Open Problems TO GO

Yury Lifshits

Caltech

http://yury.name

MIT, 30 November 2007

Open Problems TO GO:

- Short mathematical statement
- No background required
- Motivation (importance) is guaranteed

Open Problems TO GO:

- Short mathematical statement
- No background required
- Motivation (importance) is guaranteed

Today:

Three classic problems
Three problems from YL

1

Classic Problems

Mean payoff games
Semi-Thue systems
Ulam conjecture (graph reconstruction)

1.1. Rules of mean payoff games

Input for a **mean payoff game**:

- Weighted directed graph (integer weights)
- Graph does not contain simple cycles with zero sum
- Vertices are divided into disjoint sets A and B
- The starting vertex

Rules of Mean Payoff Games

- Two players: Alice and Bob
- Players move the token over arcs
- Game starts from the starting vertex and it is infinite
- Alice plays from vertices of A, Bob from these of B
- Alice wins if the sum of already passed arcs goes to $+infty$
- Bob wins if the sum of already passed arcs goes to $-infty$

Computational Problem

Given a game graph with an A, B decomposition and a starting vertex to determine the winner (and find the winning strategy)

MPG is Very Challenging

MPG Problem belongs to NP∩co-NP
Direct applications in μ-calculus verification

MPG is Very Challenging

MPG Problem belongs to NP∩co-NP
Direct applications in μ-calculus verification

Known algorithms:

- Randomized algorithm $\mathcal{O}^*(2^{\sqrt{n}})$ expected time
- Deterministic algorithm $\mathcal{O}^*(2^n)$ time

References

Y. Lifshits, D. Pavlov

Potential Theory for Mean Payoff Games

Journal of Mathematical Sciences, 2007

http://yury.name/papers/lifshits2006fast.pdf

M. Jurdziński, M. Paterson, U. Zwick

A deterministic subexponential algorithm for solving parity games

SODA'06

http://www.dcs.warwick.ac.uk/~mju/Papers/JPZ07-manuscript.pdf

H. Björklund, S. Vorobyov

A combinatorial strongly subexponential strategy improvement algorithm for mean payoff games

Discrete Applied Mathematics, 2007

http://portal.acm.org/citation.cfm?id=1222484

Ulam Conjecture

A vertex-deleted subgraph of a graph G is a subgraph $G - v$ obtained by deleting a vertex v and its incident edges. The deck of a graph G is the family of (unlabelled) vertex-deleted subgraphs of G; these are the cards of the deck. A reconstruction of a graph G is a graph H with the same deck as G. A graph G is reconstructible if every reconstruction of G is isomorphic to G.

Ulam Conjecture

A vertex-deleted subgraph of a graph G is a subgraph $G - v$ obtained by deleting a vertex v and its incident edges. The deck of a graph G is the family of (unlabelled) vertex-deleted subgraphs of G; these are the cards of the deck. A reconstruction of a graph G is a graph H with the same deck as G. A graph G is reconstructible if every reconstruction of G is isomorphic to G.

Conjecture: every graph with at least three vertices is reconstructible

Reference

J.A. Bondy

A graph reconstructor’s manual

Surveys in Combinatorics, 1991

http://www.ecp6.jussieu.fr/pageperso/bondy/research/papers/recon.ps

Semi-Thue Systems

Rewriting (α, β) rule allows to rewrite any $u\alpha v$ in $u\beta v$

Semi-Thue Systems

Rewriting (α, β) rule allows to rewrite any $u\alpha v$ in $u\beta v$

Word problem: Given system of rules and two words w_1 and w_2 to decide whether one can be obtained from another by a sequence of such rules?

Challenge

There is a system with three rules such that word problem is undecidable

Challenge

There is a system with three rules such that word problem is undecidable

Is word problem decidable or not for systems of one (two) rules?

Reference

Y. Matiyasevich and G. Senizerguez

Decision Problems for Semi-Thue Systems with a Few Rules

LICS'96

http://dept-info.labri.u-bordeaux.fr/~ges/termination.ps

2

Open Problems from YL

Compressed Arithmetic

Input: Two grammars of size n, m generating binary strings P and Q of the same length

Task: Compute a close-to-minimal grammar generating "bitwise OR between P and Q"

Compressed Arithmetic

Input: Two grammars of size n, m generating binary strings P and Q of the same length

Task: Compute a close-to-minimal grammar generating "bitwise OR between P and Q"

Can we do it in time $poly(n + m + output)$?

References

Yury Lifshits

Processing Compressed Texts: A Tractability Border

CPM'07

http://yury.name/papers/lifshits2007processing.pdf

Yury Lifshits and Markus Lohrey

Querying and Embedding Compressed Texts

MFCS'06

http://yury.name/papers/lifshits2006querying.pdf

Patrick Cégielski, Irène Guessarian, Yury Lifshits and Yuri Matiyasevich

Window Subsequence Problems for Compressed Texts

CSR'06

http://yury.name/papers/cegielski2006window.pdf

Impossibility of Preprocessing

Input

Circuits $C_1 \ldots, C_n$
of size $poly(m)$ with input size m

Query task

Given string y of length m to answer whether $\exists i : C_i(y) = yes$

Impossibility of Preprocessing

Input

Circuits $C_1 \ldots, C_n$
of size $poly(m)$ with input size m

Query task

Given string y of length m to answer
whether $\exists i : C_i(y) = yes$

Constraints:

$poly(n, m)$ preprocessing
$poly(\log n, m)$ search

Impossibility of Preprocessing

Input

Circuits $C_1 \ldots, C_n$
of size $poly(m)$ with input size m

Query task

Given string y of length m to answer
whether $\exists i : C_i(y) = yes$

Constraints:

$poly(n, m)$ preprocessing
$poly(\log n, m)$ search

Open problem: Is there a solution within given constraints?

Dual Problem

Input

Strings $x_1 \ldots, x_n$ of length m,

Query task

Given circuit C of size $polym$ with input length m to answer whether $\exists i : C(x_i) = yes$

Dual Problem

Input

Strings $x_1 \ldots, x_n$ of length m,

Query task

Given circuit C of size $polym$ with input length m to answer whether $\exists i : C(x_i) = yes$

Constraints:

$poly(n, m)$ preprocessing

$poly(\log n, m)$ search

Dual Problem

Input

Strings $x_1 \ldots, x_n$ of length m,

Query task

Given circuit C of size $polym$ with input length m to answer whether $\exists i : C(x_i) = yes$

Constraints:

$poly(n, m)$ preprocessing

$poly(\log n, m)$ search

Open problem: Is there a solution within given constraints?

Reference

Yury Lifshits

Algorithms for Nearest Neighbors: Classic Ideas, New Ideas

Talk at University of Toronto

MP3 recording

http://yury.name/talks/toronto-talk.pdf

Positive Subgraph

Input

$n \times n$ bipartite graph (pretty sparse)

Weights on edges

Positive Subgraph

Input

$n \times n$ bipartite graph (pretty sparse)
Weights on edges

Task

Find a $k \times k$ subgraph
with maximal average edge weight

Positive Subgraph

Input

$n \times n$ bipartite graph (pretty sparse)

Weights on edges

Task

Find a $k \times k$ subgraph

with maximal average edge weight

Polynomial approximate algorithm?

Reference

Y. Lifshits and D. Nowotka

Estimation of the click volume by large scale regression analysis

CSR'07

http://yury.name/papers/lifshits2007click.pdf

http://www.netflixprize.com

Voting

Which problem you like the most?

- Mean Payoff Games
- Ulam Conjecture
- Semi-Thue Systems
- Compressed Arithmetics
- Impossibility of Preprocessing
- Positive Subgraph

http://yury.name

Thanks for your attention!
Questions?