Abstract

Let G be a directed graph (digraph) with m edges and n vertices, and let $G \setminus e$ (resp., $G \setminus v$) be the digraph obtained after deleting edge e (resp., vertex v) from G. We show how to compute in $O(m + n)$ worst-case time:

- The total number of strongly connected components in $G \setminus e$ (resp., $G \setminus v$), *for all edges e* (resp., *for all vertices v*) in G.

- The size of the largest and of the smallest strongly connected components in $G \setminus e$ (resp., $G \setminus v$), *for all edges e* (resp., *for all vertices v*) in G.

Let G be strongly connected. We say that edge e (resp., vertex v) separates two vertices x and y, if x and y are no longer strongly connected in $G \setminus e$ (resp., $G \setminus v$). We also show how to build in $O(m + n)$ time $O(n)$-space data structures that can answer in optimal time the following basic connectivity queries on digraphs:

- Report in $O(n)$ worst-case time all the strongly connected components of $G \setminus e$ (resp., $G \setminus v$), for a query edge e (resp., vertex v).

- Test whether an edge or a vertex separates two query vertices in $O(1)$ worst-case time.

- Report all edges (resp., vertices) that separate two query vertices in optimal worst-case time, i.e., in time $O(k)$, where k is the number of separating edges (resp., separating vertices). (For $k = 0$, the time is $O(1)$).

All our bounds are tight and are obtained with a common algorithmic framework, based on a novel compact representation of the decompositions induced by 1-edge and 1-vertex cuts in digraphs, which might be of independent interest. With the help of our data structures we can design efficient algorithms for several other connectivity problems on digraphs and we can also obtain in linear time a strongly connected spanning subgraph of G with $O(n)$ edges that maintains the 1-connectivity cuts of G and the decompositions induced by those cuts.