Improvements on Circuit Lattices

Igor Semaev
University of Bergen
Norway

Norsk Kryptoseminar, 10 November 2011
(also Dagstuhl Seminar, September 2011)

Introduction. Outline

- Motivation
- Definitions
- Basic Algorithms

Motivation

- One way function $x \to f(x)$
- E.g. $x \to a^x \mod p$
- M - plain-text,
  K - key,
  $E_K(M)$ cipher-text in DES(AES):

$$K \to E_K(M)$$

- Still one-way

Motivation

- **Compute** with low number of small gates

$$f(x_1, x_2, x_3, x_4) = F(g_1(x_1, x_2), g_2(x_2, x_3), g_3(x_3, x_4))$$

- **Invert:** solve $f(x) = y$ in x
- **Simplify:** introduce new variables

$$f(x_1, x_2, x_3, x_4) = y \Leftrightarrow \begin{array}{r} g_1(x_1, x_2) = y_1 \\ g_2(x_2, x_3) = y_2 \\ g_3(x_3, x_4) = y_3 \\ F(y_1, y_2, y_3) = y \end{array}$$

- 3-sparse equations system

DES and TripleDES equations

- 64-bit plain-text, cipher-text, convenient to write variables
- 64-bit internal state blocks and 56(112)-bit key
- Equations from S-boxes(6-bit$\rightarrow$ 4-bit)

$$Y_4 \oplus Z_4 = S(X_6 \oplus K_6)$$

- 20 variables(20-sparse), 2^{16} solutions each
- DES: 632 variables, 128 equations
- TDES: 1712 variables, 384 equations

Zakrevskij-Raddum representation

- $f_i(X_i) = 0 \Leftrightarrow$ solutions V_i in variables $X_i \Leftrightarrow E_i = (X_i, V_i)$

$$x_1 x_2 + x_3 \equiv 0 \mod 2 \Leftrightarrow \begin{array}{ccc} x_1 & x_2 & x_3 \\ \hline 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{array}$$

- **Solve with**:
- Gluing(enlarge equations by combining)
- Guess variable values
- Pairwise Agreeing(propagation, decision)

Local Reduction (Pairwise Agreeing)

x_1	x_2	x_3
0	0	1
0	0	0
0	**1**	**0**
1	1	1

x_1	x_2	x_4
0	0	0
1	**0**	**1**
1	1	0
1	1	1

- Common variables $\{x_1, x_2\}$
- Projections on $\{x_1, x_2\}$:
- $00, 01, 11$ and $00, 10, 11$
- Remove vectors with projection not in the projections of another list
- Due to [Zakrevskij-Vasilkova,00] and [Raddum,04]

Agreeing Algorithm

- **Repeat**:
- Find E_i and E_j which disagree
- Remove some local solutions in E_i or E_j and make them agree.

Related Algorithms Running Time(q=2)

n l-sparse Boolean equations in n variables

$l =$	3	4	5	6
the worst case	1.324^n	1.474^n	1.569^n	1.637^n
expectation[Semaev,10]	1.029^n	1.107^n	1.182^n	1.239^n

.

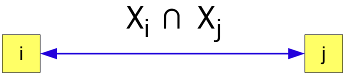
- Worst and average cases of the problem are excitingly different
- A software implementation is comparable with SAT-solvers in speed[Schilling-Raddum,10]

Circuit Lattices. Contribution Outline

- Equation Graph simplification, New versus [Semaev,WCC'09]
- A faster agreeing [Raddum-Semaev,07]
- Circuit Lattices, New versus [WCC'09]
- Circuit Lattice for TripleDES

Equation Graph and Pairwise Agreeing

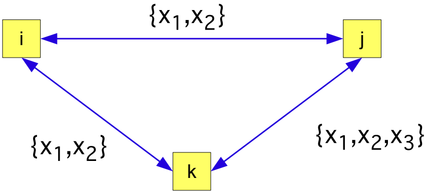
- Connect $E_i = (X_i, V_i)$ and $E_j = (X_j, V_j)$ by



- if $X_i \cap X_j \neq \emptyset$
- **Pairwise Agreeing:**
- Learn $X_i \cap X_j \neq a$ from E_i. Expand to E_j
- or vice versa

Obsolete Edges

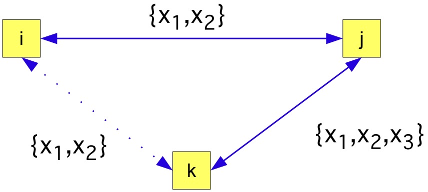
- Remove some edges and keep Algorithm's output



- 6 connections(arcs) initially

Obsolete Edges

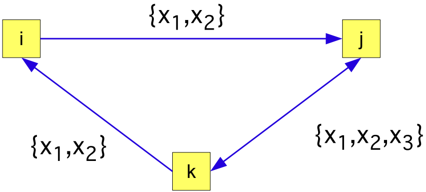
- Remove some edges and keep Algorithm's output



- 4 connections(arcs) as in WCC'09

Obsolete Edges

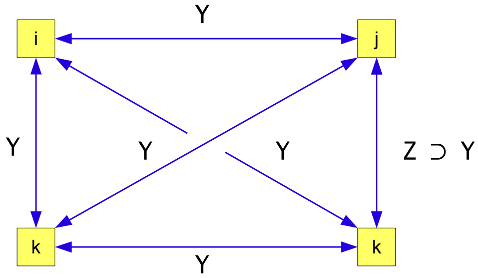
- Remove some edges and keep Algorithm's output



- 4 connections(arcs) now

Obsolete Edges

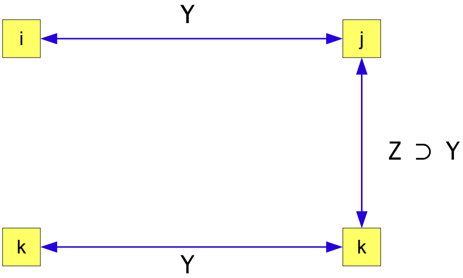
- Remove some edges and keep Algorithm's output



- 12 connections(arcs) initially

Obsolete Edges

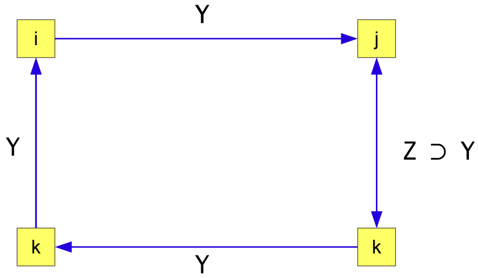
- Remove some edges and keep Algorithm's output



- 6 connections(arcs) as in WCC'09

Obsolete Edges

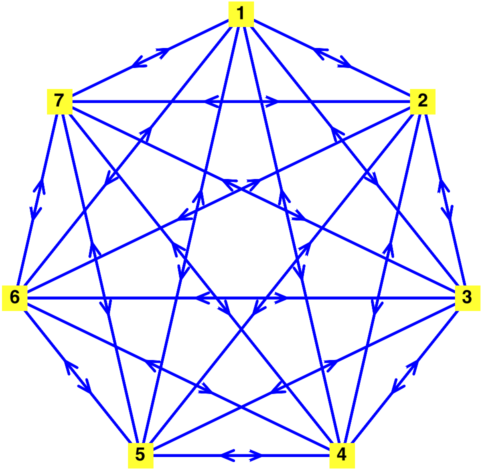
- Remove some edges and keep Algorithm's output



- 5 connections(arcs) now

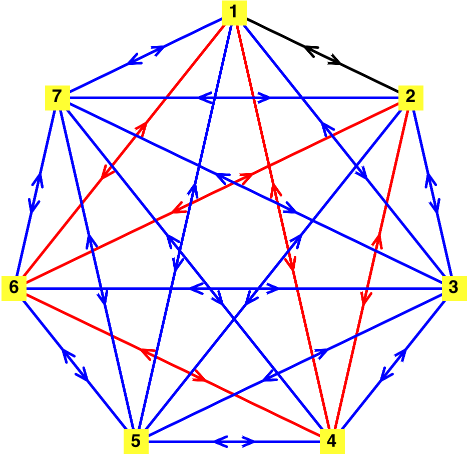
Hidden Cyclic Structure

$$\{x_1, x_2, x_4, x_6\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$
$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, x_7\}, \{x_1, x_3, x_5, x_7\}$$



Hidden Cyclic Structure

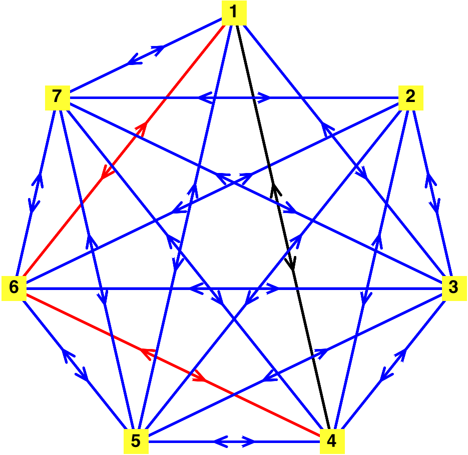
$$\{x_1, \mathbf{x_2}, x_4, x_6\}, \{\mathbf{x_2}, x_3, x_5, x_7\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$
$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, x_7\}, \{x_1, x_3, x_5, x_7\}$$



Hidden Cyclic Structure

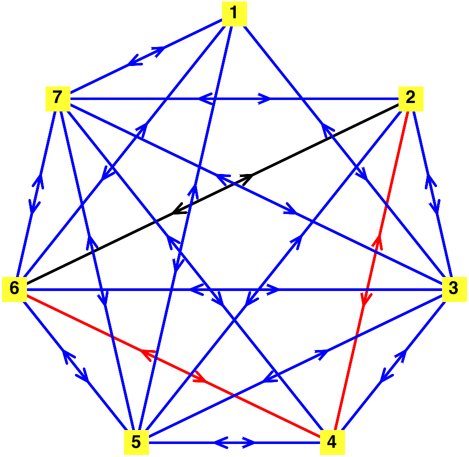
$$\{x_1, \mathbf{x_2}, \mathbf{x_4}, x_6\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, x_4, x_6\}, \{\mathbf{x_2}, \mathbf{x_4}, x_5, x_7\},$$

$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, x_7\}, \{x_1, x_3, x_5, x_7\}$$



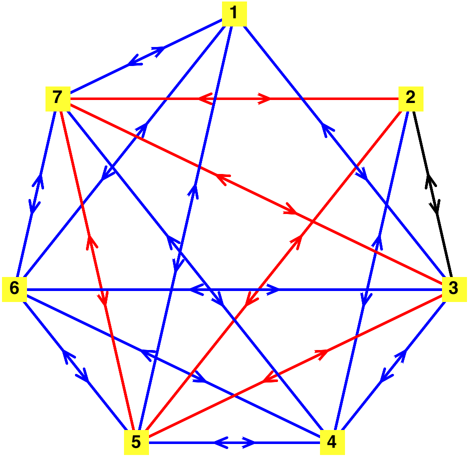
Hidden Cyclic Structure

$$\{x_1, x_2, x_4, x_6\}, \{\mathbf{x_2}, x_3, x_5, \mathbf{x_7}\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$
$$\{x_1, x_3, x_5, x_6\}, \{\mathbf{x_2}, x_4, x_6, \mathbf{x_7}\}, \{x_1, x_3, x_5, x_7\}$$



Hidden Cyclic Structure

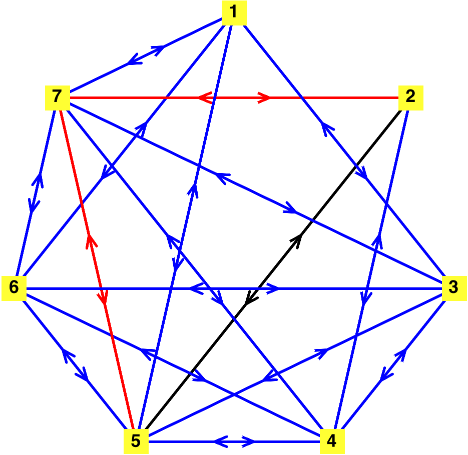
$$\{x_1, x_2, x_4, x_6\}, \{x_2, \mathbf{x_3}, x_5, x_7\}, \{x_1, \mathbf{x_3}, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$
$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, x_7\}, \{x_1, x_3, x_5, x_7\}$$



Hidden Cyclic Structure

$$\{x_1, x_2, x_4, x_6\}, \{x_2, \mathbf{x_3}, \mathbf{x_5}, x_7\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$

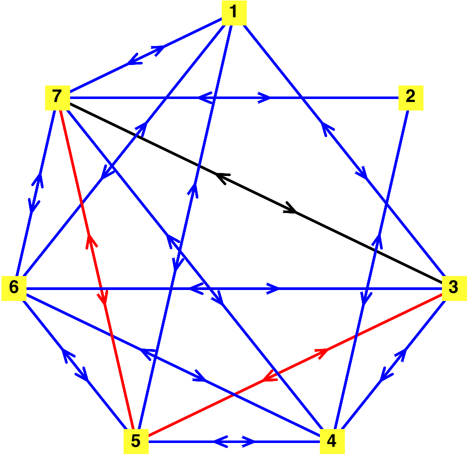
$$\{x_1, \mathbf{x_3}, \mathbf{x_5}, x_6\}, \{x_2, x_4, x_6, x_7\}, \{x_1, x_3, x_5, x_7\}$$



Hidden Cyclic Structure

$$\{x_1, x_2, x_4, x_6\}, \{x_2, x_3, x_5, x_7\}, \{\mathbf{x_1}, \mathbf{x_3}, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$

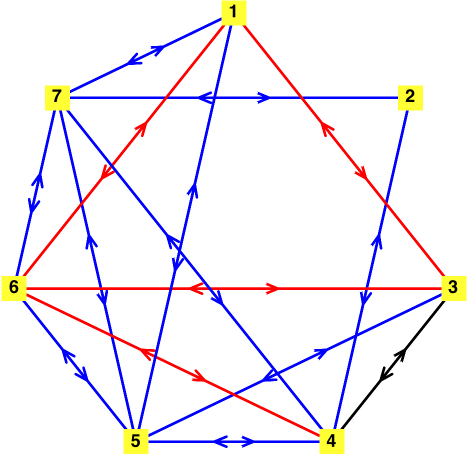
$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, x_7\}, \{\mathbf{x_1}, \mathbf{x_3}, x_5, x_7\}$$



Hidden Cyclic Structure

$$\{x_1, x_2, x_4, x_6\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, \mathbf{x_4}, x_6\}, \{x_2, \mathbf{x_4}, x_5, x_7\},$$

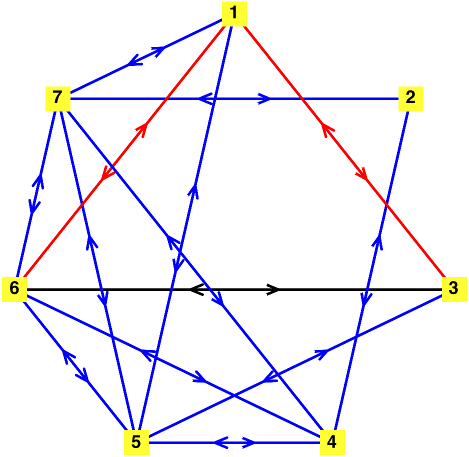
$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, x_7\}, \{x_1, x_3, x_5, x_7\}$$



Hidden Cyclic Structure

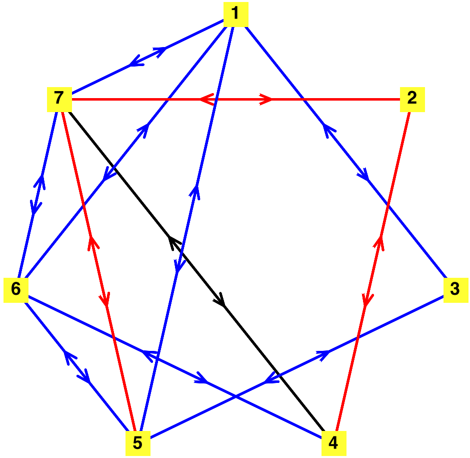
$$\{x_1, x_2, x_4, x_6\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, \mathbf{x_4}, \mathbf{x_6}\}, \{x_2, x_4, x_5, x_7\},$$

$$\{x_1, x_3, x_5, x_6\}, \{x_2, \mathbf{x_4}, \mathbf{x_6}, x_7\}, \{x_1, x_3, x_5, x_7\}$$



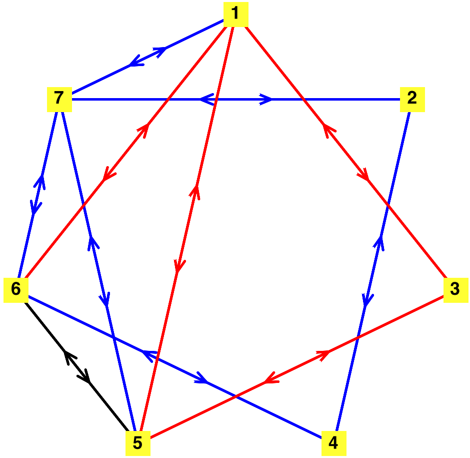
Hidden Cyclic Structure

$$\{x_1, x_2, x_4, x_6\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, \mathbf{x_5}, \mathbf{x_7}\},$$
$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, x_7\}, \{x_1, x_3, \mathbf{x_5}, \mathbf{x_7}\}$$



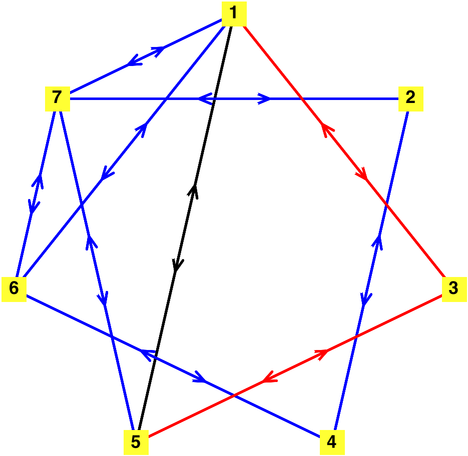
Hidden Cyclic Structure

$$\{x_1, x_2, x_4, x_6\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$
$$\{x_1, x_3, x_5, \mathbf{x_6}\}, \{x_2, x_4, \mathbf{x_6}, x_7\}, \{x_1, x_3, x_5, x_7\}$$



Hidden Cyclic Structure

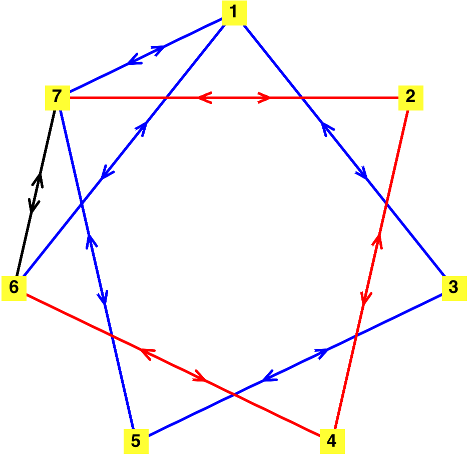
$$\{\mathbf{x_1}, x_2, x_4, \mathbf{x_6}\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$
$$\{\mathbf{x_1}, x_3, x_5, \mathbf{x_6}\}, \{x_2, x_4, x_6, x_7\}, \{x_1, x_3, x_5, x_7\}$$



Hidden Cyclic Structure

$$\{x_1, x_2, x_4, x_6\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$

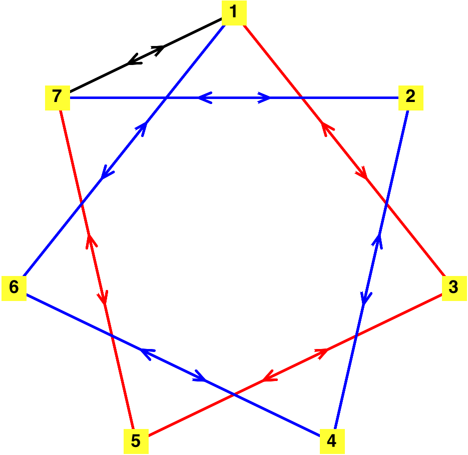
$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, \mathbf{x_7}\}, \{x_1, x_3, x_5, \mathbf{x_7}\}$$



Hidden Cyclic Structure

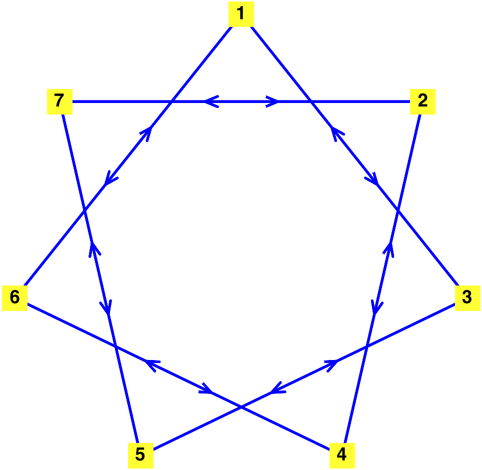
$$\{\mathbf{x_1}, x_2, x_4, x_6\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$

$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, x_7\}, \{\mathbf{x_1}, x_3, x_5, x_7\}$$



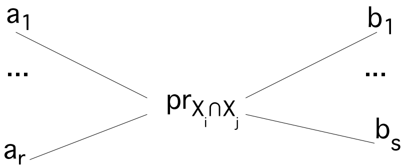
Hidden Cyclic Structure

$$\{x_1, x_2, x_4, x_6\}, \{x_2, x_3, x_5, x_7\}, \{x_1, x_3, x_4, x_6\}, \{x_2, x_4, x_5, x_7\},$$
$$\{x_1, x_3, x_5, x_6\}, \{x_2, x_4, x_6, x_7\}, \{x_1, x_3, x_5, x_7\}$$



Faster Pairwise Agreeing

- $E_i \to E_j$
- $a_1, \ldots, a_r$ and $b_1, \ldots, b_s$ local solutions to E_i and E_j
- with the same projection to $X_i \cap X_j$



- Pre-compute all such tuples $(a_1, \ldots, a_r; b_1, \ldots, b_s)$

Faster Pairwise Agreeing

- **Notation:** $a_i \neq$ part of a global solution $\Rightarrow$ mark $\bar{a}_i$
- $(a_1, \ldots, a_r; b_1, \ldots, b_s)$ equivalent to
- $\bar{a}_1, \ldots, \bar{a}_r \Rightarrow \bar{b}_1, \ldots, \bar{b}_s$
- **Solving the system**:
- Introduce a guess $\equiv$ mark some of a_i
- Expand marking through the imlications

Example

- Equations by local solutions:

	x_1	x_2	x_3
a_1	0	0	1
a_2	0	1	1
a_3	1	1	0

,

	x_1	x_4
b_1	0	1
b_2	1	0

,

	x_2	x_3	x_4
c_1	0	1	1
c_2	1	0	1
c_3	1	1	0

- Tuples

$$
\begin{array}{lll}
(a_1, a_2; b_1), (b_1; a_1, a_2), & (a_3, b_2), (b_2; a_3), & (b_1; c_2), (c_2; b_1) \\
(c_1, c_3; b_2), (b_2; c_1, c_3), & (a_1; c_1), (c_1; a_1), & \\
(a_2; c_3),\ (c_2; a_3) & &
\end{array}
$$

Example

- Assume $x_4 = 0 \Rightarrow b_1$ should be marked(wrong local solution)
- Marking expansion

$$\begin{tikzcd}
(b_1;\, a_1, a_2) \arrow[r] \arrow[d] & (a_1;\, c_1) \arrow[d] & & & \\
(a_2;\, c_3) \arrow[r] & (c_1, c_3;\, b_2) \arrow[r] & (b_2;\, a_3) \arrow[r] & (a_3;\, c_2) \arrow[r] & (c_2;\, b_1)
\end{tikzcd}$$

- All instances(b_2 at early stage) got marked
- The system is inconsistent for $x_4 = 0$

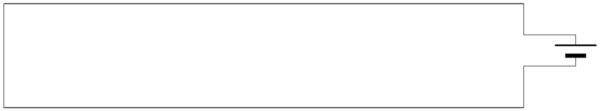
Circuit Lattice (Basic Construction)

- Circuit Lattice is a combination of switches and wires
- Two types of switches:

- 1-Switch controls vertical circuit by the horizontal
- 2-Switch controls horizontal circuit by the vertical

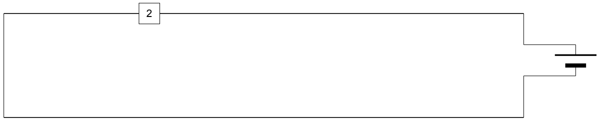
Constructing the Lattice

Local solution $\Leftrightarrow$ Horizontal circuit
Local solution wrong $\Leftrightarrow$ Potential in the circuit



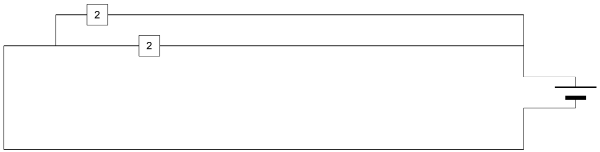
Constructing the Lattice

2-switch controls the circuit



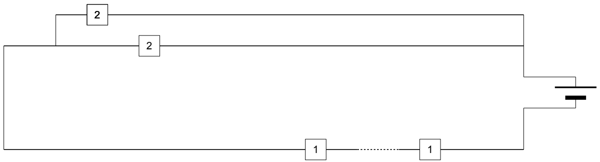
Constructing the Lattice

Several 2-switches may control the circuit



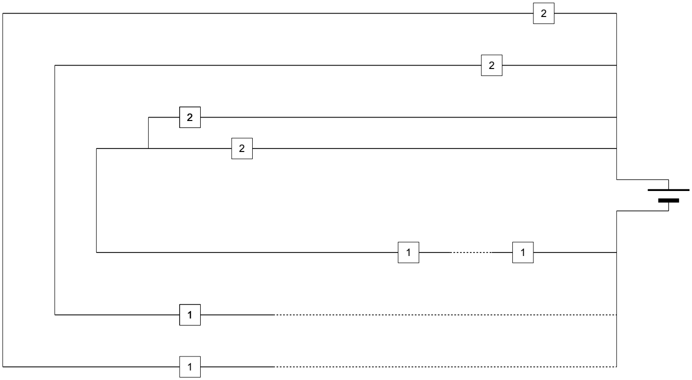
Constructing the Lattice

1-switches control some vertical circuits



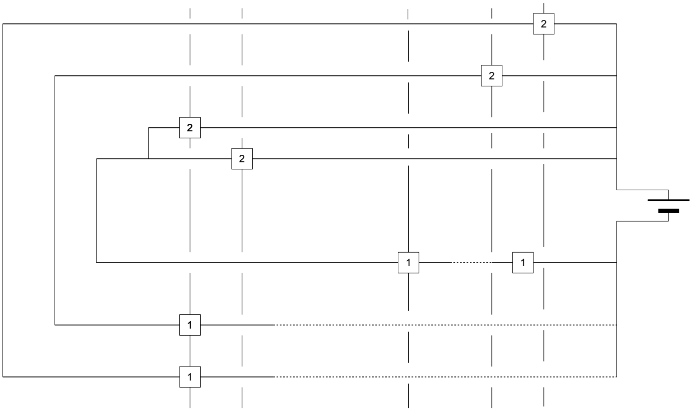
Constructing the Lattice

Many horizontal circuits



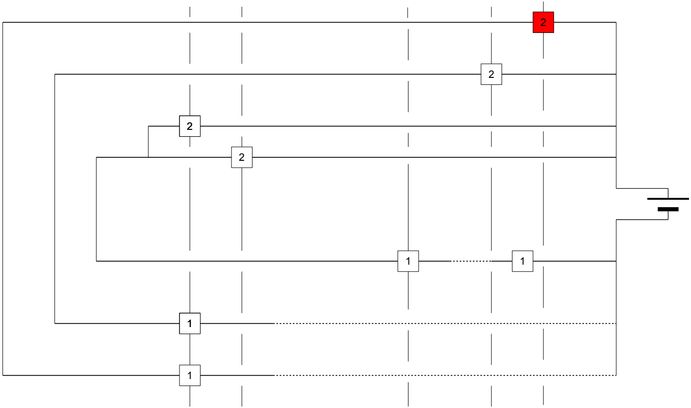
Constructing the Lattice

Tuple $\Leftrightarrow$ Vertical circuit



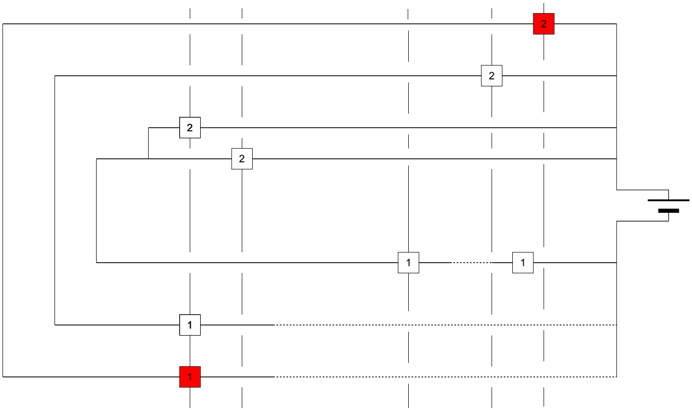
How It Works

Inducing potential in some circuits



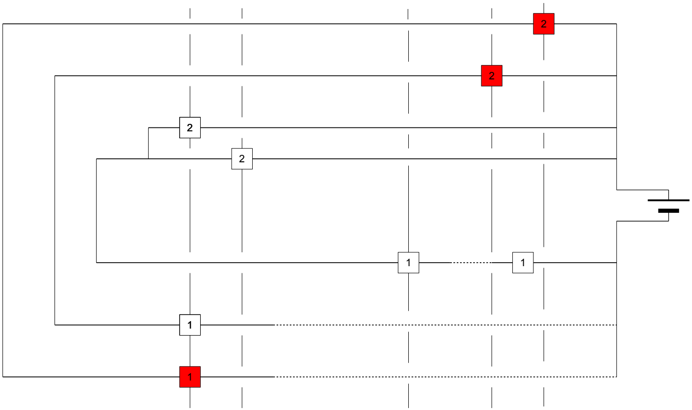
How It Works

Expands potential to new circuits



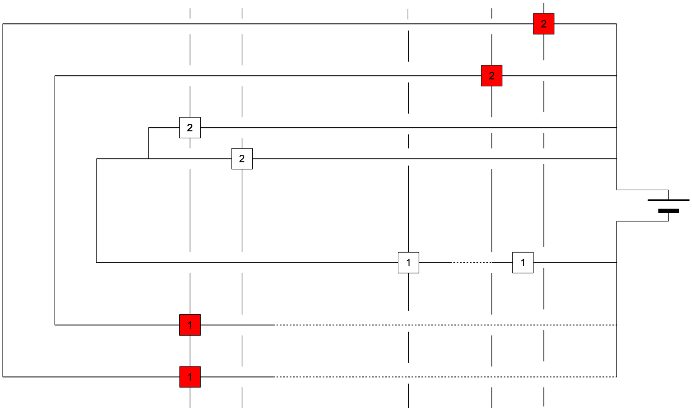
How It Works

Expands potential to new circuits



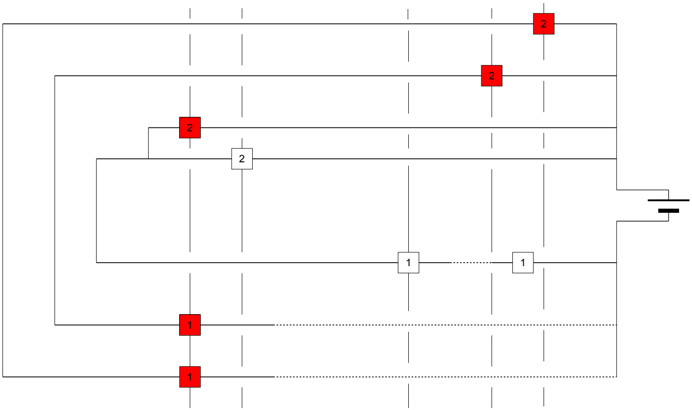
How It Works

Expands potential to new circuits



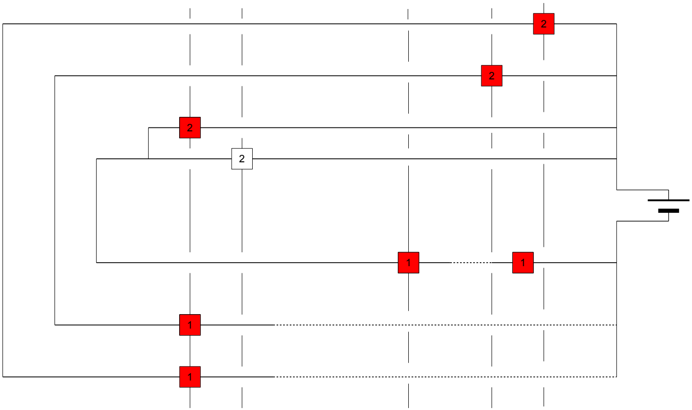
How It Works

Expands potential to new circuits



How It Works

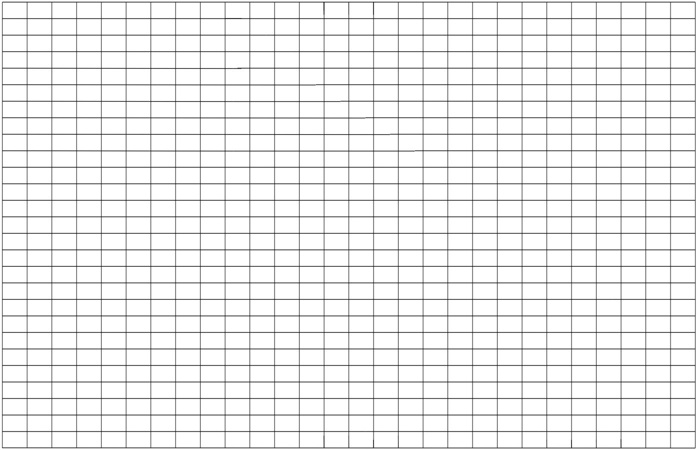
Expands potential to new circuits



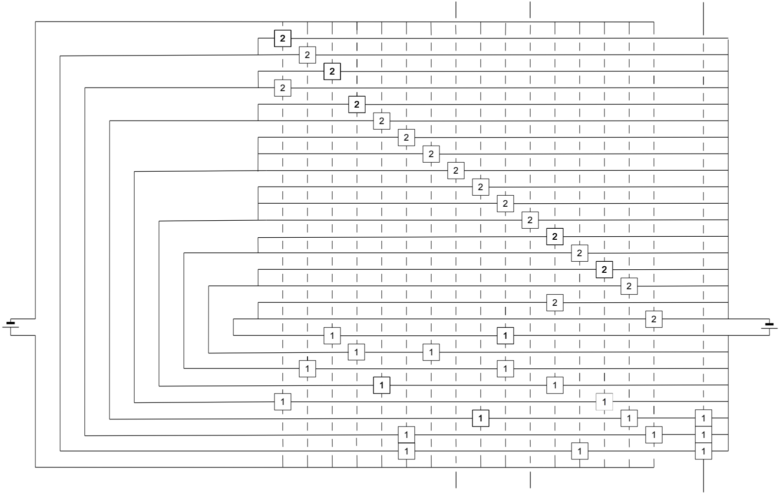
Introduce the guess

- Generally, no voltage in initial circuit lattice
- Assume E_i depends on x_j
- $a_1, \ldots, a_2$ solutions to E_i, where $x_j = 0$
- Add 2-Switch to each $a_1, \ldots, a_2$, connect them
- Guessing $x_j = 0$ is inducing voltage in new circuit
- Similarly, guessing $x_j = 1$
- s-variable guess - $2s$ new vertical circuits

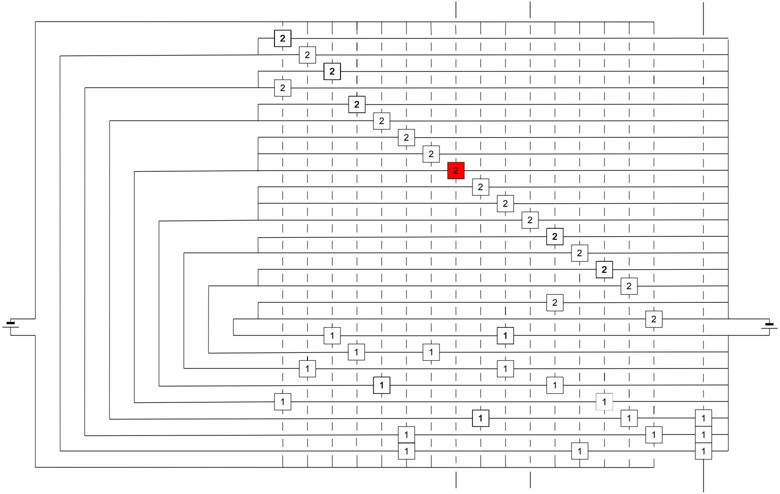
Grid Lattice



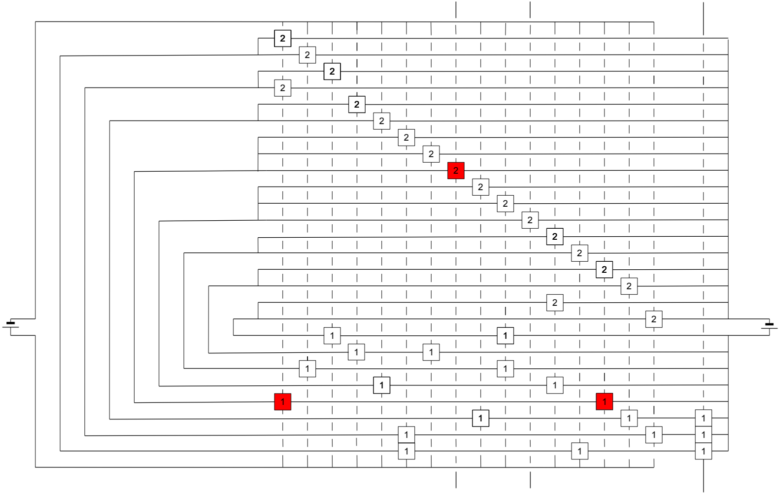
Exemplary Circuit Lattice



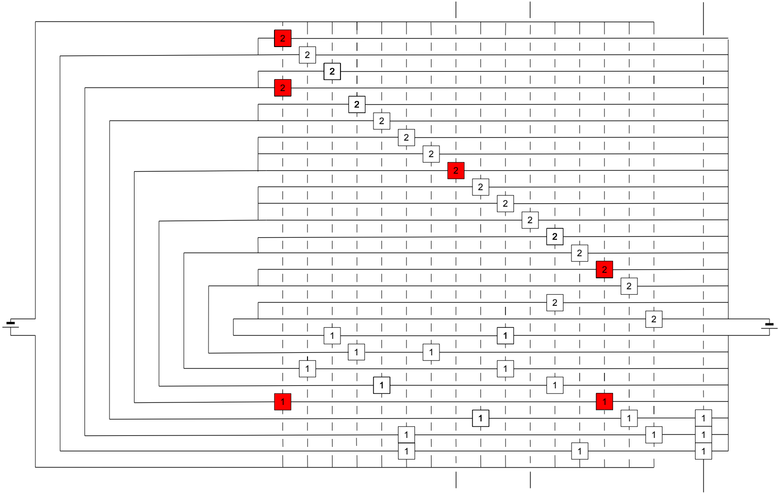
Exemplary Circuit Lattice. Introduce guess $x_4 = 0$



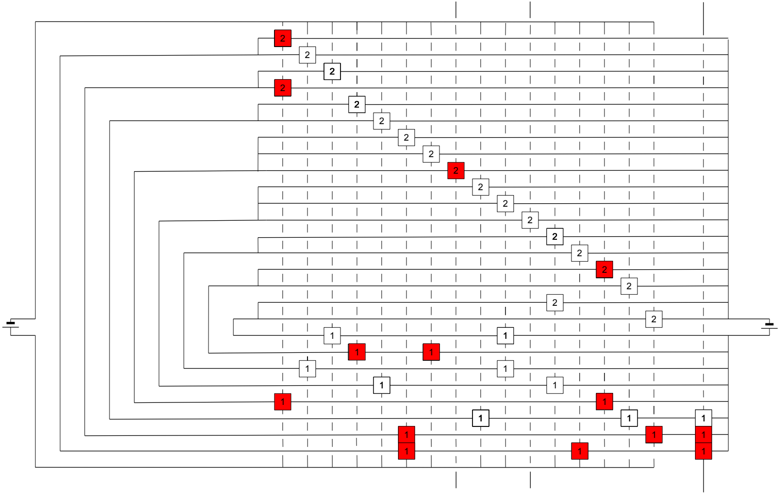
Exemplary Circuit Lattice



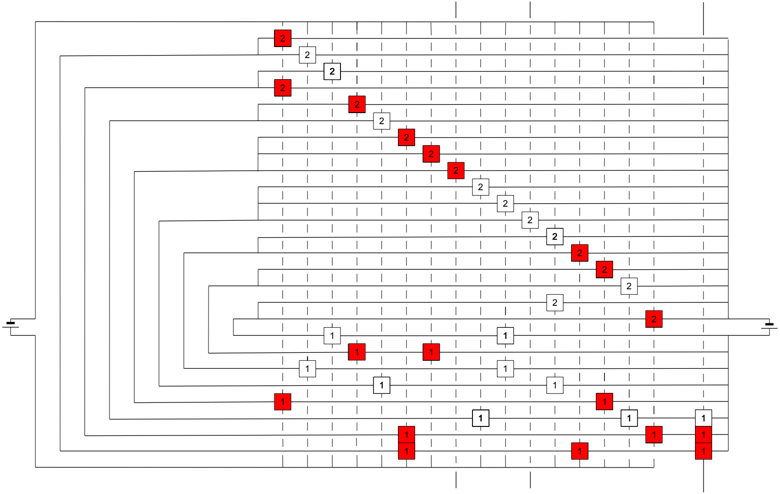
Exemplary Circuit Lattice



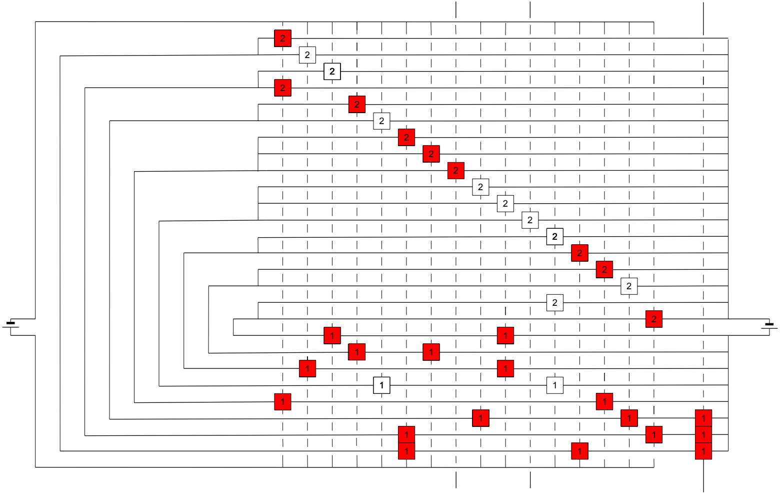
Exemplary Circuit Lattice



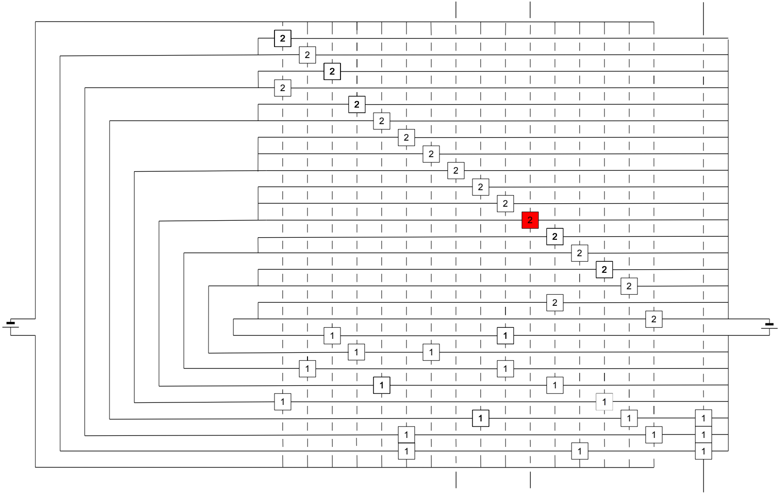
Exemplary Circuit Lattice



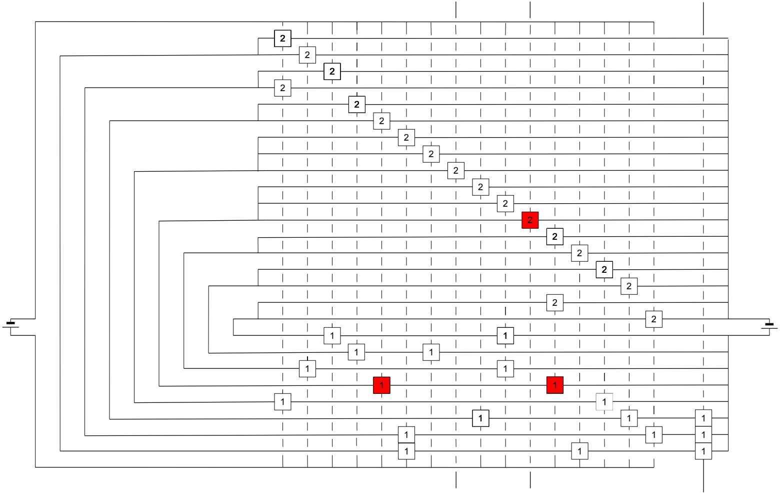
Exemplary Circuit Lattice



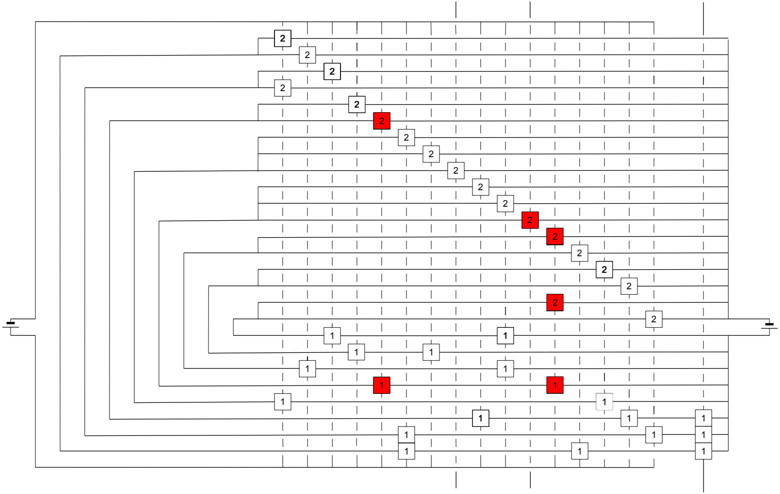
Exemplary Circuit Lattice. Introduce guess $x_4 = 1$



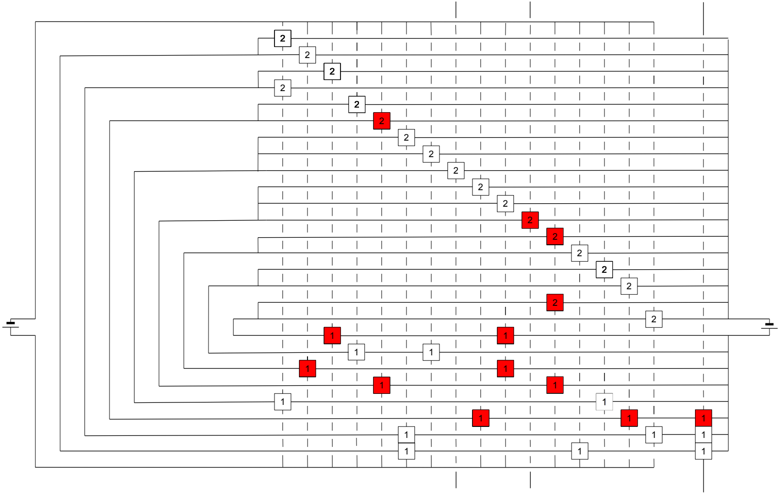
Exemplary Circuit Lattice



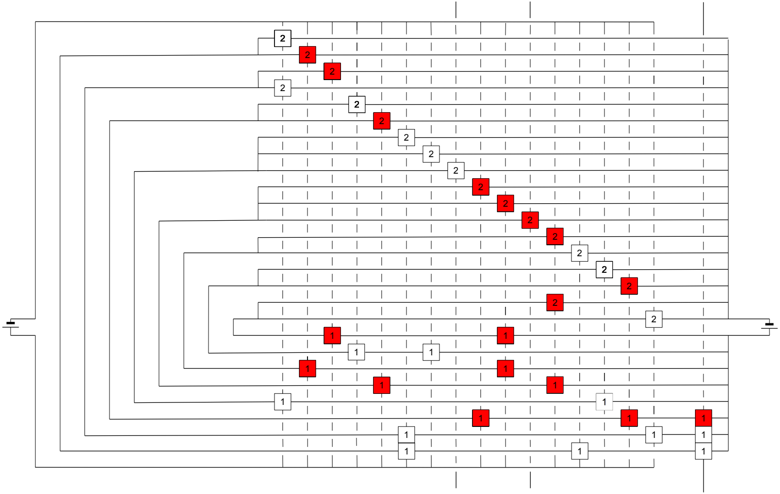
Exemplary Circuit Lattice



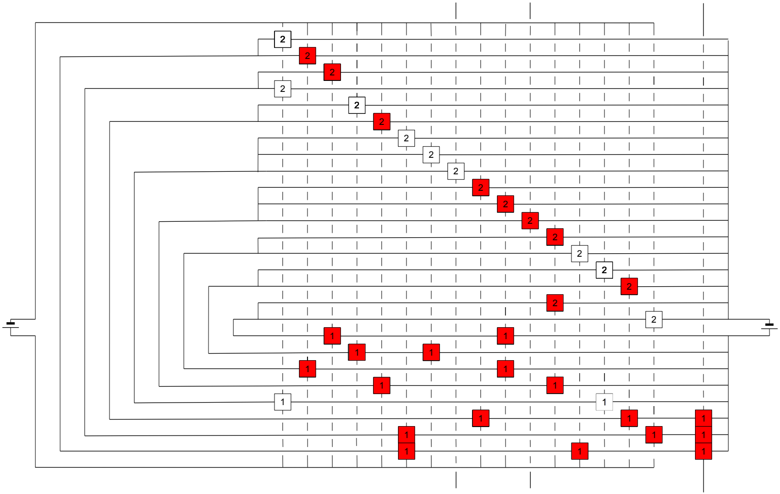
Exemplary Circuit Lattice



Exemplary Circuit Lattice



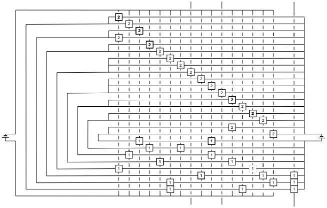
Exemplary Circuit Lattice



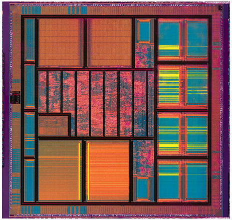
TripleDES system parameters

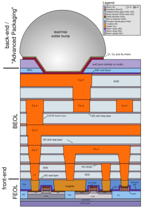
- 1712 variables, 384 equations
- 3929 maximal edges
- 71320 tuples
- 1.1×10^9 switches
- $480 = 2 \times 128 + 2 \times 112$ input contacts
- The device doesn't require synchronization

Circuit Lattice, as in WCC'09 topologically. Not much wiring intersection now. Implementable with two layers on a crystal.



Common Integrated Circuit, about 10 semiconductor layers





Implement on Modern Semiconductor Crystals for brute force?

- Transistor works as a switch
- 1.7×10^9 transistors on Dual-Core Itanium2 processor
- Circuit Lattice speed $\leq 2\times$(number of rounds) transistor turns
- $2 \times 48 + 2$ turns for TripleDES
- One transistor turn, say 100GHz(1000GHz reported)
- 1GHz key-rejecting rate when using for brute force
- Reported(2006) 0.13GHz per chip with implementing encryption

Conclusions

- WCC'09 design was improved
- Equation solving is shown as voltage expansion through a lattice of switches
- Our approach seems more flexible than implementing encryption as enables handling any equation system representing cipher
- Applications to DES, TripleDES are discussed