Computational Complexity of Locally Injective Homomorphisms to Weight Graphs: A Full Classification of Simple Weights

Ondřej Bílka, Jiří Fiala, Jan Kratochvíl, Bernard Lidický, Marek Tesař

Charles University

16.2.2010 - ATCAGC 2010
Definitions

A *weight graph* is a connected multi-graph $G$ with two vertices $A, B$ of degree at least three and all other of degree two. Moreover, $G - A$ and $G - B$ contain a cycle.

A weight graph is *simple* if $\text{deg}(A) = \text{deg}(B) = 3$. Also known as dumbbell or barbell.

$W(1,1,1)$

$W(1,2,3)$
Simple weight graphs

Name: $H$-LIHOM
Parameter: graph $H$
Input: graph $G$
Question: Does exist a locally injective homomorphism $f : V(G) \rightarrow V(H)$?

Theorem
If $H$ is a bipartite simple weight graph ($A, B$ in different parts), then $H$-LIHOM is solvable in polynomial time.

Theorem
If $H$ is a non-bipartite simple weight graph, then $H$-LIHOM is NP-complete.
Simple overview of the polynomial algorithm

simple bipartite weight graph $H$
a graph $G$ as input

• fix a bipartition of $G$
• compute possible mappings of paths of 2-vertices in $G$
• replace paths by gadgets and obtain $G'$
• compute a "matching" in $G'$
• construct a mapping $G \to H$ according to the matching
Simple bipartite case (polynomial time)

Let $H = W(a, b, c)$ be a bipartite weight graph, $G$ an input graph.

Consider the weight graph $W(2, 2, 1)$.

Fix a bipartition of 3-vertices in $G$ (2 possibilities).

For every path, decide possible mappings on ends:
(a-c, b-c, a-a, a-b, c-a, c-b, c-c)
Simple bipartite case (polynomial time)

$H = W(a, b, c)$ be a bipartite weight graph

fix a bipartition of 3-vertices of $G$

for every path decide possible mappings on ends

(a-c, b-c, a-a, a-b, c-a, c-b, c-c)
Simple bipartite case (polynomial time)

We want to determine the mapping of $c$ around every 3-vertex one at every vertex - like matching

**Lemma**

Let $G$ be a graph and a mapping $f : V(G) \rightarrow I$, where $I$ is a set of intervals. A subgraph $G'$ of $G$ such that $\deg_{G'}(v) \in f(v)$ for all $v \in V(G)$ can be found in polynomial time.

replace paths by gadgets (includes $f$) - $G'$
assign $f(v) = 1$ for all 3-vertices of $G'$
LIH\textsubscript{om} to simple weight graphs

replace paths \( \begin{array}{c}
\bullet \quad \bullet \\
1 & 1
\end{array} \) by gadgets (includes \( f \))

\[
\begin{array}{ccc}
a - a & a - c & a - c \\
1 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - c & c - a & a - c \\
1 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - a & a - c & a - a \\
0, 1 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - a & a - c & a - a \\
0, 1 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - a & a - c & a - a \\
0, 1, 1 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - c & c - a & c - c \\
1 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - c & c - a & c - c \\
1, 2 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - a & a - c & a - a \\
1, 2 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - a & a - c & a - a \\
1, 2 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - a & a - c & a - a \\
1, 2 & 1 & 1
\end{array}
\]

\[
\begin{array}{ccc}
a - a & a - c & a - a \\
1, 2 & 1 & 1
\end{array}
\]
Simple bipartite case (polynomial time) - Example

Replacing path by a gadget and getting a factor.
Simple bipartite case (polynomial time) - Overview

- fix a bipartition of $G$
- compute possible mappings of paths in $G$
- replace paths by gadgets ($G'$)
- compute $f$-factor in ($G'$)
- map $c$ in $G$ according to edges in $f$-factor of $G'$
- map $a, b$ around $A$ and $B$
- map remaining vertices
LIHOM to simple weight graphs

Simple non-bipartite case (NP-complete)

$G = W(a, b, c)$, assume $\text{GCD}(a, b, c) = 1$ and $\text{GCD}(a, b, 2c) = 1$

**Lemma**

*Let $a, b, c \in \mathbb{N}$ such that $\text{GCD}(a, b, 2c) = 1$. Then exist $x, y, z \in \mathbb{N}$ such that $ax = by + 2cz + c$ and $x, y \geq z$. \qed*

Let $k \in \mathbb{N}$ be the smallest such that exists a mapping of path of length $k$:

$A \sim k \sim X$

$A \sim k - X$

$B \sim k - X$

$B \sim k \sim X$

$X \in \{A, B\}$
Simple non-bipartite case (NP-complete)

decide reduction according to what the path of length $k$ allows

$$A \sim k \sim X$$  $2$-$\text{IN-3-SAT}$

$$B \sim k - X$$

$$A \sim k \sim X$$  $A \sim k \sim Y$  $\text{NAEQ-SAT}$

$$B \sim k - X$$  $B \sim k - Y$

$$A \sim k \sim X$$  $A \sim k - Y$  $\text{NAEQ-SAT}$

$$B \sim k - X$$  $B \sim k \sim Y$

$Y \neq X, X, Y \in \{A, B\}$
LIHOM to simple weight graphs

Variable gadget - 2-in-3-SAT

\[
\begin{align*}
X & \quad A \\
& \quad B \\
& \quad k \\
X & \quad X
\end{align*}
\]
LIHOM to simple weight graphs

Variable gadget - 2-IN-3-SAT
Variable gadget - 2-IN-3-SAT

LIHOM to simple weight graphs
LIHOM to simple weight graphs

Variable gadget - 2-IN-3-SAT
Variable gadget - 2-IN-3-SAT

LIHOM to simple weight graphs
Variable gadget - 2-IN-3-SAT
Variable gadget - 2-IN-3-SAT
Use three paths of length $k$ to form a clause.
Other partial results

- symmetric weight graph, which contains $W(a, a, a)$ (NP-complete)
- generalization of the polynomial time algorithm
- some small cases for weight graphs (NP-complete)
- some small cases for graphs

\[ \text{Graphs} \]