Patrik Drbal

patrik.drbal@fit.cvut.cz

presents:

On the Complexity of Distance-d Independent Set Reconfiguration

Duc A. Hoang

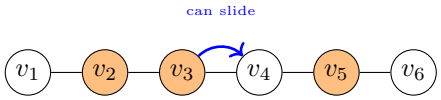
https://arxiv.org/pdf/2208.07199

Combinatorial Reconfiguration of Tokens

Let G be a simple undirected graph and Π be a *vertex subset problem*, i.e., a problem defined on undirected graphs such that a solution consists of a subset of vertices. Then:

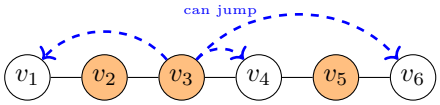
- **Configuration** I is any yes-instance of Π on G. We mark configurations by placing so-called *tokens* on all its vertices. The tokens are undistinguishable from each other.
- **Configuration Space** $\mathcal{S}_\Pi(G,k) \subseteq \binom{V(G)}{k}$ consists of all k-element configurations.
- **Token Movement Rule** $R \in \{\mathsf{TS}, \mathsf{TJ}\}$ defines the single-step relocation of a single token from $u \in I$ to an unoccupied vertex $v \in V(G) \setminus I$, yielding $I' = (I \setminus \{u\}) \cup \{v\}$:

Token Sliding (TS)



A token moves strictly along an existing graph edge to an adjacent unoccupied vertex.

Token Jumping (TJ)



A token jumps directly to any unoccupied vertex in the graph, regardless of edge connectivity.

- **Reconfiguration Graph** $\mathcal{R}_R(G, \mathcal{S}_\Pi)$ is the underlying state-space graph $(\mathcal{S}, E_\mathcal{R})$ where vertices are configurations $I \in \mathcal{S}$, and edges $E_\mathcal{R} = \{\{I, I'\} \subseteq \mathcal{S} \mid I' \text{ is reachable from } I \text{ via rule } R\}$ connect valid adjacent states.
- **Reconfiguration Sequence** is an ordered sequence of valid configurations $\langle I_0, I_1, \ldots, I_\ell \rangle$ with $I_i \in \mathcal{S}$ and $\{I_i, I_{i+1}\} \in E_\mathcal{R}$ for $0 \leq i < \ell$, defining an (I_0, I_ℓ)-path in $\mathcal{R}_R(G, \mathcal{S})$.
- **Reachability Problem** (Π, G, I_s, I_t, R) asks to decide if configurations I_s and I_t belong to the same connected component of $\mathcal{R}_R(G, \mathcal{S}_\Pi)$.

Problem of Interest

Distance-d Independent Set (DdIS)

For a fixed integer $d \geq 2$, a vertex subset $I \subseteq V(G)$ is a **Distance-d Independent Set** if the shortest path distance between any two distinct members in G is at least d.

** Note: The standard independent set is equivalent to D2IS.*

DdIS Reconfiguration (DdISR) Problem

Instance: A graph G, integer $d \geq 2$, movement rule $R \in \{\mathsf{TS}, \mathsf{TJ}\}$, and two D$d$IS configurations $I_s, I_t \subseteq V(G)$ with $|I_s| = |I_t| = k$.

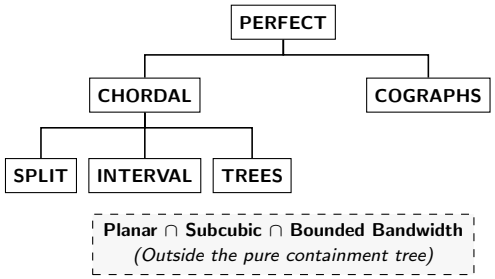
Question: Does there exist a valid reconfiguration sequence in $\mathcal{R}_R(G,k)$ from I_s to I_t?

Results Overview

Graph Class	Token Sliding (TS)		Token Jumping (TJ)	
	$d = 2$ **(ISR)**	$d \geq 3$	$d = 2$ **(ISR)**	$d \geq 3$
Perfect	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-c
Cograph	P	P	P	P
Chordal	PSPACE-c	Unknown	P	Even d: P / Odd d: PSPACE-c
Split	PSPACE-c	P	P	$d = 3$: PSPACE-c / $d \geq 4$: P
Interval	PSPACE-c	Unknown	P	P
Trees	P	Unknown	P	P
Planar*	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-c

*Restricted to planar graphs of maximum degree three and bounded bandwidth.

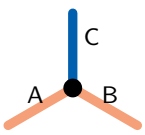
Graph Classes & Structural Hierarchy



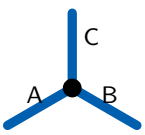
- **perfect graph:** The chromatic number is equal to the size of the maximum clique.
- **chordal graph:** Contains no induced cycles of length greater than 3. The odd power of a chordal graph remains chordal.
- **cograph:** Is P_4-free. Any cograph can be constructed from K_1 by repeated application of union and join operations.
- **split graph:** Can be partitioned into a clique and an independent set, with arbitrary edges in between.
- **interval graph:** There is an interval on the real line for each vertex and there is an edge if and only if the two intervals intersect.

Useful Definitions For Proofs

- **Graph Power Equivalence (G^{d-1}):** A subset $I \subseteq V(G)$ is a DdIS of $G \iff I$ is a standard independent set (D2IS) in the power graph G^{d-1}, where two vertices are adjacent if their distance in G is at most $d-1$. However, this may **not** apply for their reconfiguration variants, as it is true for TJ but not always for TS.
- **Shortest Path Reconfiguration (SPR):** Given a graph H, two vertices $s, t \in V(H)$, and two shortest s-t paths P_s, P_t, decide if P_s can be transformed into P_t by replacing one vertex at a time such that every intermediate step forms an s-t shortest path. SPR serves as a crucial intermediate reduction problem from NCL to establish PSPACE-completeness in restricted classes (e.g., split and chordal graphs).
- **Nondeterministic Constraint Logic (NCL):** The foundational framework for proving PSPACE-completeness in planar, subcubic, and bounded-bandwidth graphs.
  - **And/Or Constraint Graph:** A directed graph containing only And and Or vertices:



And vertex. Edge C may be directed outward if and only if edges A and B are both directed inward.



Or vertex. Edge C may be directed outward if and only if either edge A or edge B is directed inward.

  - **Reconfiguration Move:** Reversing a single directed edge $e = (u, v)$ such that the in-weight constraint remains satisfied at both u and v.
- **Rigid (Frozen) Tokens:** A token on $v \in I$ is *rigid* if no valid reconfiguration sequence in $\mathcal{R}_R(G, \mathcal{S})$ can ever move it off v. It plays a crucial role in the linear-time algorithm for trees by identifying non-movable tokens to recursively prune and reduce the instance.