

DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE
UNIVERSITY OF SOUTHERN DENMARK, ODENSE

COMPUTER SCIENCE COLLOQUIUM

Planar Reachability in Linear Space and Constant Time

Eva Rotenberg
DTU COMPUTE
Technical University of Denmark

Tuesday, 22 January, 2019 at 14:15

IMADA's Seminar Room

Abstract:

We show how to represent a planar digraph in linear space so that reachability queries can be answered in constant time. The data structure can be constructed in linear time. This representation of reachability is thus optimal in both time and space, and has optimal construction time. The previous best solution used $O(n \log n)$ space for constant query time [Thorup FOCS'01].

Announced at FoCS'15. See also the conference proceedings article: <https://doi.org/10.1109/FOCS.2015.30>

Warm-up-exercise: Given a planar directed acyclic graph with special vertices s and t such any vertex can reach t and any vertex is reachable from s , show how to assign labels of size $O(\log n)$ bits to each vertex, such that for any pair a, b of vertices, one can determine whether there is a path from a to b by inspecting their labels. (That is, determine the *reachability* of b from a .)

Hint: Consider first the case where s and t lie on the same face (e.g. the outer face). In this case, the embedding gives rise to two obvious depth first searches of the graph (starting at s): That of clockwise, and that of counter-clockwise processing of the edges. Assign for each vertex v a tuple $(L(v), R(v))$ containing its post-order numbers according to those depth first searches. Show how to use this tuple to determine reachability.

Hint: To solve the case where s and t do not lie on the same face, make a reduction by “cutting” along a path from s to t , duplicating the vertices on that path. Attach an extra pair of numbers to each vertex: the number-on-path of the first-on-path vertex it can reach, and that of the last-on-path vertex it can be reached from. Show how to use this information to determine reachability.

Host: Jørgen Bang-Jensen