Computation of the connectivity number $\kappa(G)$

$G = (V, E)$ is a graph with $n := |V|$, $m := |E|$.

(i) Check whether $\kappa(G) \geq 1$ holds (G is connected):
in linear time $O(n + m)$ time by applying Depth First Search (DFS).

(ii) Check whether $\kappa(G) \geq 2$ holds (G is 2-connected):
in linear time $O(n + m)$ time by applying Depth First Search (DFS).

(iii) Check whether $\kappa(G) \geq 3$ holds (G is 3-connected):
in linear time $O(n + m)$ time, see e.g J.E.Hopcroft and R.E.Tarjan, Dividing a graph into triconnected components, *SIAM J. on Computing* **2**, 1973, 135–158.

(iv) compute $\kappa(G)$
in $O(n^3\sqrt{m})$ time by a straightforward application of Menger's theorem and the push-relabel algorithm for the max-flow problem.
Faster: in $O(\sqrt{n}m^2)$ time by S. Even and R.E.Tarjan, Network flow and testing graph connectivity, *SIAM J. on Computing* **4**, 1975, 507–518.

Computation of the edge-connectivity number $\lambda(G)$

$G = (V, E)$ is a graph with $n := |V|$, $m := |E|$.

Apply Mengers's theorem:
$\lambda(G)$ equals the minimum cut in G

A minimum cut in G can be computed in $O(mn + n^2 \log n)$
by the Stoe-and-Wagner algorithm (SVA)
M. Stoer and F. Wagner, A simple min-cut algorithm, *Journal of the ACM* **44**, 1997, 585–591.

SVA uses *t*he maximum adjacency order (MA order) and
has been discussed in Combinatorial Optimization 1

see also `https://en.wikipedia.org/wiki/Stoer-Wagner_algorithm`

DFS: definitions and notations

Apply $DFS(G, s)$ for a <u>connected</u> graph $G = (V, E)$ with $n := |V|$, $m := |E|$, $s \in V$.

Definition 1.

The edges $\{prec(v), v)\} \in E$, for $v \in V$, are called **tree-edges**. *Set $E(T) := \big\{\{prec(v), v)\} \colon v \in V\big\}$ and $T := (V, E(T))$.*

Observe: T is a spanning tree in G and is called the **DFS-tree**. Think of it as beeing rooted at s with tree-order $\preceq$.

Definition 2.

If $v \preceq w$ holds, then w is called a **descendant** *of v and v is called an* **ancestor** *of w, for $v, w \in V$. For $v \in V$, let T_v be the tree formed by the descendants of v and rooted at v.*

A **non-tree edge** *$\{v, w\} \in E \setminus E(T)$ is called a* **backward edge** *starting at v, if $DFSNum(w) < DFSNum(v)$ at the moment where the DFS passes $\{v, w\}$ for the first time starting at v. The set of backward edges is denoted by B.*

DFS: definitions and notations

Observe: All non-tree edges are backward edges, i.e. $B = E \setminus E(T)$. For every $\{v, w\} \in B$, $w \prec v$ holds, w is contained in the s-v-path in T.

For $v \in V$ set

$$LowPoint(v) := \min\Big\{\{DFSNum(v)\} \cup \{DFSNum(z) \colon v \preceq x\,, \{x, z\} \in B\}\Big\}\,.$$

Aquivalently: $LowPoint(v) := \min(\{DFSNum(v)\} \cup A_v)$, where

$$A_v := \{DFSNum(z) \colon \{v, z\} \in B\} \cup \{LowPoint(w) \colon \{v, w\} \in E(T)\}\,.$$

Definition 3.

A tree-edge $\{v, w\} \in E(T)$ is called a **leading edge** *iff $LowPoint(w) \geq DFSNum(v)$.*

DFS and connectivity

Lemma 1.

Let $\{v, w\}$ be a leading edge in T. Consider the subtree $T' := T_w + \{v, w\}$ of T. Consider a backward edge $\{x, y\} \in B$ starting at a vertex $x \in V(T')$. Then $y \in V(T')$ holds.

Theorem 1.

For all $\{v, w\} \in E$, starting at v and ending at w, with $v \in V \setminus \{s\}$ the following holds: $\{v, w\}$ and the tree-edge ending at v (i.e. an $\{u, v\} \in E(T)$) belong to the same block iff (a) $\{v, w\} \in B$ or (b) $\{v, w\} \in E(T)$ and $\{v, w\}$ is not a leading edge.

Theorem 2.

(a) *The root s of T is a cut-vertex (with $DFSNum(s) = 1$) iff there exist more than one leading edge incident to s.*

(b) *A vertex $v \in V \setminus \{s\}$ (with $DFSNum(s) > 1$) is a cut-vertex iff there exists at least one leading edge starting at v (w.r.t. $\preceq$).*

DFS and connectivity

Theorem 3.

(Tarjan 1972)
The block and the cut-vertices of G can be determined in linear time, i.e. in $O(n+m)$, where $n := |V|$ and $m := |E|$.

Corollary 4.

For a given graph G with $n := |G| > 3$ and $m := |E(G)|$ it can be tested in $O(n+m)$ time whether $\kappa(G) \geq 2$.

Proof: Observe that G is 2-connected iff $bc(G)$ is a singleton.