

## Shifting any path to an avoidable one

Matjaž Krnc

*University of Primorska, Koper, Slovenia*

matjaz.krnc@upr.si

Vladimir Gurvich

*National Research University Higher School of Economics*

vladimir.gurvich@gmail.com

Mikhail Vyalyi

*National Research University Higher School of Economics; Moscow Institute of Physics and Technology; Dorodnicyn Computing Centre, FRC CSC RAS*

vyalyi@gmail.com

Martin Milanič

*University of Primorska, Koper, Slovenia*

martin.milanic@upr.si

A vertex  $v$  in a graph  $G$  is avoidable if every induced path on three vertices with middle vertex  $v$  is contained in an induced cycle. Dirac's classical result from 1961 on the existence of simplicial vertices in chordal graphs is equivalent to the statement that every chordal graph has an avoidable vertex. Beisegel, Chudovsky, Gurvich, Milanič, and Servatius (2019) generalized the notion of avoidable vertices to *avoidable paths*, and conjectured that every graph that contains an induced  $k$ -vertex path also contains an avoidable induced  $k$ -vertex path; they proved the result for  $k = 2$ . The case  $k = 1$  was known much earlier, due to a work of Ohtsuki, Cheung, and Fujisawa in 1976. The conjecture was proved for all  $k$  in 2020 by Bonamy, Defrain, Hatzel, and Thiebaut. This result generalizes the mentioned results regarding avoidable vertices, as well as a result by Chvátal, Rusu, and Sritharan (2002) suggested by West on the existence of simplicial  $k$ -vertex paths in graphs excluding induced cycles of length at least  $k + 3$ .

In this talk we discuss an adaptation of the approach by Bonamy et al. from a reconfiguration point of view. We say that two induced  $k$ -vertex paths are *shifts* of each other if their union is an induced path with  $k + 1$  vertices and show that in every graph, every induced path can be shifted to an avoidable one. We also present an analogous result about paths that are not necessarily induced, where an efficient reconfiguration sequence relies on the properties of depth-first search trees.