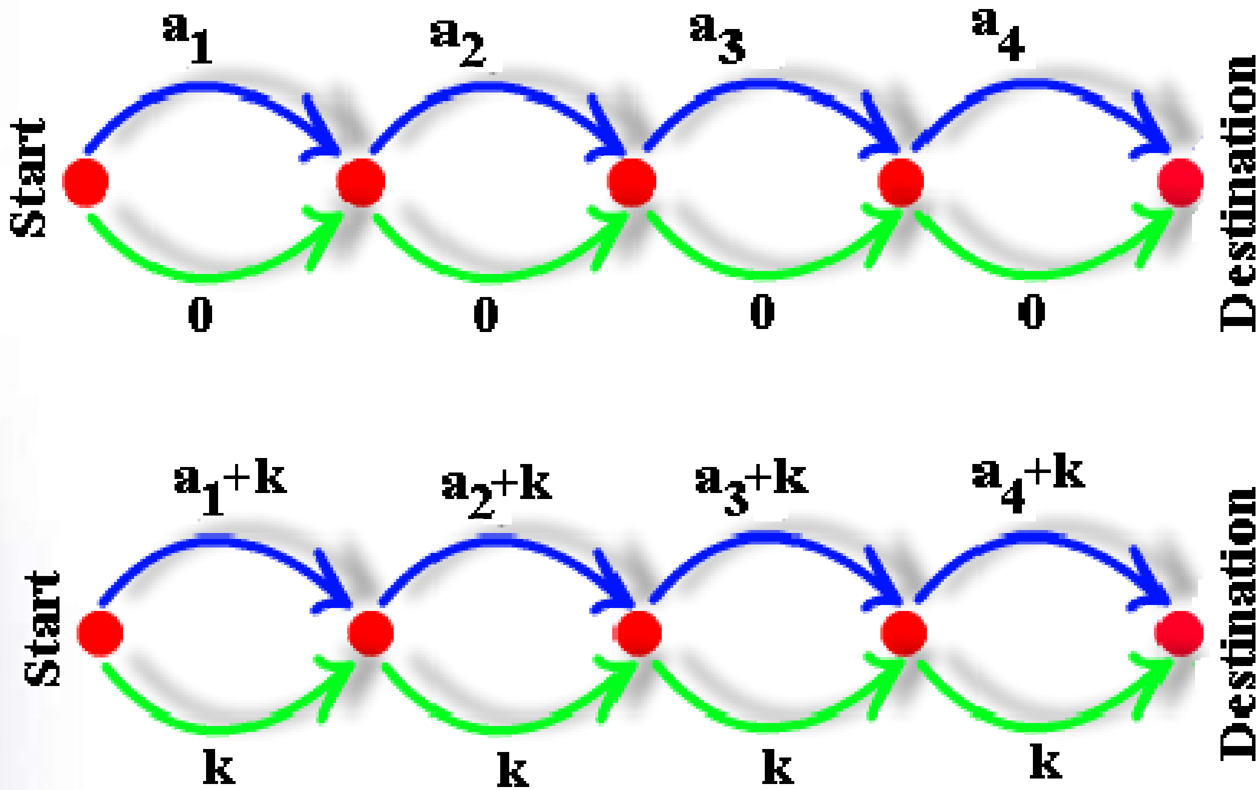
Is there is a signal arriving at moment *M* ?

- YES – problem solved.
- NO – no solution for the problem.

- Multiple signals arriving at moment *M* means multiple solutions for the problem

Example – Subset Sum

$A = \{a_1, a_2, a_3, a_4\}$.   Target sum = B

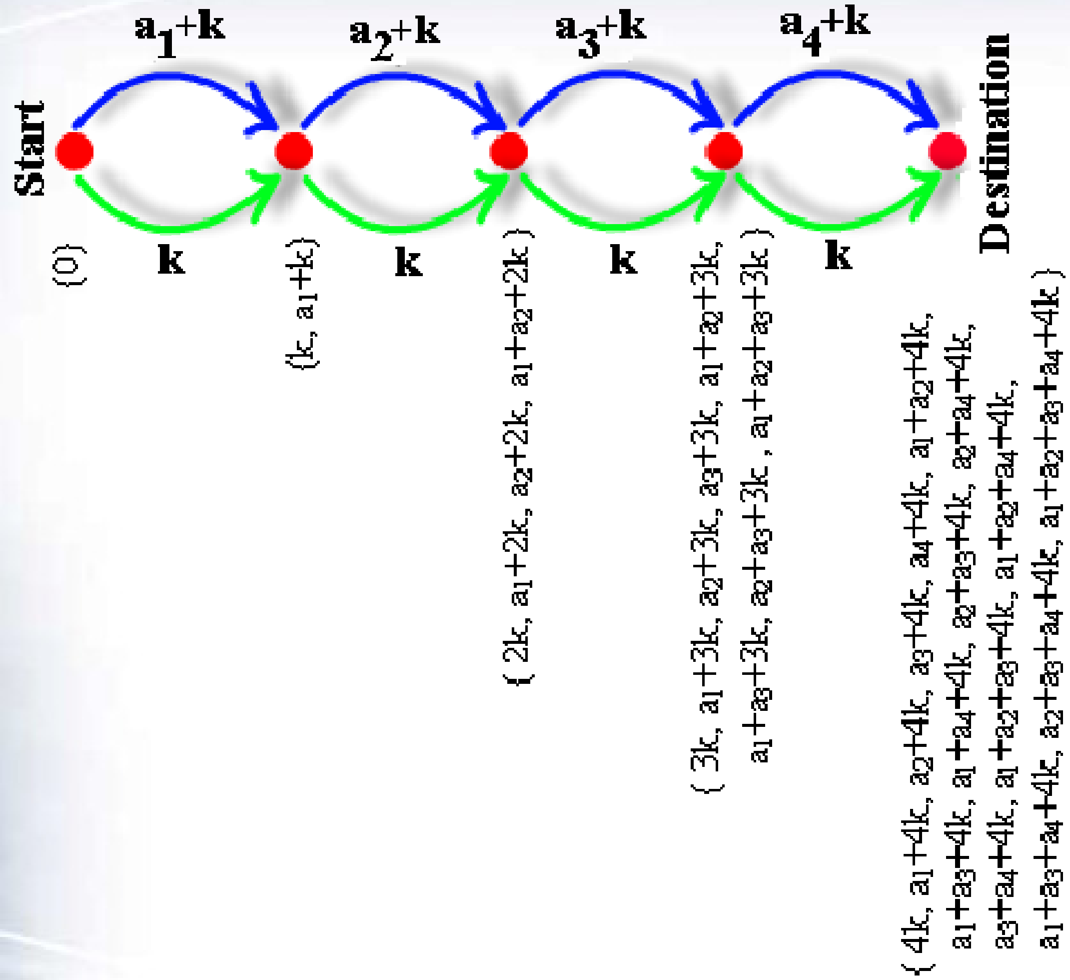


$M = B + n * k.$

Computer Simulation for Subset sum

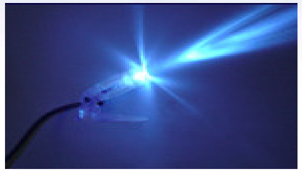
- http://www.youtube.com/watch?v=EVrQQI5qZhI

Fluctuation moments

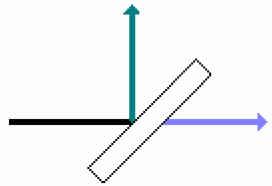


Hardware implementation

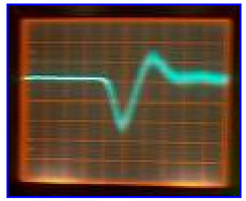
- A source of signals (laser, pulse generator, etc)



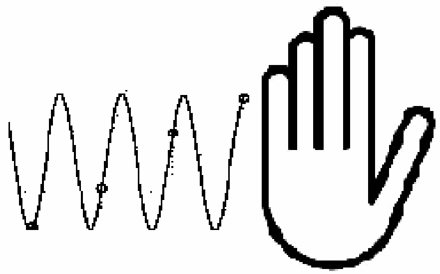
- Several splitters for dividing the signal into multiple subsignals



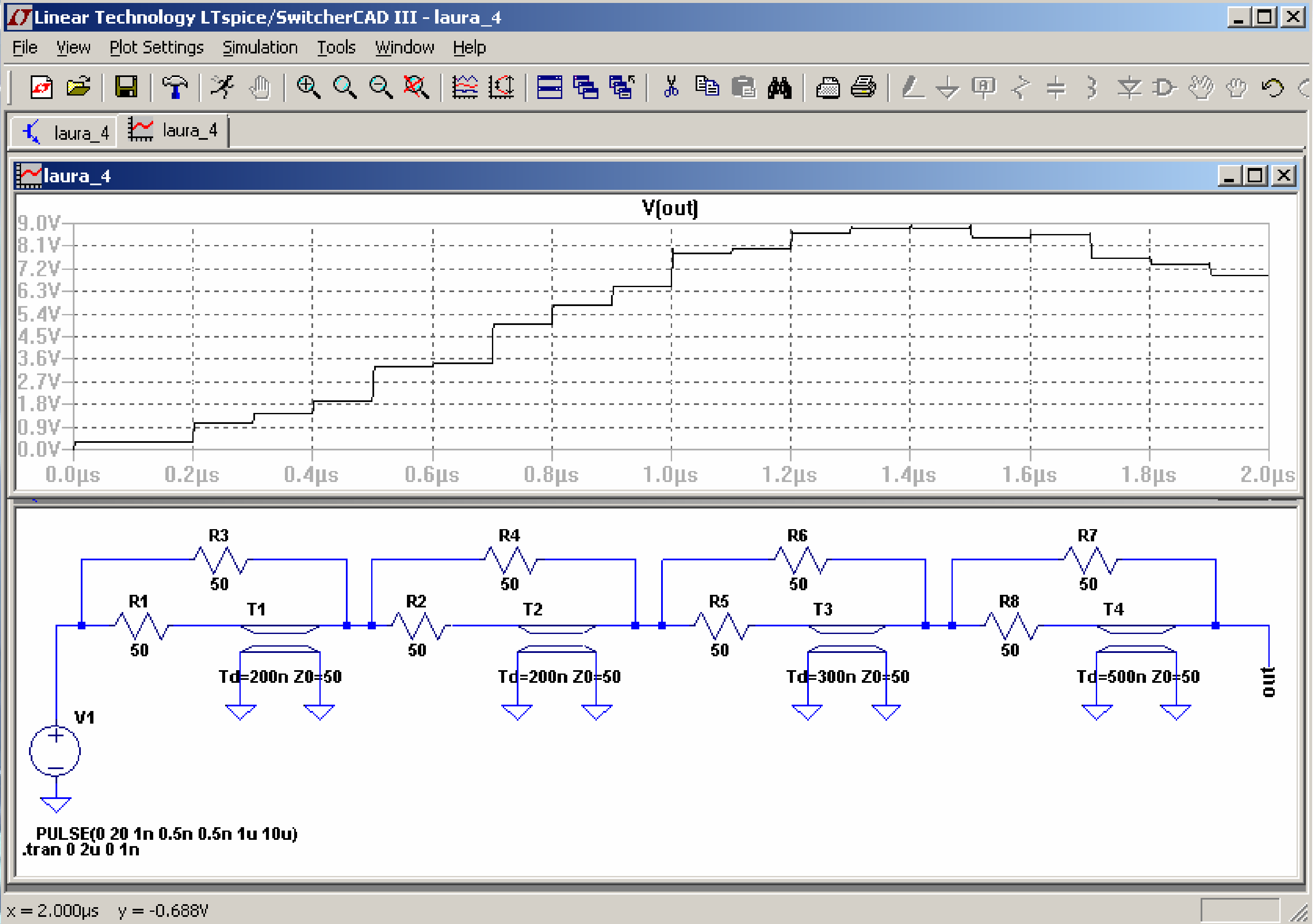
- A device for detecting fluctuations in the intensity of signal (oscilloscope for electric signals)



- Delay lines
  - cables having certain lengths.
  - by discrete inductors and capacitors.



Working with electric signals



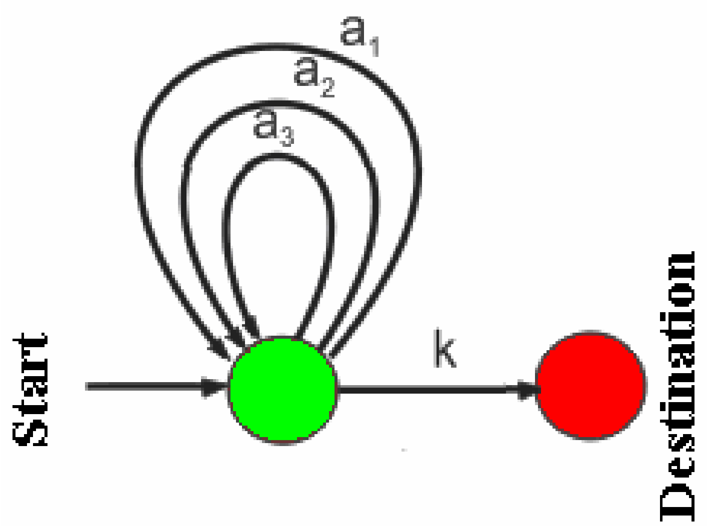
Electrical delay lines

- Programmable delay lines:
  - Number of steps: 256 (8 bits)
  - Min delay: 16.5 ns
  - Max delay: 1275 ns
  - Price: 7$ @ 1k

- Extended range by serialization !

Devices that cannot be implemented electrically

- Unbounded subset sum problem.



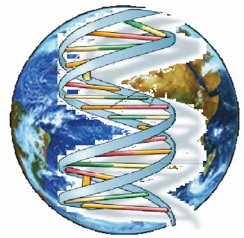
- But optically we can !

Energy consumption

- Signals are divided multiple times. The intensity of the signal decreases exponentially.

- Subset Sum 2^N
- Hamiltonian path N^N

- DNA computers trying to solve 200-nodes Hamiltonian Path require a quantity of DNA
- equal to the size of the Earth (Hartmanis '95).



Other drawbacks

- Cannot compute the actual solution in the case of YES answer. *Tobias Haist* has shown how to do that.
- Moment *M* cannot be measured exactly (with any precision).
  - We can mixed up solutions. ☹
- Large instances ???

What is next: Automation

- Each problem requires its own graph.
  - Each instance may require an extension of some existing graphs or some new graphs.

- What we need:
  - a scalable and reconfigurable graph.
  - programmable / reconfigurable delay lines.

Conclusions

- Unconventional devices for solving NP-complete problems.
  - Massive parallelism of light could be the key feature.
- Simulations ok …
- Physical implementation is possible, but still a lot of challenges to fight with.