16.6

Linear time algorithm for finding all strong connected components of a directed graph

16.6.1

Wishful thinking linear-time SCC algorithm

SCC

Finding all SCCs of a Directed Graph

Problem

Given a directed graph $G = (V, E)$, output all its strong connected components.

Straightforward algorithm:

```
Mark all vertices in V as not visited.
for each vertex u ∈ V not visited yet do
    find SCC(G, u) the strong component of u:
        Compute rch(G, u) using DFS(G, u)
        Compute rch(G^rev, u) using DFS(G^rev, u)
        SCC(G, u) ⇐ rch(G, u) ∩ rch(G^rev, u)
        ∀u ∈ SCC(G, u): Mark u as visited.
```

Running time: $O(n(n+m))$

Is there an $O(n+m)$ time algorithm?

Finding all SCCs of a Directed Graph

Problem

Given a directed graph $G = (V, E)$, output all its strong connected components.

Straightforward algorithm:

```
Mark all vertices in V as not visited.
for each vertex u ∈ V not visited yet do
    find SCC(G, u) the strong component of u:
        Compute rch(G, u) using DFS(G, u)
        Compute rch(G^rev, u) using DFS(G^rev, u)
        SCC(G, u) ⇐ rch(G, u) ∩ rch(G^rev, u)
        ∀u ∈ SCC(G, u): Mark u as visited.
```

Running time: $O(n(n+m))$

Is there an $O(n+m)$ time algorithm?

Finding all SCCs of a Directed Graph

Problem

Given a directed graph $G = (V, E)$, output all its strong connected components.

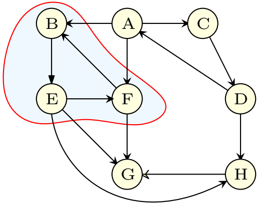
Straightforward algorithm:

```
Mark all vertices in V as not visited.
for each vertex u ∈ V not visited yet do
    find SCC(G, u) the strong component of u:
        Compute rch(G, u) using DFS(G, u)
        Compute rch(G^rev, u) using DFS(G^rev, u)
        SCC(G, u) ⇐ rch(G, u) ∩ rch(G^rev, u)
        ∀u ∈ SCC(G, u): Mark u as visited.
```

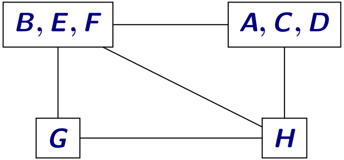
Running time: $O(n(n + m))$

Is there an $O(n + m)$ time algorithm?

Structure of a Directed Graph



Graph G



Graph of SCCs G^{SCC}

Reminder

G^{SCC} is created by collapsing every strong connected component to a single vertex.

Proposition

For a directed graph G, its meta-graph G^{SCC} is a DAG.

Linear-time Algorithm for SCCs: Ideas

Exploit structure of meta-graph...

Wishful Thinking Algorithm

1. Let u be a vertex in a sink SCC of G^{SCC}
2. Do **DFS**(u) to compute $\mathbf{SCC}(u)$
3. Remove $\mathbf{SCC}(u)$ and repeat

Justification

1. **DFS**(u) only visits vertices (and edges) in $\mathrm{SCC}(u)$
2.
3.
4.

Linear-time Algorithm for SCCs: Ideas

Exploit structure of meta-graph...

Wishful Thinking Algorithm

1. Let u be a vertex in a sink SCC of G^{SCC}
2. Do **DFS**(u) to compute $\mathbf{SCC}(u)$
3. Remove $\mathbf{SCC}(u)$ and repeat

Justification

1. **DFS**(u) only visits vertices (and edges) in $\mathrm{SCC}(u)$
2.
3.
4.

Linear-time Algorithm for SCCs: Ideas

Exploit structure of meta-graph...

Wishful Thinking Algorithm

1. Let u be a vertex in a sink SCC of G^{SCC}
2. Do **DFS**(u) to compute $\mathbf{SCC}(u)$
3. Remove $\mathbf{SCC}(u)$ and repeat

Justification

1. **DFS**(u) only visits vertices (and edges) in $\mathrm{SCC}(u)$
2. ... since there are no edges coming out a sink!
3.
4.

Linear-time Algorithm for SCCs: Ideas

Exploit structure of meta-graph...

Wishful Thinking Algorithm

1. Let u be a vertex in a sink SCC of G^{SCC}
2. Do **DFS**(u) to compute $\mathbf{SCC}(u)$
3. Remove $\mathbf{SCC}(u)$ and repeat

Justification

1. **DFS**(u) only visits vertices (and edges) in $\mathrm{SCC}(u)$
2. ... since there are no edges coming out a sink!
3. **DFS**(u) takes time proportional to size of $\mathrm{SCC}(u)$
4.

Linear-time Algorithm for SCCs: Ideas

Exploit structure of meta-graph...

Wishful Thinking Algorithm

1. Let u be a vertex in a sink SCC of G^{SCC}
2. Do **DFS**(u) to compute $\mathbf{SCC}(u)$
3. Remove $\mathbf{SCC}(u)$ and repeat

Justification

1. **DFS**(u) only visits vertices (and edges) in $\mathrm{SCC}(u)$
2. ... since there are no edges coming out a sink!
3. **DFS**(u) takes time proportional to size of $\mathrm{SCC}(u)$
4. Therefore, total time $O(n+m)$!

Big Challenge(s)

How do we find a vertex in a sink SCC of G^{SCC}?

Can we obtain an implicit topological sort of G^{SCC} without computing G^{SCC}?

Answer: **DFS**(G) gives some information!

Big Challenge(s)

How do we find a vertex in a sink SCC of $\mathsf{G}^{\mathrm{SCC}}$?

Can we obtain an implicit topological sort of $\mathsf{G}^{\mathrm{SCC}}$ without computing $\mathsf{G}^{\mathrm{SCC}}$?

Answer: **DFS**(G) gives some information!

Big Challenge(s)

How do we find a vertex in a sink SCC of $\mathsf{G}^{\mathrm{SCC}}$?

Can we obtain an implicit topological sort of $\mathsf{G}^{\mathrm{SCC}}$ without computing $\mathsf{G}^{\mathrm{SCC}}$?

Answer: **DFS**(G) gives some information!

THE END

...

(for now)