Faithful Representations of Graphs by Islands in the Extended Grid

Michael D. Coury Pavol Hell Jan Kratochvíl
Tomáš Vyskočil

Department of Applied Mathematics
and Institute for Theoretical Computer Science, Charles University

Grid

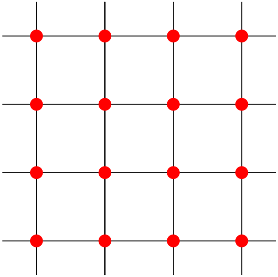
- The set of points in plane with integral coordinates.

Grid

- The set of points in plane with integral coordinates.
- Two distinct points $A = (X_A, Y_A)$ and $B = (X_B, Y_B)$ are *adjacent*, if $X_A = X_B|$ and $|Y_A - Y_B| \leq 1$ or $Y_A = Y_B$ and $|X_A - X_B| \leq 1$.

Grid

- The set of points in plane with integral coordinates.
- Two distinct points $A = (X_A, Y_A)$ and $B = (X_B, Y_B)$ are *adjacent*, if $X_A = X_B|$ and $|Y_A - Y_B| \leq 1$ or $Y_A = Y_B$ and $|X_A - X_B| \leq 1$.

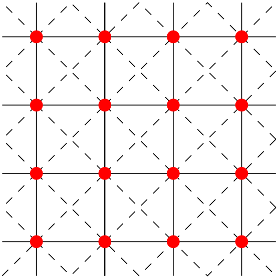


Extended grid

- Like a grid but we add diagonal edges.

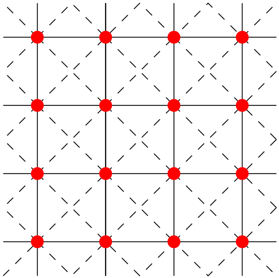
Extended grid

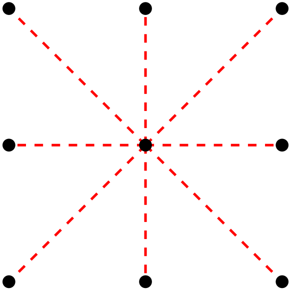
- Like a grid but we add diagonal edges.



Extended grid

- Like a grid but we add diagonal edges.



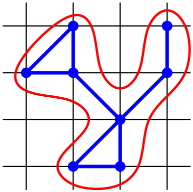


Island

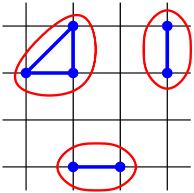
An *island* is a set of points in the extended grid which induces a connected subgraph.

Island

An *island* is a set of points in the extended grid which induces a connected subgraph.



connected, forms island



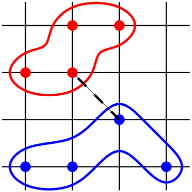
not connected

Island adjacency

Two islands i, j are adjacent if there is a pair of points $P \in i, Q \in j$, that are adjacent in the extended grid.

Island adjacency

Two islands i, j are adjacent if there is a pair of points $P \in i, Q \in j$, that are adjacent in the extended grid.

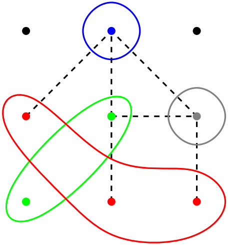


Faithful representation

- A *faithful representation* of a graph G by islands is a set $\mathcal{I}$ of vertex disjoint islands in the extended grid, such that the adjacency graph of islands of $\mathcal{I}$ is isomorphic to G.

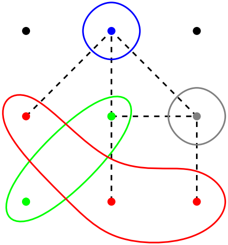
Faithful representation

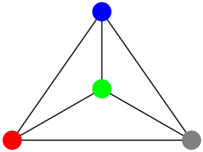
- A *faithful representation* of a graph G by islands is a set $\mathcal{I}$ of vertex disjoint islands in the extended grid, such that the adjacency graph of islands of $\mathcal{I}$ is isomorphic to G.



Faithful representation

- A *faithful representation* of a graph G by islands is a set $\mathcal{I}$ of vertex disjoint islands in the extended grid, such that the adjacency graph of islands of $\mathcal{I}$ is isomorphic to G.





Motivation

- Motivation for this structure came from adiabatic quantum computation (AQC) where vertices of islands are qubits and edges are couplers.

Motivation

- Motivation for this structure came from adiabatic quantum computation (AQC) where vertices of islands are qubits and edges are couplers.

Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation

Dorit Aharonov*
School of Computer Science and Engineering,
Hebrew University, Israel

Wim van Dam
Department of Physics,
MIT, Cambridge, MA

Julia Kempe†
CNRS-LRI UMR 8623,
Université de Paris-Sud,
Orsay, France

Zeph Landau
Department of Mathematics,
City College of New York

Seth Lloyd
Department of Mechanical Engineering,
MIT, Cambridge, MA

Oded Regev
Computer Science Department,
Tel-Aviv University, Israel

Abstract

The model of adiabatic quantum computation has recently attracted attention in the physics and computer science communities, but its exact computational power has been unknown. We settle this question and describe an efficient adiabatic simulation of any given quantum algorithm. This implies that the adiabatic computation model and the standard quantum circuit model are polynomially equivalent. We also describe an extension of this result with implications to physical implementations of adiabatic computation. We believe that our result highlights the potential importance of the adiabatic computation model in the design of quantum algorithms and in their experimental realization.

1. Introduction

The study of adiabatic quantum computation was initiated several years ago by Farhi, Goldstone, Gutmann and Sipser [9], who suggested a novel quantum algorithm for solving classical optimization problems such as SATISFIABILITY (SAT). Their algorithm is based on a celebrated theorem in quantum mechanics known as 'the adiabatic theorem' [14, 19]. Although its exact worst-case behavior is not known, several simulations (e.g., [8]) on random instances of up to 20 quantum bits led to various optimistic speculations. The bad news is that there is now mounting evidence [4, 5, 23] that the algorithm of [9] takes exponential time in the worst-case for NP-complete problems. Nevertheless, adiabatic computations were since shown to be promising in other less ambitious directions: they exhibit inherent robustness against certain types of quantum errors [3], and they possess several interesting algorithmic capabilities, as we will soon review.

We briefly describe the adiabatic computation model (a more precise description appears in Section 2.1). A computation in this model is specified by two Hamiltonians named H_{init} and H_{final} (a Hamiltonian is simply a Hermitian matrix). The ground state (eigenvector with smallest eigenvalue) of H_{init} is required to be an easy to prepare state, such as a tensor product state. The output of the adiabatic computation is the lowest eigenvector of the final Hamiltonian H_{final}. Hence, we choose an H_{final} whose ground state represents the solution to our problem. We require the Hamiltonians to be *local*, i.e., to only involve interactions between a constant number of particles (this is analogous to allowing gates operating on a constant number of qubits in the standard model). This, in particular, makes sure that the Hamiltonians have a short classical description (by simply listing the matrix entries of each local term). The running time of the adiabatic computation is determined by the minimal spectral gap[1] of all the Hamiltonians on the straight line connecting H_{init} and H_{final}: $H(s) = (1-s)H_{\text{init}} + sH_{\text{final}}$ for $s \in [0, 1]$. More precisely, the adiabatic computation is polynomial time if this minimal spectral gap is at least inverse polynomial.

The motivation for the above definition is physical. The Hamiltonian operator corresponds to the energy of the quantum system, and for it to be physically realis-

* Also Computer Science Division, UC Berkeley, CA 94720
† Also Computer Science Division and Department of Chemistry, UC Berkeley, CA 94720

1 The spectral gap is the difference between the lowest and second lowest eigenvalue.

Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS'04)
0272-5428/04 $20.00 © 2004 IEEE

IEEE COMPUTER SOCIETY

ISLANDS

- The set of graphs which have faithful representations by islands will be denoted ISLAND.

ISLANDS

- The set of graphs which have faithful representations by islands will be denoted ISLAND.
- An island with at most k vertices will be called a k-island.

ISLANDS

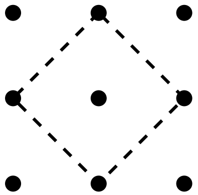
- The set of graphs which have faithful representations by islands will be denoted ISLAND.
- An island with at most k vertices will be called a k-island.
- The set of graphs which have faithful representations by k-islands will be denoted k-ISLAND

Connections

- ISLAND and k-ISLAND graphs are intersection graphs of connected regions in plane.

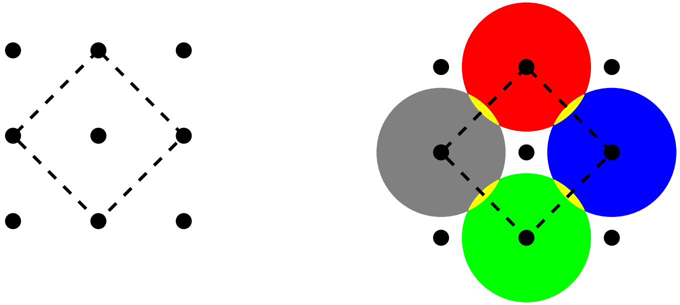
Connections

- ISLAND and k-ISLAND graphs are intersection graphs of connected regions in plane.



Connections

- ISLAND and k-ISLAND graphs are intersection graphs of connected regions in plane.

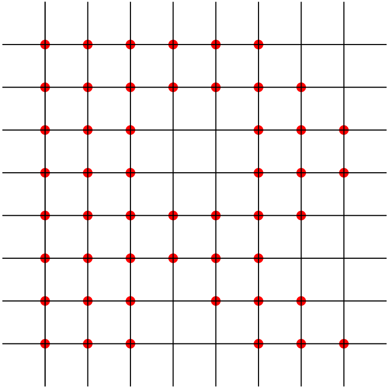


Induce subgraph of extended grid

- The graphs from 1-ISLAND are exactly induced subgraphs of extended grid.

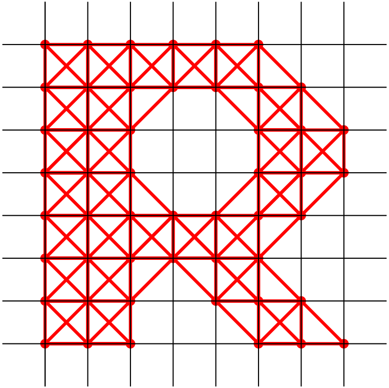
Induce subgraph of extended grid

- The graphs from 1-ISLAND are exactly induced subgraphs of extended grid.



Induce subgraph of extended grid

- The graphs from 1-ISLAND are exactly induced subgraphs of extended grid.

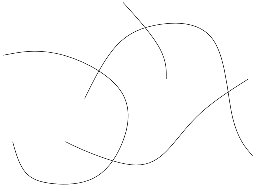


String graphs

- STRING graphs are intersection graphs of curves in plane.

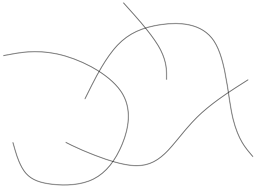
String graphs

- STRING graphs are intersection graphs of curves in plane.



String graphs

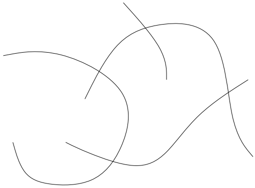
- STRING graphs are intersection graphs of curves in plane.



- STRING graphs have been introduced independently by Benzer(1959) and Sinden(1966).

String graphs

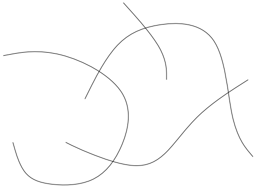
- STRING graphs are intersection graphs of curves in plane.



- STRING graphs have been introduced independently by Benzer(1959) and Sinden(1966).
- Ehrlich, Even and Tarjan (1976) showed computing the chromatic number of string graphs to be NP-hard.

String graphs

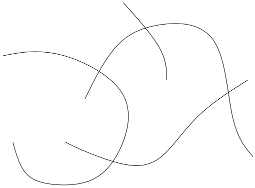
- STRING graphs are intersection graphs of curves in plane.



- STRING graphs have been introduced independently by Benzer(1959) and Sinden(1966).
- Ehrlich, Even and Tarjan (1976) showed computing the chromatic number of string graphs to be NP-hard.
- Kratochvíl (1991) showed string graph recognition to be NP-hard.

String graphs

- STRING graphs are intersection graphs of curves in plane.



- STRING graphs have been introduced independently by Benzer(1959) and Sinden(1966).
- Ehrlich, Even and Tarjan (1976) showed computing the chromatic number of string graphs to be NP-hard.
- Kratochvíl (1991) showed string graph recognition to be NP-hard.
- In that time it was not even known if this problem is decidable!!!

String graphs continued

- There are STRING graphs which need exponential number of crossings, it was proven by Kratochvíl and Matoušek (1991).

String graphs continued

- There are STRING graphs which need exponential number of crossings, it was proven by Kratochvíl and Matoušek (1991).
- The naive method to show that the recognition of STRING graphs is in NP is not working.

String graphs continued

- There are STRING graphs which need exponential number of crossings, it was proven by Kratochvíl and Matoušek (1991).
- The naive method to show that the recognition of STRING graphs is in NP is not working.
- Pach,Janos and Schaefer, Štefankovič independently proved in 2002 that recognition of STRING graphs is decidable.

String graphs continued

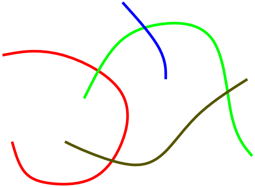
- There are STRING graphs which need exponential number of crossings, it was proven by Kratochvíl and Matoušek (1991).
- The naive method to show that the recognition of STRING graphs is in NP is not working.
- Pach,Janos and Schaefer, Štefankovič independently proved in 2002 that recognition of STRING graphs is decidable.
- Schaefer,Sedgwick and Štefankovič (2003) proved that recognizing of STRING graphs is NP-complete.

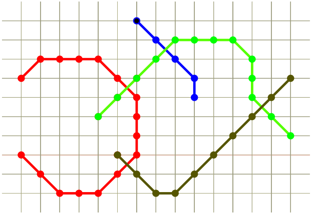
Consequence of STRING graphs

- STRING graphs are ISLAND graphs.

Consequence of STRING graphs

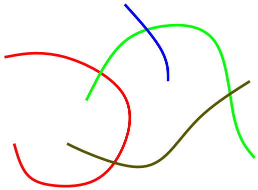
- STRING graphs are ISLAND graphs.

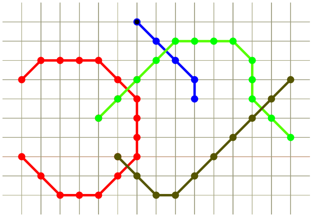




Consequence of STRING graphs

- STRING graphs are ISLAND graphs.

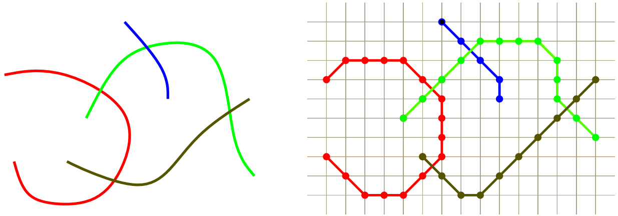




- There exist graphs which require exponential number of grid points.

Consequence of STRING graphs

- STRING graphs are ISLAND graphs.



- There exist graphs which require exponential number of grid points.
- The recognition of ISLAND graphs is NP-complete.

Problem

- The complexity of recognition of k-ISLAND graph.

Problem

- The complexity of recognition of k-ISLAND graph.
- We show that k-ISLAND is NP-complete for $k < 3$ and $k > 5$.

Our approach

- We use different method for the cases $k < 3$ and $k > 5$

Our approach

- We use different method for the cases $k < 3$ and $k > 5$
- For cases $k = 1, 2$ we made reduction from NAE-3-SAT and use so called "logic engine".

Our approach

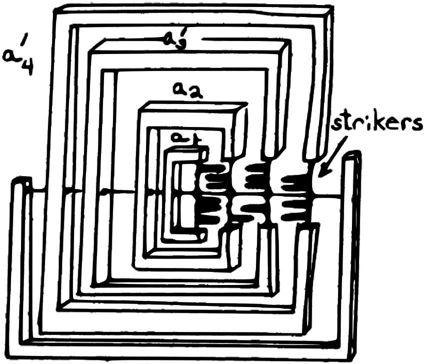
- We use different method for the cases $k < 3$ and $k > 5$
- For cases $k = 1, 2$ we made reduction from NAE-3-SAT and use so called "logic engine".
- For cases $k > 5$ we use reduction from PLANAR-3-CONNECTED-(3,4)-SAT.

Logic engine

- The logic engine was designed by Eades and Whitesides.

Logic engine

- The logic engine was designed by Eades and Whitesides.
- Very intuitive model of a standard reduction from the problem NAE-SAT



Small islands

Theorem

The problem 1-ISLAND is NP-complete.

Small islands

Theorem

The problem 1-ISLAND is NP-complete.

Theorem

The problem 2-ISLAND is NP-complete.

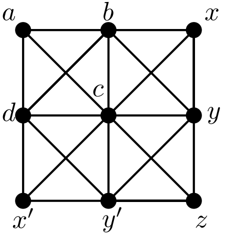
How to prove it

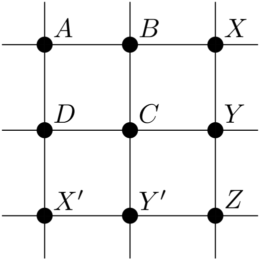
For "logic engine" construction are important to produce:

- rigid element (frames, shafts)
- flexible element (rod)

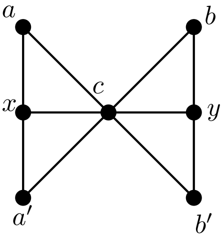
Case k = 1

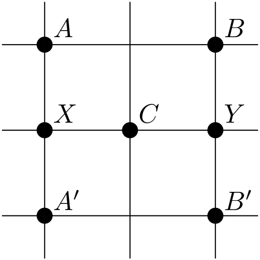
- rigid element





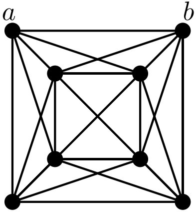
- flexible element

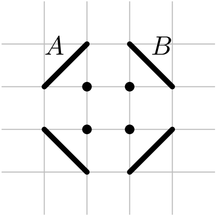




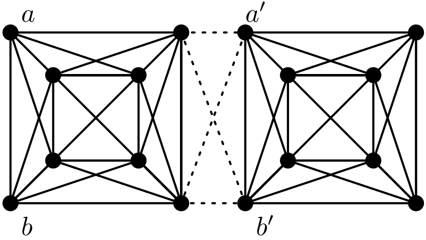
Case $k = 2$

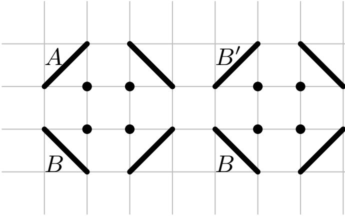
- rigid element





- flexible element



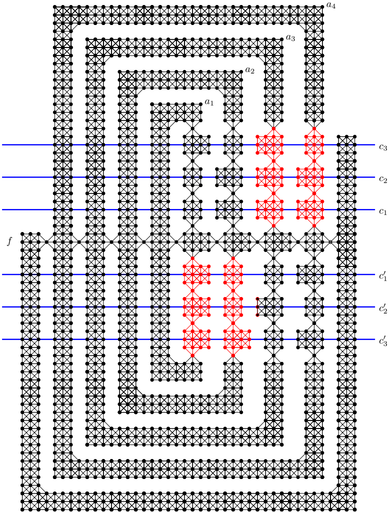


Example k = 1

$$(\neg X_1 \vee X_2 \vee \neg X_3) \wedge (\neg X_1 \vee X_2 \neg X_4) \vee (\neg X_2 \vee \neg X_3 \vee X_4)$$

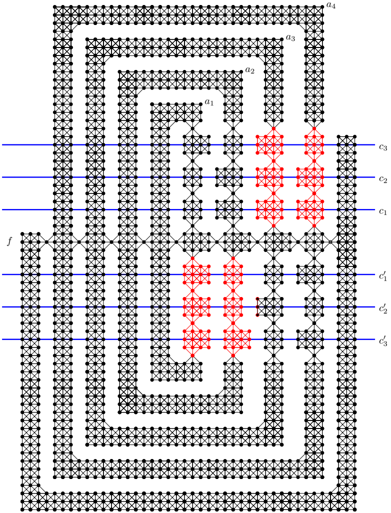
Example k = 1

$$(\neg X_1 \vee X_2 \vee \neg X_3) \wedge (\neg X_1 \vee X_2 \neg X_4) \vee (\neg X_2 \vee \neg X_3 \vee X_4)$$



Example k = 1

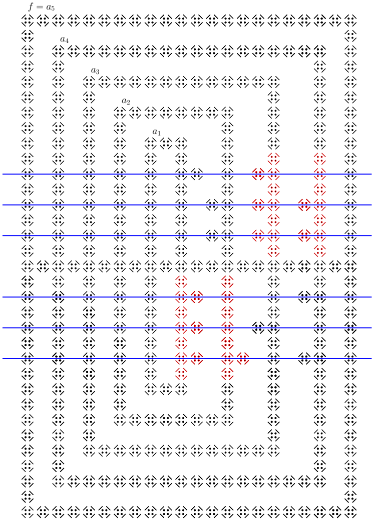
$$(\neg X_1 \vee X_2 \vee \neg X_3) \wedge (\neg X_1 \vee X_2 \neg X_4) \vee (\neg X_2 \vee \neg X_3 \vee X_4)$$



X_1 false
X_2 false
X_3 true
X_4 true

Example k = 2

$$(\neg X_1 \vee X_2 \vee \neg X_3) \wedge (\neg X_1 \vee X_2 \neg X_4) \vee (\neg X_2 \vee \neg X_3 \vee X_4)$$



X_1 false
X_2 false
X_3 true
X_4 true

PLANAR-3-CONNECTED-(3,4)-SAT

- A variant of satisfiability problem.

PLANAR-3-CONNECTED-(3,4)-SAT

- A variant of satisfiability problem.
- Input formula has exactly 3 distinct literals in each clause.

PLANAR-3-CONNECTED-(3,4)-SAT

- A variant of satisfiability problem.
- Input formula has exactly 3 distinct literals in each clause.
- Each variable occurs in at most 4 clauses

PLANAR-3-CONNECTED-(3,4)-SAT

- A variant of satisfiability problem.
- Input formula has exactly 3 distinct literals in each clause.
- Each variable occurs in at most 4 clauses
- Incidence graph of formula is vertex-3-connected and planar.

Reduction for large islands

- PLANAR-3-CONNECTED-(3,4)-SAT was introduced by by Kratochvíl. It can be used to show that STRING graphs are NP-hard.

Reduction for large islands

- PLANAR-3-CONNECTED-(3,4)-SAT was introduced by by Kratochvíl. It can be used to show that STRING graphs are NP-hard.
- We will use the same reduction for recognition of large islands.

Idea of the proof

- For given ϕ formula produce a graph H_ϕ with following properties:

Idea of the proof

- For given ϕ formula produce a graph H_ϕ with following properties:
  - If ϕ is satisfiable then $H_\phi \in 6-$ISLAND.

Idea of the proof

- For given ϕ formula produce a graph H_ϕ with following properties:
  - If ϕ is satisfiable then $H_\phi \in 6-$ISLAND.
  - If ϕ is not satisfiable then $H_\phi \notin$ STRING $=$ ISLAND.

The first step of construction

- We fix rectilinear drawing of incidence graph of ϕ.

The first step of construction

- We fix rectilinear drawing of incidence graph of ϕ.
- Clauses and variables are located in points of planar grid.

The first step of construction

- We fix rectilinear drawing of incidence graph of ϕ.
- Clauses and variables are located in points of planar grid.
- Edges piece-wise linear and following the grid lines.

The first step of construction continued

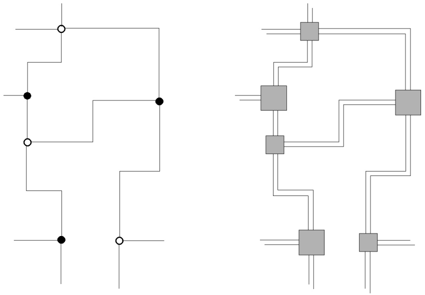
- Consider a refinement of grid so that the variable and clause are replaced by disjoint rectangles

The first step of construction continued

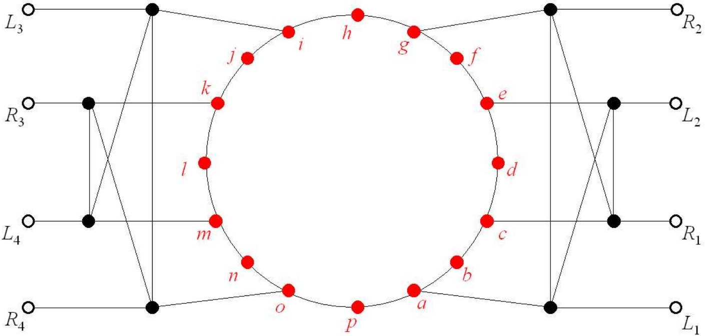
- Consider a refinement of grid so that the variable and clause are replaced by disjoint rectangles
- Edges replaced by pair of parallel paths.

The first step of construction continued

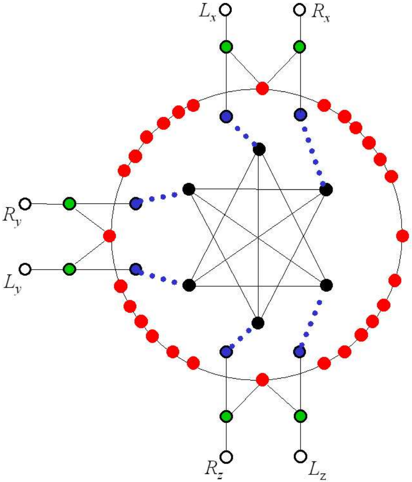
- Consider a refinement of grid so that the variable and clause are replaced by disjoint rectangles
- Edges replaced by pair of parallel paths.



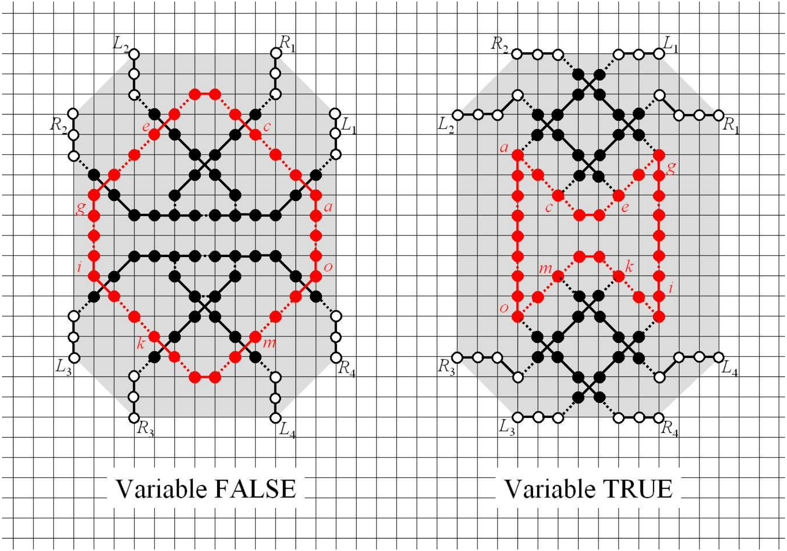
Variable gadget



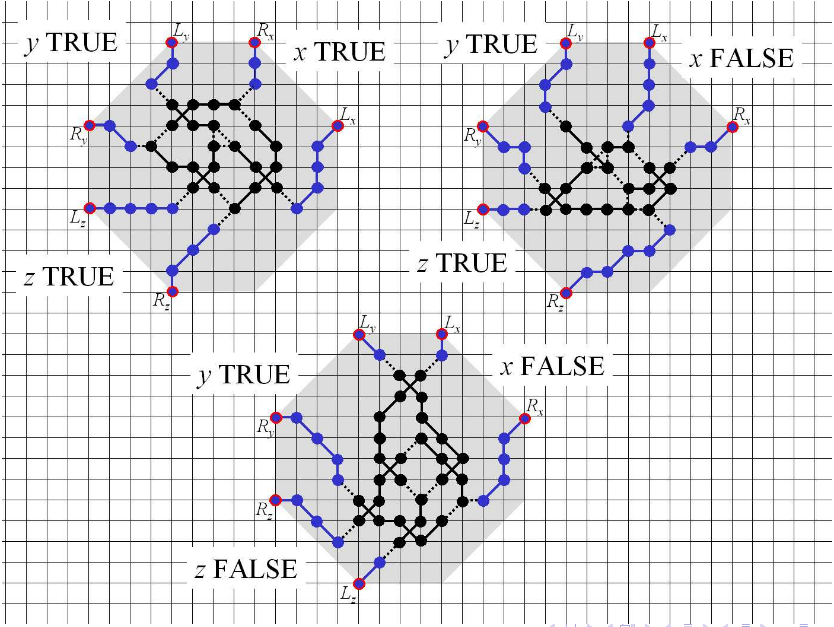
Clause gadget



Representations of the variable gadget



Representations of the clause gadget



The end

Thank you for your attention