Sliding tokens on unicyclic graphs

Duc A. Hoang [1] Ryuhei Uehara [1]

[1] Japan Advanced Institute of Science and Technology
Asahidai 1-1, Nomi, Ishikawa 923-1292, Japan.
{hoanganhduc, uehara}@jaist.ac.jp

March 06, 2016

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*.

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	

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

Independent Set Reconfiguration (ISRECONF)

- INSTANCE:
  1. A graph $G = (V, E)$.

Independent Set Reconfiguration (ISReconf)

- Instance:
  1. A graph $G = (V, E)$.
  2. Two independent sets I, J.

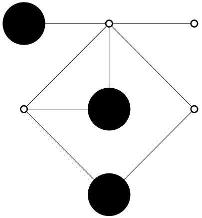


Figure: An independent set of a graph. Independent vertices are marked with black tokens.

Independent Set Reconfiguration (ISRECONF)

- INSTANCE:
  1. A graph $G = (V, E)$.
  2. Two independent sets I, J.
  3. "Reconfiguration" rules: TS, TJ, TAR.

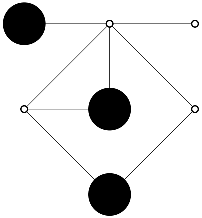


Figure: An independent set of a graph. Independent vertices are marked with black tokens.

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

Independent Set Reconfiguration (ISRECONF)

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

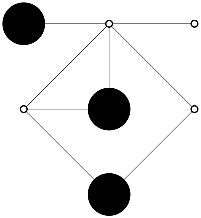


Figure: An independent set of a graph. Independent vertices are marked with black tokens.

1. TS: Slide tokens along edges (SLIDING TOKEN).
2. TJ: A token “jumps” from one vertex to another.
3. TAR: Add or Remove tokens.

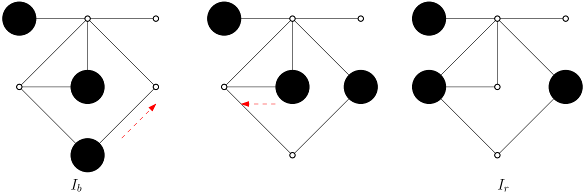
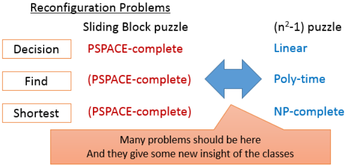


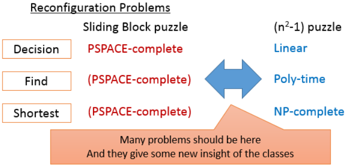
Figure: A YES-instance under TS rule.

Why study these problems?



(a) New insight of "complexity" inspired by games/puzzles (Picture © Ryuhei Uehara @ ICALP 2015).

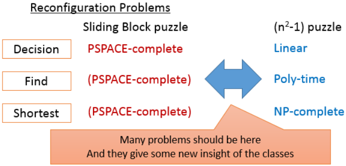
Why study these problems?



(a) New insight of "complexity" inspired by games/puzzles (Picture © Ryuhei Uehara @ ICALP 2015).

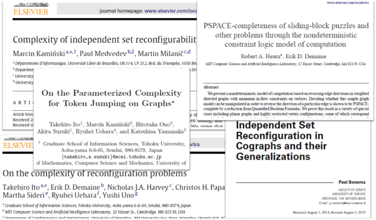
(b) CoRe 2015 (Sendai, Japan) - Recent results + future directions of combinatorial reconfiguration.

Why study these problems?



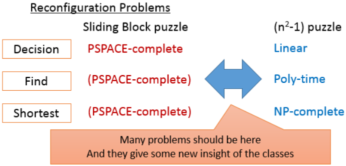
(a) New insight of "complexity" inspired by games/puzzles (Picture © Ryuhei Uehara @ ICALP 2015).

(b) CoRe 2015 (Sendai, Japan) - Recent results + future directions of combinatorial reconfiguration.



(c) Several PSAPCE-hardness results were shown using reduction from ISRECONF.

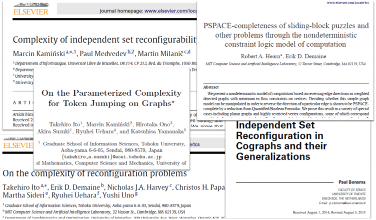
Why study these problems?



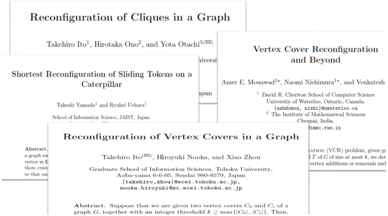
2015
Combinatorial Reconfiguration

(a) New insight of "complexity" inspired by games/puzzles (Picture © Ryuhei Uehara @ ICALP 2015).

(b) CoRe 2015 (Sendai, Japan) - Recent results + future directions of combinatorial reconfiguration.



(c) Several PSAPCE-hardness results were shown using reduction from ISRECONF.



(d) Recently, several problems related to ISRECONF have been extensively studied.

Sliding tokens on unicyclic graphs

Linear-time algorithm for sliding tokens on trees

Erik D. Demaine[a], Martin L. Demaine[a], Eli Fox-Epstein[b], Duc A. Hoang[c,*], Takehiro Ito[d,e], Hirotaka Ono[f], Yota Otachi[c], Ryuhei Uehara[c], Takeshi Yamada[c]

[a] MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar St., Cambridge, MA 02139, USA
[b] Department of Computer Science, Brown University, 115 Waterman Street, Providence, RI 02912-1910, USA
[c] School of Information Science, Japan Advanced Institute of Science and Technology, Asahidai 1-1, Nomi, Ishikawa 923-1292, Japan
[d] Graduate School of Information Sciences, Tohoku University, Aoba-yama 6-6-05, Sendai, 980-8579, Japan
[e] CREST, JST, 4-1-8 Honcho, Kawaguchi, Saitama, 332-0012, Japan
[f] Faculty of Economics, Kyushu University, Hakozaki 6-19-1, Higashi-ku, Fukuoka, 812-8581, Japan

ARTICLE INFO

Article history:
Received 3 March 2015
Received in revised form 2 July 2015
Accepted 16 July 2015

ABSTRACT

Suppose that we are given two independent sets I_b and I_r of a graph s
and imagine that a token is placed on each vertex in I_b. Then, the SLID
is to determine whether there exists a sequence of independent sets

(a) (Demaine et al. 2015) @ Theo. Comp. Sci. [I am a co-author]

Sliding tokens on unicyclic graphs

JAPAN ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY 1990

Linear-time algorithm for sliding tokens on trees

Erik D. Demaine [a], Martin L. Demaine [a], Eli Fox-Epstein [b], Duc A. Hoang [c], Takehiro Ito [d,e], Hirotaka Ono [f], Yota Otachi [c], Ryuhei Uehara [c], Takeshi Ya

[a] MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar St., Cambridge, MA 02139, USA
[b] Department of Computer Science, Brown University, 115 Waterman Street, Providence, RI 02912-1910, USA
[c] School of Information Science, Japan Advanced Institute of Science and Technology, Asahidai 1-1, Nomi, Ishikawa 923-1292, Japa
[d] Graduate School of Information Sciences, Tohoku University, Aoba-yama 6-6-05, Sendai, 980-8579, Japan
[e] CREST, JST, 4-1-8 Honcho, Kawaguchi, Saitama, 332-0012, Japan
[f] Faculty of Economics, Kyushu University, Hakozaki 6-19-1, Higashi-ku, Fukuoka, 812-8581, Japan

ARTICLE INFO

Article history:
Received 3 March 2015
Received in revised form 2 July 2015
Accepted 16 July 2015

ABSTRACT

Suppose that we are given two independent sets I_b and I_r
and imagine that a token is placed on each vertex in I_b. Th
is to determine whether there exists a sequence of indep

(a) (Demaine et al. 2015) @ Theo. Comp. Sci. [I am a co-author]

Sliding Token on Bipartite Permutation Graphs

Eli Fox-Epstein[1(✉)], Duc A. Hoang[2], Yota Otachi[2], and Ryuhei Uehara[2]

[1] Brown University, Providence, USA
ef@cs.brown.edu
[2] JAIST, Nomi, Japan
{hoanganhduc,otachi,uehara}@jaist.ac.jp

Abstract. Sliding Token is a natural reconfiguration problem in which vertices of independent sets are iteratively replaced by neighbors. We develop techniques that may be useful in answering the conjecture that Sliding Token is polynomial-time decidable on bipartite graphs. Along the way, we give efficient algorithms for Sliding Token on bipartite permutation and bipartite distance-hereditary graphs.

(b) (Fox-Epstein et al. 2015) @ ISAAC 2015. [I am a co-author]

Sliding tokens on unicyclic graphs

Linear-time algorithm for sliding tokens on trees

Erik D. Demaine [a], Martin L. Demaine [a], Eli Fox-Epstein [b], Duc A. Hoang [c], Takehiro Ito [d,e], Hirotaka Ono [f], Yota Otachi [c], Ryuhei Uehara [c], Takeshi Ya

[a] MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar St., Cambridge, MA 02139, USA
[b] Department of Computer Science, Brown University, 115 Waterman Street, Providence, RI 02912-1910, USA
[c] School of Information Science, Japan Advanced Institute of Science and Technology, Asahidai 1-1, Nomi, Ishikawa 923-1292, Japa
[d] Graduate School of Information Sciences, Tohoku University, Aoba-yama 6-6-05, Sendai, 980-8579, Japan
[e] CREST, JST, 4-1-8 Honcho, Kawaguchi, Saitama, 332-0012, Japan
[f] Faculty of Economics, Kyushu University, Hakozaki 6-19-1, Higashi-ku, Fukuoka, 812-8581, Japan

ARTICLE INFO

Article history:
Received 3 March 2015
Received in revised form 2 July 2015

ABSTRACT

Suppose that we are given two independent sets I_b and I_r and imagine that a token is placed on each vertex in I_b. Tl is to determine whether there exists a sequence of indep

(a) (Demaine et al. 2015) @ Theo. Comp. Sci. [I am a co-author]

Sliding Token on Bipartite Permutation Graphs

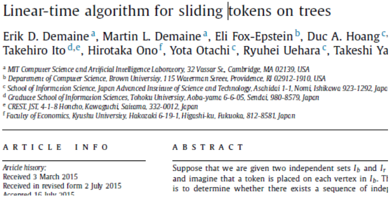
Eli Fox-Epstein[1(✉)], Duc A. Hoang[2], Yota Otachi[2], and Ryuhei Uehara[2]

[1] Brown University, Providence, USA
ef@cs.brown.edu
[2] JAIST, Nomi, Japan
{hoanganhduc,otachi,uehara}@jaist.ac.jp

Abstract. Sliding Token is a natural reconfiguration problem in which vertices of independent sets are iteratively replaced by neighbors. We develop techniques that may be useful in answering the conjecture that Sliding Token is polynomial-time decidable on bipartite graphs. Along the way, we give efficient algorithms for Sliding Token on bipartite permutation and bipartite distance-hereditary graphs.

(b) (Fox-Epstein et al. 2015) @ ISAAC 2015. [I am a co-author]

In both of two papers, the characterization of rigid tokens (tokens that cannot be slid at all) plays a key role.



Linear-time algorithm for sliding tokens on trees

Erik D. Demaine [a], Martin L. Demaine [a], Eli Fox-Epstein [b], Duc A. Hoang [c],
Takehiro Ito [d,e], Hirotaka Ono [f], Yota Otachi [c], Ryuhei Uehara [c], Takeshi Ya

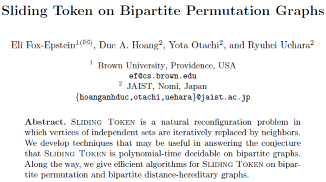
[a] MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar St., Cambridge, MA 02139, USA
[b] Department of Computer Science, Brown University, 115 Waterman Street, Providence, RI 02912-1910, USA
[c] School of Information Science, Japan Advanced Institute of Science and Technology, Asahidai 1-1, Nomi, Ishikawa 923-1292, Japa
[d] Graduate School of Information Sciences, Tohoku University, Aoba-yama 6-6-05, Sendai, 980-8579, Japan
[e] CREST, JST, 4-1-8 Honcho, Kawaguchi, Saitama, 332-0012, Japan
[f] Faculty of Economics, Kyushu University, Hakozaki 6-19-1, Higashi-ku, Fukuoka, 812-8581, Japan

ARTICLE INFO

Article history:
Received 3 March 2015
Received in revised form 2 July 2015

ABSTRACT

Suppose that we are given two independent sets I_b and I_r and imagine that a token is placed on each vertex in I_b. Th is to determine whether there exists a sequence of indep



Sliding Token on Bipartite Permutation Graphs

Eli Fox-Epstein[1(✉)], Duc A. Hoang[2], Yota Otachi[2], and Ryuhei Uehara[2]

[1] Brown University, Providence, USA
ef@cs.brown.edu
[2] JAIST, Nomi, Japan
{hoanganhduc,otachi,uehara}@jaist.ac.jp

Abstract. Sliding Token is a natural reconfiguration problem in which vertices of independent sets are iteratively replaced by neighbors. We develop techniques that may be useful in answering the conjecture that Sliding Token is polynomial-time decidable on bipartite graphs. Along the way, we give efficient algorithms for Sliding Token on bipartite permutation and bipartite distance-hereditary graphs.

(a) (Demaine et al. 2015) @ Theo. Comp. Sci. [I am a co-author]

(b) (Fox-Epstein et al. 2015) @ ISAAC 2015. [I am a co-author]

In both of two papers, the characterization of rigid tokens (tokens that cannot be slid at all) plays a key role.

- For bipartite graphs, all (G, I)-rigid tokens can be determined in linear time. (Fox-Epstein et al. 2015)

Sliding tokens on unicyclic graphs

JAPAN ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY 1990

Linear-time algorithm for sliding tokens on trees

Erik D. Demaine[a], Martin L. Demaine[a], Eli Fox-Epstein[b], Duc A. Hoang[c], Takehiro Ito[d,e], Hirotaka Ono[f], Yota Otachi[c], Ryuhei Uehara[c], Takeshi Ya

[a] MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar St., Cambridge, MA 02139, USA
[b] Department of Computer Science, Brown University, 115 Waterman Street, Providence, RI 02912-1910, USA
[c] School of Information Science, Japan Advanced Institute of Science and Technology, Asahidai 1-1, Nomi, Ishikawa 923-1292, Japa
[d] Graduate School of Information Sciences, Tohoku University, Aoba-yama 6-6-05, Sendai, 980-8579, Japan
[e] CREST, JST, 4-1-8 Honcho, Kawaguchi, Saitama, 332-0012, Japan
[f] Faculty of Economics, Kyushu University, Hakozaki 6-19-1, Higashi-ku, Fukuoka, 812-8581, Japan

ARTICLE INFO

Article history:
Received 3 March 2015
Received in revised form 2 July 2015
Accepted 16 July 2015

ABSTRACT

Suppose that we are given two independent sets I_b and I_r and imagine that a token is placed on each vertex in I_b. Tl is to determine whether there exists a sequence of indej

(a) (Demaine et al. 2015) @ Theo. Comp. Sci. [I am a co-author]

Sliding Token on Bipartite Permutation Graphs

Eli Fox-Epstein[1](✉), Duc A. Hoang[2], Yota Otachi[2], and Ryuhei Uehara[2]

[1] Brown University, Providence, USA
ef@cs.brown.edu
[2] JAIST, Nomi, Japan
{hoanganhduc,otachi,uehara}@jaist.ac.jp

Abstract. Sliding Token is a natural reconfiguration problem in which vertices of independent sets are iteratively replaced by neighbors. We develop techniques that may be useful in answering the conjecture that Sliding Token is polynomial-time decidable on bipartite graphs. Along the way, we give efficient algorithms for Sliding Token on bipartite permutation and bipartite distance-hereditary graphs.

(b) (Fox-Epstein et al. 2015) @ ISAAC 2015. [I am a co-author]

In both of two papers, the characterization of rigid tokens (tokens that cannot be slid at all) plays a key role.

- For bipartite graphs, all (G, I)-rigid tokens can be determined in linear time. (Fox-Epstein et al. 2015)
- What about graphs containing odd cycles?
  - Unicyclic graphs is a good start.

A unicyclic graph is a connected graph that contains exactly one cycle.

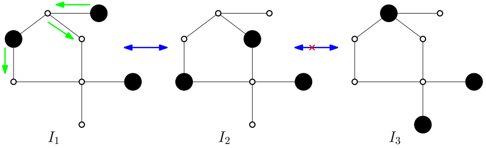


Figure: $I_1 \overset{G}{\leftrightsquigarrow} I_2$ and $I_2 \overset{G}{\not\leftrightsquigarrow} I_3$.

We say that the token t placed at $u \in I$ is (G, I)-rigid if for every independent set I' such that $I \overset{G}{\leftrightsquigarrow} I'$, $u \in I'$. Denote by $\mathsf{R}(G, I)$ the set of all (G, I)-rigid tokens.

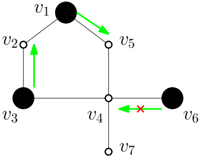


Figure: The token on v_6 is (G, I)-rigid, while the tokens on v_1 and v_3 are not.

Rigid tokens in unicyclic graphs

Lemma: Let I be an independent set of a unicyclic graph G ($|V(G)| = n$). Assume that the unique cycle C of G is of length k ($3 \leq k \leq |V(G)|$).

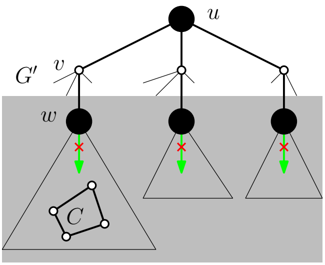
Rigid tokens in unicyclic graphs

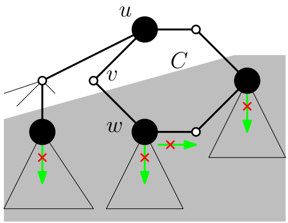
Lemma: Let I be an independent set of a unicyclic graph G ($|V(G)| = n$). Assume that the unique cycle C of G is of length k ($3 \leq k \leq |V(G)|$). For any vertex $u \in I$, the token t on u is (G, I)-rigid if and only if for every vertex $v \in N(G, u)$, there exists a vertex $w \in \big(N(G, v) \setminus \{u\}\big) \cap I$ satisfying one of the following conditions:

Lemma: Let I be an independent set of a unicyclic graph G ($|V(G)| = n$). Assume that the unique cycle C of G is of length k ($3 \leq k \leq |V(G)|$). For any vertex $u \in I$, the token t on u is (G, I)-rigid if and only if for every vertex $v \in N(G, u)$, there exists a vertex $w \in \big(N(G, v) \setminus \{u\}\big) \cap I$ satisfying one of the following conditions:

(i) The token t_w on w is $(G', I \cap G')$-rigid, where $G' = G - N[G, u]$.

(ii) $u \notin V(C)$, $\{v, w\} \subseteq V(C)$, the token t_w on w is not $(G', I \cap G')$-rigid, and for any independent set I' of G' such that $I \cap G' \overset{G'}{\leftrightsquigarrow} I'$, the path $P = C - v$ satisfies $|P \cap I'| = \lfloor k/2 \rfloor$.





$O(n)$ for forests

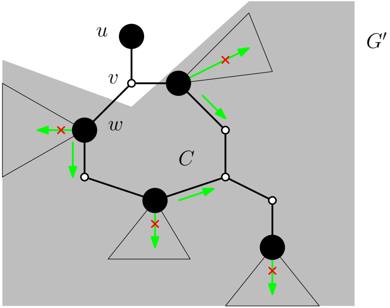
(Demaine et al. 2015)

Rigid tokens in unicyclic graphs

Lemma: Let I be an independent set of a unicyclic graph G ($|V(G)| = n$). Assume that the unique cycle C of G is of length k ($3 \leq k \leq |V(G)|$). For any vertex $u \in I$, the token t on u is (G, I)-rigid if and only if for every vertex $v \in N(G, u)$, there exists a vertex $w \in \left(N(G, v) \setminus \{u\}\right) \cap I$ satisfying one of the following conditions:

(i) The token t_w on w is $(G', I \cap G')$-rigid, where $G' = G - N[G, u]$.

(ii) $u \notin V(C)$, $\{v, w\} \subseteq V(C)$, the token t_w on w is not $(G', I \cap G')$-rigid, and for any independent set I' of G' such that $I \cap G' \overset{G'}{\leftrightsquigarrow} I'$, the path $P = C - v$ satisfies $|P \cap I'| = \lfloor k/2 \rfloor$.



$O(n^2)$

Our contribution

(happen only when k is odd)

The following algorithm checks if $I \overset{G}{\leftrightsquigarrow} J$.

Step 1. Compute $\mathsf{R}(G, I)$ and $\mathsf{R}(G, J)$. If $\mathsf{R}(G, I) \neq \mathsf{R}(G, J)$, then return NO; otherwise go to **Step 2**.

Step 2. Delete the vertices in $\mathsf{R}(G, I) = \mathsf{R}(G, J)$ and its neighbors from G, and obtain a subgraph $\mathcal{F}$ consisting of q connected components $G_1, G_2, \ldots, G_q$. If the number of tokens in I and J are equal for every component, then return YES; otherwise, return NO.

The following algorithm checks if $I \overset{G}{\leftrightsquigarrow} J$.

Step 1. Compute $\mathsf{R}(G, I)$ and $\mathsf{R}(G, J)$. If $\mathsf{R}(G, I) \neq \mathsf{R}(G, J)$, then return NO; otherwise go to **Step 2**.

Time: $O(n^2) \times (|I| + |J|) \Rightarrow O(n^3)$

Step 2. Delete the vertices in $\mathsf{R}(G, I) = \mathsf{R}(G, J)$ and its neighbors from G, and obtain a subgraph $\mathcal{F}$ consisting of q connected components $G_1, G_2, \ldots, G_q$. If the number of tokens in I and J are equal for every component, then return YES; otherwise, return NO.

Time: $O(n)$

Open Questions

1. Standard question: What is the complexity of ISRECONF for ... graph under ... rule?
   - bipartite graphs, cactus graphs, block graphs, interval graphs, etc.
   - TS, TJ, TAR.
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).
   - The SLIDING TOKEN problem for trees was shown to be in P (Demaine et al. 2015), but the corresponding SHORTEST RECONFIGURATION problem is still open?
   - Known polynomial-time result:

     Takeshi Yamada and Ryuhei Uehara. "Shortest Reconfiguration of Sliding Tokens on a Caterpillar". In: *WALCOM 2016, Nepal, March 29-31, 2016 (To be appeared)*
3. The connection decision problem v.s. reconfiguration v.s. shortest reconfiguration provides a different view of "complexity" inspired by games/puzzles.