An introduction to the independent set reconfiguration problem (ISReconf)

Duc A. HOANG

Ph.D. Student (D1) @ Uehara Lab
School of Information Science, JAIST
hoanganhduc@jaist.ac.jp

November 11, 2015

The Reconfiguration Problem

- INSTANCE:
  1. Collection of configurations.
  2. Allowed transformation rule(s).
- QUESTION: For any two configurations A, B from the given collection, can A be transformed to B using the given rule(s)?

A classic example is the so-called 15-*puzzle*.

- INSTANCE:
  1. **Configuration:** 15 titles numbered from 1 to 15 are arranged on an 4×4 grid, leaving one empty square.
  2. **Rule:** A title can move to the empty square if it is above, or below, or on the left, or on the right of that empty square.
- QUESTION: Starting from any *source* configuration, can we reach the *target* configuration, where all numbers are in order (right-hand picture in the figure below)?

8	15	13	3
10		14	7
5	1	2	4
9	12	11	6

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

What is ISRECONF?

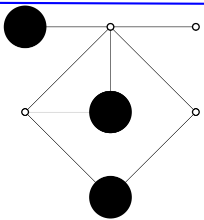

Figure: An independent set of a graph. Independent vertices are marked with black tokens.

- INSTANCE:
  1. A graph $G = (V, E)$.
  2. Two independent sets I_b, I_r.
  3. "Reconfiguration" rules: TS, TJ, TAR.
- QUESTION: Can I_b be transformed to I_r using one of the given rules such that all intermediate sets are independent?

1. TS: Slide tokens along edges.
2. TJ: A token "jumps" from one vertex to another.
3. TAR: Add or Remove tokens.

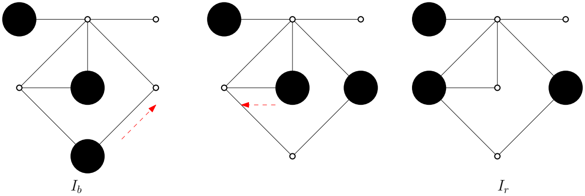

Figure: A YES-instance under TS rule.

Why study ISRECONF?

1. "Independent set" is an important object in computational complexity and graph theory.

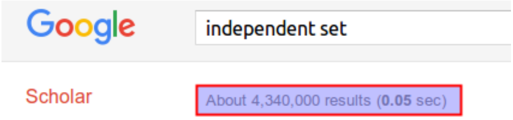

Figure: A search on Google Scholar with keyword "independent set".

2. ISRECONF (under any of the three given rules) is PSPACE-complete for general graphs, perfect graphs, and even planar graphs. Several PSPACE-hardness results are shown using reduction from ISRECONF and its variants.

Marcin Kamiński, Paul Medvedev, and Martin Milanič (2012). "Complexity of independent set reconfigurability problems". In: *Theoretical Computer Science* 439, pp. 9–15

Jan van den Heuvel (2013). "The complexity of change". In: *Surveys in Combinatorics 2013*. Ed. by Simon R. Blackburn, Stefanie Gerke, and Mark Wildon. Cambridge University Press, pp. 127–160

Recent Results

Graph	Rule(s)	Complexity	Paper(s)
planar	TS, TJ, TAR	PSPACE-complete	Robert A. Hearn and Erik D. Demaine (2005). "PSPACE-completeness of Sliding-block Puzzles and Other Problems Through the Nondeterministic Constraint Logic Model of Computation". In: *Theoretical Computer Science* 343.1-2, pp. 72–96
general line	TS, TJ, TAR TJ, TAR	PSPACE-complete P	Takehiro Ito, Erik D. Demaine, et al. (2011). "On the complexity of reconfiguration problems". In: *Theoretical Computer Science* 412.12-14, pp. 1054–1065
perfect even-hole-free cograph (P_4-free)	TS, TJ, TAR TJ, TAR TS	PSPACE-complete P P	Marcin Kamiński, Paul Medvedev, and Martin Milanič (2012). "Complexity of independent set reconfigurability problems". In: *Theoretical Computer Science* 439, pp. 9–15
cograph (P_4-free)	TJ, TAR	P	Paul Bonsma (2014). "Independent Set Reconfiguration in Cographs". In: *WG 2014*. Ed. by Dieter Kratsch and Ioan Todinca. Vol. 8747. LNCS. Springer, pp. 105–116
bounded bandwidth	TJ, TAR	PSPACE-complete	Marcin Wrochna (2014). "Reconfiguration in bounded bandwidth and treedepth". In: *arXiv preprints* arXiv:1405.0847
claw-free	TS, TJ	P	Paul Bonsma, Marcin Kamiński, and Marcin Wrochna (2014). "Reconfiguring Independent Sets in Claw-Free Graphs". In: *SWAT 2014*. Ed. by R. Ravi and IngeLi Gørtz. Vol. 8503. LNCS. Springer, pp. 86–97
tree	TS	P	Erik D. Demaine et al. (2015). "Linear-time algorithm for sliding tokens on trees". In: *Theoretical Computer Science* 600, pp. 132–142
bipartite permutation bipartite distance-hereditary	TS TS	P P	Eli Fox-Epstein et al. "Sliding Token on Bipartite Permutation Graphs". In: ISAAC 2015, Nagoya, Japan, Dec. 9–11. (To be appeared)

Table: Recent results on studying ISRECONF.

Useful tools for studying ISReconf

In the process of studying reconfiguration problem and its variants (including ISReconf), the following tools are particularly useful.

1. Nondeterministic Constraint Logic (NCL): This framework is motivated from the so-called *sliding-block puzzle*, which is a generalization of the classic 15-puzzle. NCL is a general framework built for proving PSPACE-hardness which simply requires the construction of a couple of gadgets that can be connected together in a planar graph. More details can be found in
   Robert A. Hearn and Erik D. Demaine (2005). "PSPACE-completeness of Sliding-block Puzzles and Other Problems Through the Nondeterministic Constraint Logic Model of Computation". In: *Theoretical Computer Science* 343.1-2, pp. 72–96
2. Dynamic Programming: Roughly speaking, Dynamic Programming is a method for solving a complex problem by breaking it down into a collection of simpler subproblems. A general setting of Dynamic Programming method for solving reconfiguration problems was recently presented in
   Paul Bonsma and Daniel Paulusma (2015). "Using Contracted Solution Graphs for Solving Reconfiguration Problems". In: *arXiv preprints* arXiv:1509.06357

Clique Reconfiguration and ISReconf

- Instance:
  1. A graph $G = (V, E)$.
  2. Two cliques C_0, C_r.
  3. Reconfiguration rules: TS, TJ, TAR.
- Question: Can C_0 be transformed to C_r using one of the given rules such that all intermediate sets are clique?

1. TS: Slide a token in clique C to a vertex in $V \setminus C$ along an edge of G.
2. TJ: A token in clique C jump to a vertex in $V \setminus C$.
3. TAR: Add or Remove tokens.

Interestingly, in Clique Reconfiguration, the three rules TS, TJ and TAR are equivalent, while in ISReconf they are not. More details are in Takehiro Ito, Hirotaka Ono, and Yota Otachi (2015). "Reconfiguration of Cliques in a Graph". In: *TAMC 2015, Singapore, May 18-20, 2015*, pp. 212–223

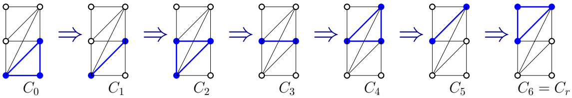

Figure: A sequence $\langle C_0, C_1, \ldots, C_6 \rangle$ of cliques in the same graph, where the vertices in cliques are depicted by large (blue) circles (tokens). (© Ito, Ono, and Otachi 2015)

Research Directions

Some interesting directions for ISRECONF study are as follows:

1. There are a long list of open problems: What is the complexity of ISRECONF for ... graph under ... rule?
2. Also, it is natural to ask questions about the *length of the reconfiguration sequence* (number of intermediate sets required to transform the source configuration to the target one). Recently, the ISRECONF problem for trees was shown to be in P (under any of the three rules), but the corresponding "shortest reconfiguration sequence" problem is still open?
3. Many researchers are interested in studying the connection between a decision problem (such as the famous INDEPENDENT SET problem) and its corresponding reconfiguration version (such as ISRECONF). Interestingly, for a general graph, INDEPENDENT SET is NP-complete, and ISRECONF is PSPACE-complete.