The scaling limit of a critical random directed graph

Christina Goldschmidt (Oxford),
joint work with

Robin Stephenson (Sheffield)

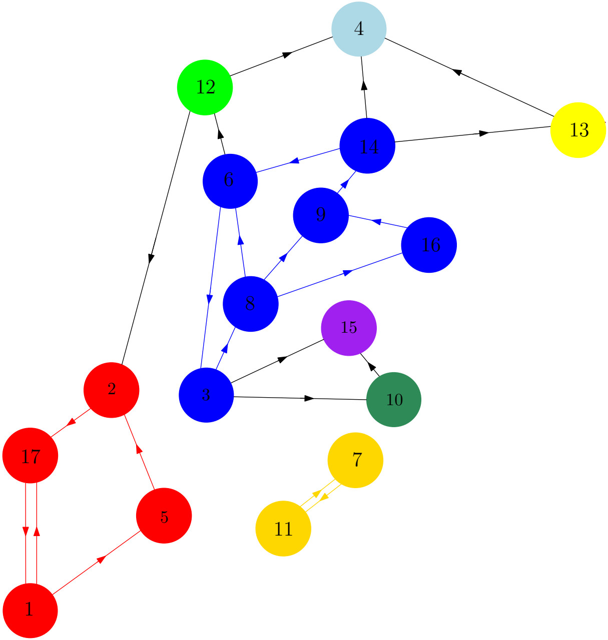
arXiv:1905.05397 [math.PR]

Random directed graphs

Consider the simplest model of a random directed graph, $D(n, p)$, on vertices labelled by $1, 2, \ldots, n$, in which each of the $n(n-1)$ possible directed edges is present independently with probability p, and absent otherwise.

In this talk, I will concentrate on the strongly connected components. This is the collection of maximal subgraphs which are such that for any pair $\{u, v\}$ of vertices in the same subgraph there is a directed path from u to v and a directed path from v to u. If, for a particular u, there is no such v, we consider the singleton $\{u\}$ to be a strongly connected component, so that the strongly connected components partition the vertex set, but not (in general) the edge set.

A digraph and its strongly connected components



The undirected case

The undirected case is the usual Erdős–Rényi random graph $G(n,p)$, in which each of the $\binom{n}{2}$ possible undirected edges is present independently with probability p. This model undergoes a phase transition as follows.

Suppose that $p = c/n + o(n^{-1})$ as $n \to \infty$, for some $c \geq 0$. Then

- if $c < 1$, the components are all $o_{\mathbb{P}}(n)$ in size;
- if $c > 1$, there is a unique component of size $\Theta_{\mathbb{P}}(n)$ (the giant), and all others are $o_{\mathbb{P}}(n)$ in size.

If $p = 1/n + \lambda n^{-4/3} + o(n^{-4/3})$ for some fixed $\lambda \in \mathbb{R}$, the largest component is $\Theta_{\mathbb{P}}(n^{2/3})$, and there is a whole sequence of components on the same order. λ parametrises the critical window.

Phase transition

The digraph $D(n, p)$ undergoes the "same" phase transition.

Theorem (Karp 1990, Łuczak 1990).
Suppose that $p = c/n + o(n^{-1})$ as $n \to \infty$, for $c \geq 0$. Then

- if $c < 1$, all strongly connected components are $o_{\mathbb{P}}(n)$ in size;
- if $c > 1$, there is a unique strongly connected component of size $\Theta_{\mathbb{P}}(n)$ (the giant), and all others are $o_{\mathbb{P}}(n)$ in size.

Phase transition

Theorem (Łuczak and Seierstad, 2009).
Let $p = 1/n + \epsilon_n/n^{4/3}$, where $\epsilon_n \ll n^{1/3}$.

- If $\epsilon_n \to \infty$, then the largest strongly connected component of $D(n, p)$ has size $(4 + o_{\mathbb{P}}(1))\epsilon_n^2 n^{1/3}$ and the second largest has size $O_{\mathbb{P}}(n^{1/3}/\epsilon_n)$;
- if $\epsilon_n \to -\infty$ then the largest strongly connected component has size $O_{\mathbb{P}}(n^{1/3}/|\epsilon_n|)$.

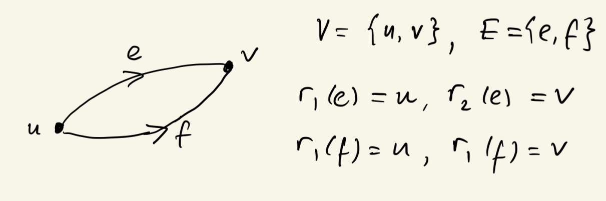
We will concentrate on the critical window, where $\epsilon_n \to \lambda$ for some $\lambda \in \mathbb{R}$ as $n \to \infty$. This result suggests that we should get critical components of size $\Theta_{\mathbb{P}}(n^{1/3})$. (See also [Coulson, 2019] who shows that the size of the largest strongly connected component is tight when rescaled by $n^{-1/3}$.)

We prove a scaling limit: on rescaling the ordered sequence of components by $n^{-1/3}$, we obtain a continuum limit object.

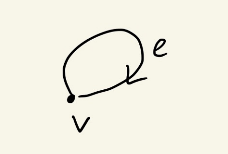
Some terminology

Since we are going to rescale, we need a continuum notion of a directed graph (or, in fact, a directed multigraph).

- A directed multigraph is a triple (V, E, r) where V and E are finite sets and $r = (r_1, r_2)$ is a function from E to $V \times V$. The tail of the directed edge e is $r_1(e)$ and its head is $r_2(e)$.



- The case where $V = \{v\}$, $E = \{e\}$ and $r_1(e) = r_2(e) = v$ is called a loop.



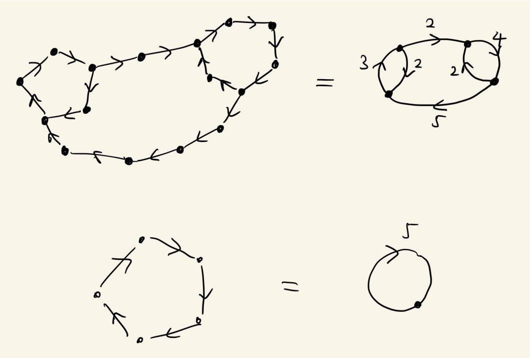
Some terminology

- We call a metric directed multigraph (MDM) a 4-tuple (V, E, r, ℓ) such that (V, E, r) is a directed multigraph and $\ell : E \to (0, \infty)$ is a function which assigns each edge a length.
- We will also allow the degenerate case of a loop of zero length, which we denote by $\mathfrak{L}$, and which will play a special role.

The notion of strong connectivity clearly carries over straightforwardly to these settings.

Main result

Let $C_1(n), C_2(n), \ldots$ be the strongly connected components of $D(n,p)$. We think of this as a sequence of MDMs by thinking of a maximal path of length k of degree 2 vertices as a single edge of length k between the end-points (in the case of a loop, we leave a single vertex).



Main result

Let $C_1(n), C_2(n), \ldots$ be the strongly connected components of $D(n,p)$. We think of this as a sequence of MDMs by thinking of a maximal path of length k of degree 2 vertices as a single edge of length k between the end-points (in the case of a loop, we leave a single vertex). Complete the list with an infinite sequence of copies of $\mathfrak{L}$.

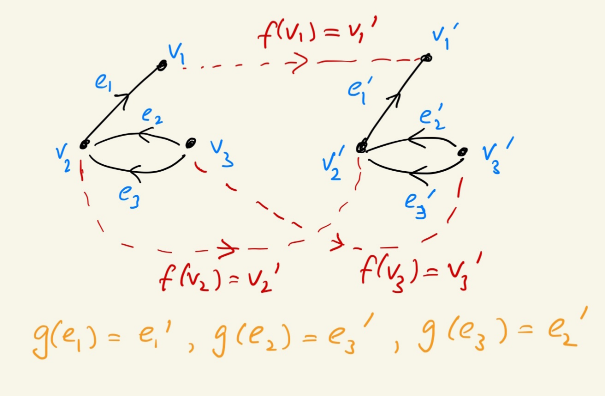
For an MDM $X = (V, E, r, \ell)$ and a real number $a > 0$, write aX as a shorthand for $(V, E, r, a\ell)$.

Theorem (G. and Stephenson, 2019+).
Suppose that $p = 1/n + \lambda n^{-4/3} + o(n^{-4/3})$. Then there exists a sequence $(\mathcal{C}_i, i \geq 1)$ of random strongly connected MDMs such that, for each $i \geq 1$, $\mathcal{C}_i$ is either 3-regular or a loop, and such that

$$\left(\frac{C_i(n)}{n^{1/3}}, i \geq 1\right) \xrightarrow{d} (\mathcal{C}_i, i \geq 1) \qquad \text{as } n \to \infty.$$

The sense of the convergence

For two MDMs $X = (V, E, r, \ell)$ and $X' = (V', E', r', \ell')$, let $\mathrm{Isom}(X, X')$ be the set of graph isomorphisms from X to X' i.e. pairs of bijections $f : V \to V'$ and $g : E \to E'$ such that, for all $e \in E$, $r'(g(e)) = (f(r_1(e)), f(r_2(e)))$.



The sense of the convergence

For two MDMs $X = (V, E, r, \ell)$ and $X' = (V', E', r', \ell')$, let $\mathrm{Isom}(X, X')$ be the set of graph isomorphisms from X to X' i.e. pairs of bijections $f : V \to V'$ and $g : E \to E'$ such that, for all $e \in E$, $r'(g(e)) = (f(r_1(e)), f(r_2(e)))$.

Then set

$$d(X, X') = \inf_{(f,g)\in \mathrm{Isom}(X,X')} \sum_{e\in E} |\ell(e) - \ell'(g(e))|.$$

Note that if $\mathrm{Isom}(X, X')$ is empty we set $d(X, X') = \infty$.

For sequences $\mathbf{X} = (X_1, X_2, \ldots)$ and $\mathbf{X}' = (X_1', X_2', \ldots)$ of MDMs, let

$$\mathbf{dist}(\mathbf{X}, \mathbf{X}') = \sum_{i=1}^{\infty} d(X_i, X_i').$$

Main result (precise version)

Let $C_1(n), C_2(n), \ldots$ be the strongly connected components of $D(n,p)$. We think of this as a sequence of MDMs by thinking of a maximal path of length k of degree 2 vertices as a single edge of length k between the end-points (in the case of a loop, we leave a single vertex). Complete the list with an infinite sequence of copies of $\mathfrak{L}$.

Theorem (G. and Stephenson, 2019+) Suppose that $p = 1/n + \lambda n^{-4/3} + o(n^{-4/3})$. Then there exists a sequence $(\mathcal{C}_i, i \geq 1)$ of random strongly connected MDMs such that, for each $i \geq 1$, $\mathcal{C}_i$ is either 3-regular or a loop, and such that

$$\left(\frac{C_i(n)}{n^{1/3}}, i \geq 1 \right) \xrightarrow{d} (\mathcal{C}_i, i \geq 1)$$

as $n \to \infty$ with respect to **dist**.

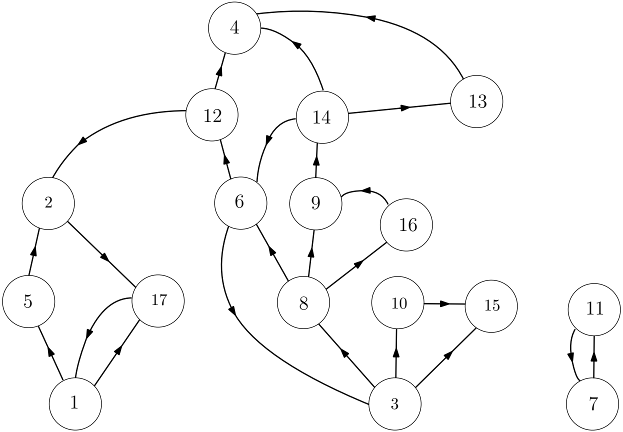
In particular, the limit object has finite total length.

Algorithms

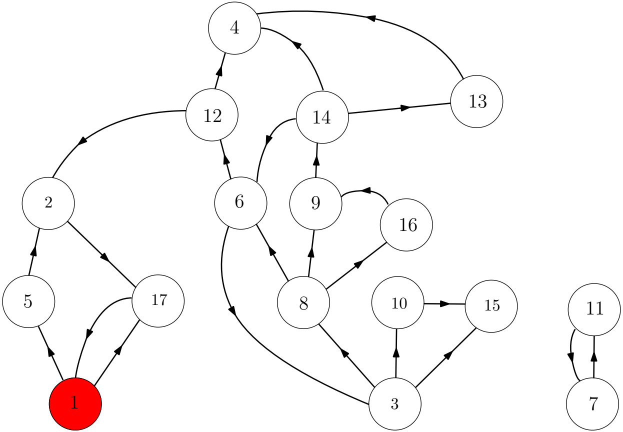
There are several linear-time algorithms for finding the strongly connected components of a digraph. We will use a variant of Tarjan's algorithm.

Let D be an arbitrary digraph with vertices $[n]$. First, we extract from D a directed forest which spans the vertices.

Depth-first exploration

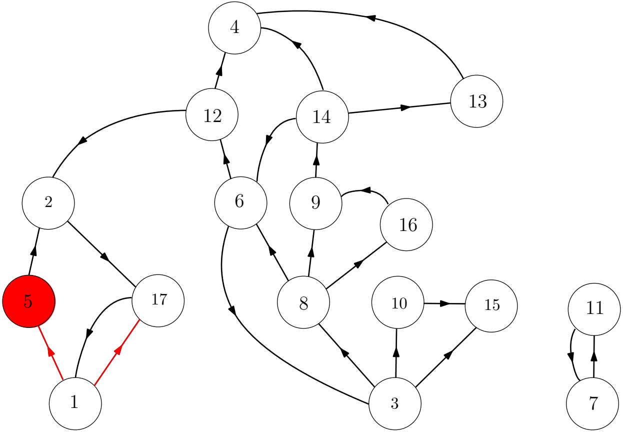


Depth-first exploration



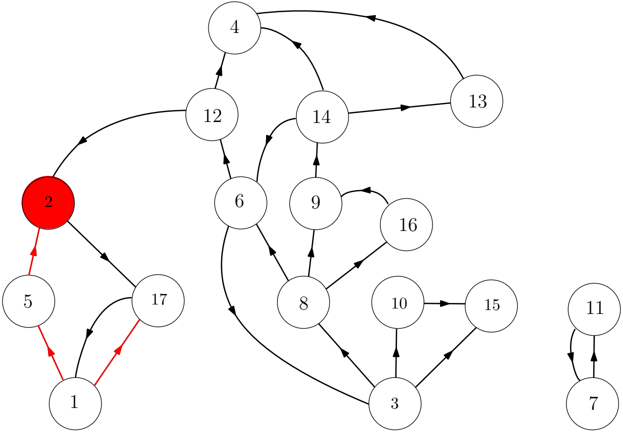
Start from the lowest-labelled vertex.

Depth-first exploration



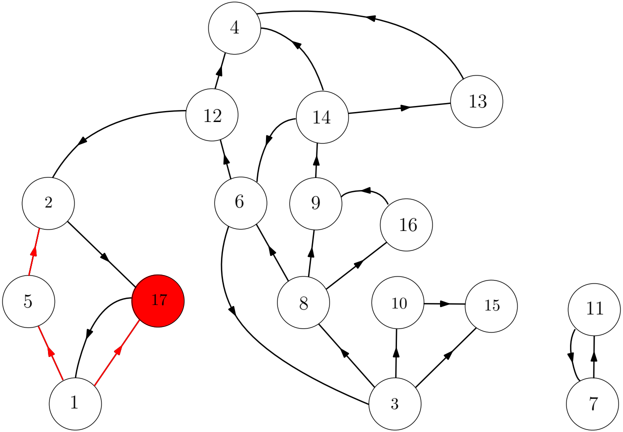
Reveal the out-neigbours (if any) and move to the lowest-labelled.
Keep a stack of vertices seen but not yet explored.

Depth-first exploration



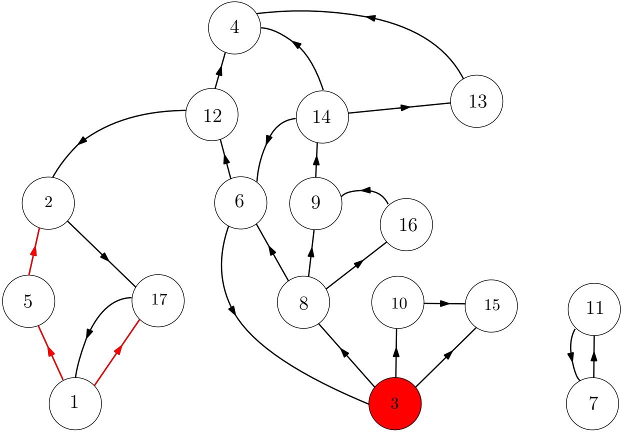
Reveal the unseen out-neigbours (if any) and move to the lowest-labelled. Keep a stack of vertices seen but not yet explored.

Depth-first exploration



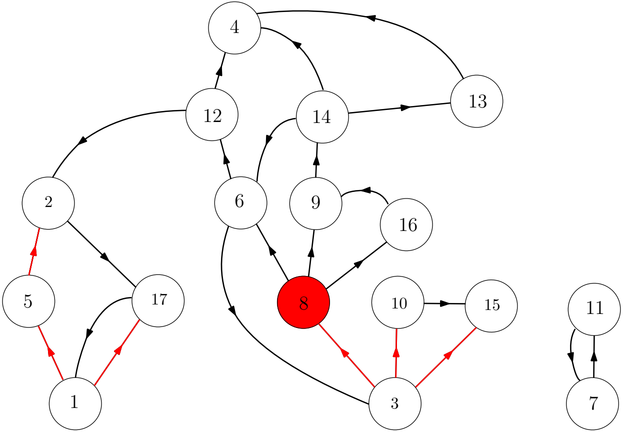
If there are no unseen out-neighbours, move to the vertex on top of the stack.

Depth-first exploration

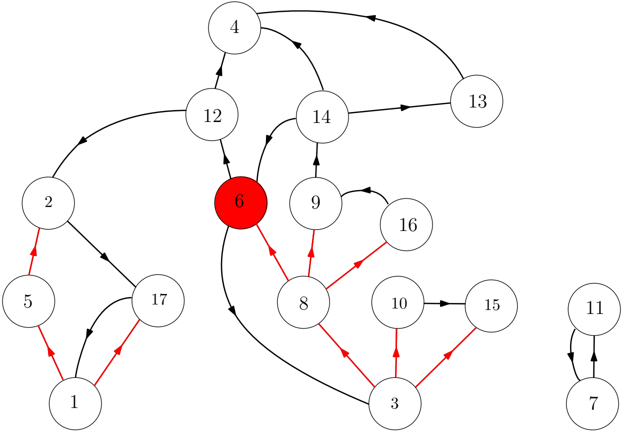


If the stack becomes empty, pick the lowest-labelled vertex which has not yet been visited.

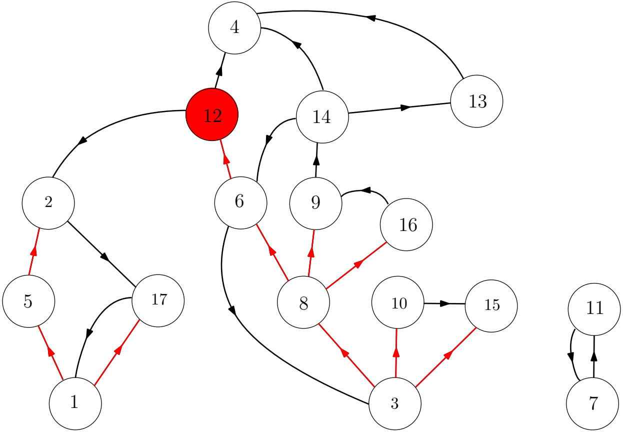
Depth-first exploration



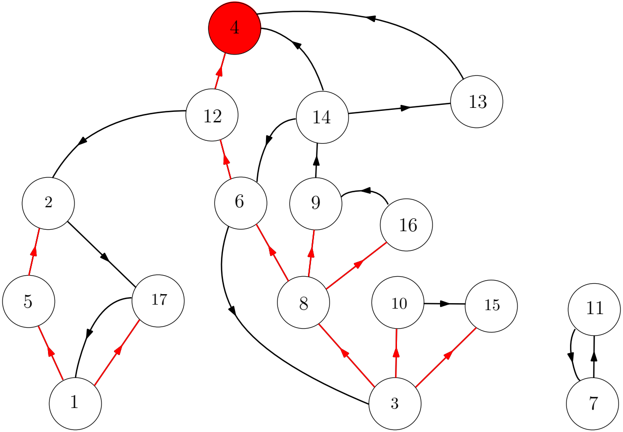
Depth-first exploration



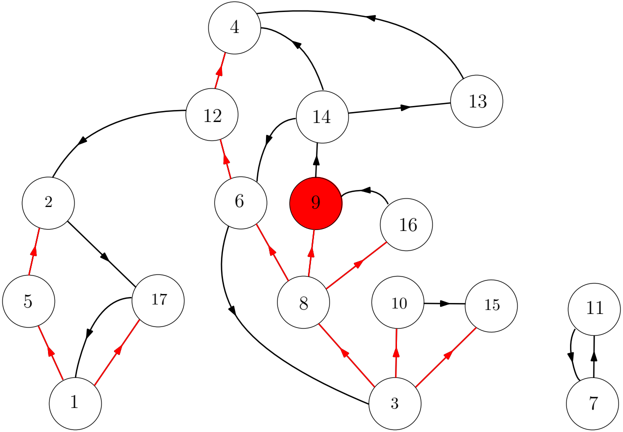
Depth-first exploration



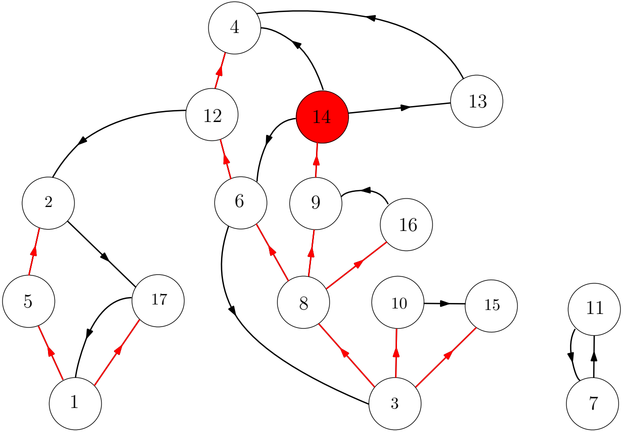
Depth-first exploration



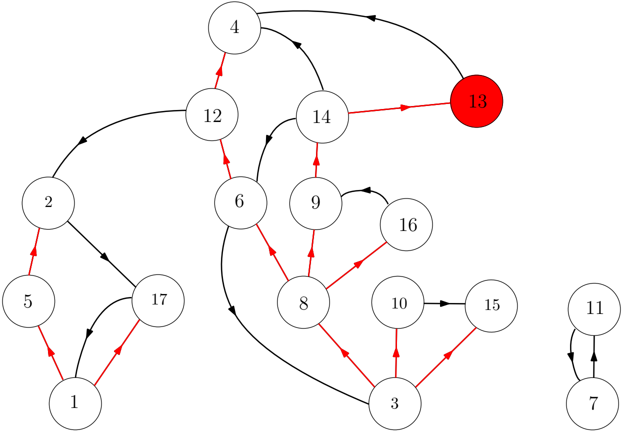
Depth-first exploration



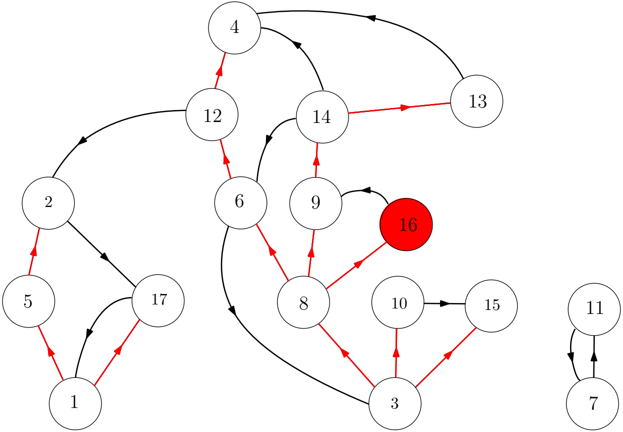
Depth-first exploration



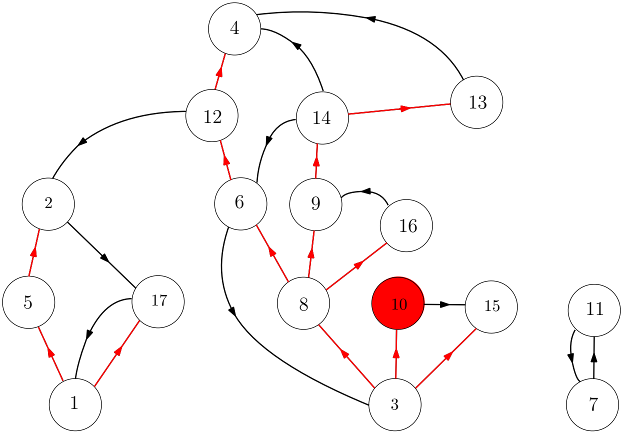
Depth-first exploration



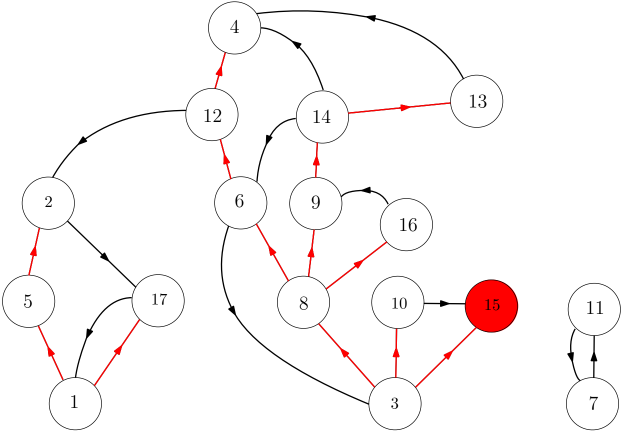
Depth-first exploration



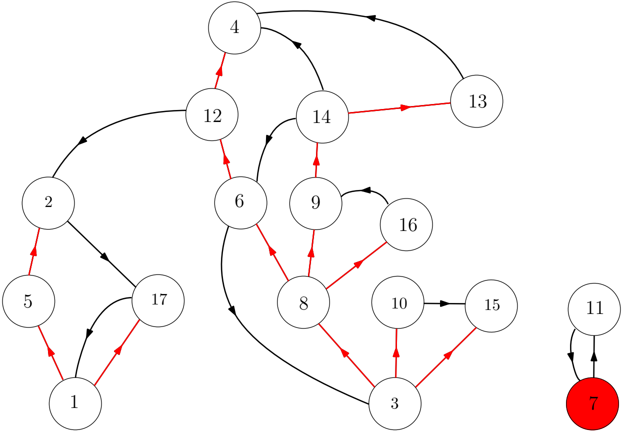
Depth-first exploration



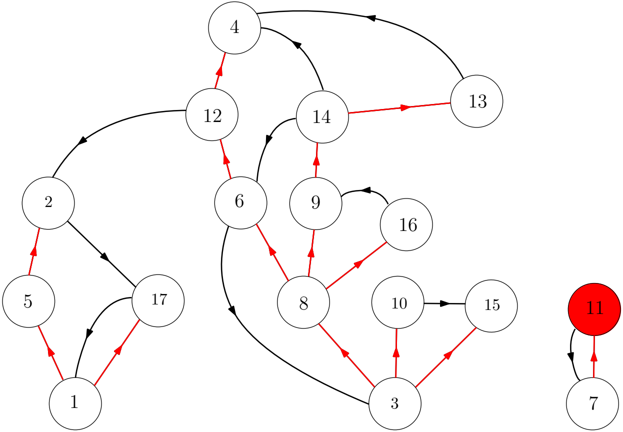
Depth-first exploration



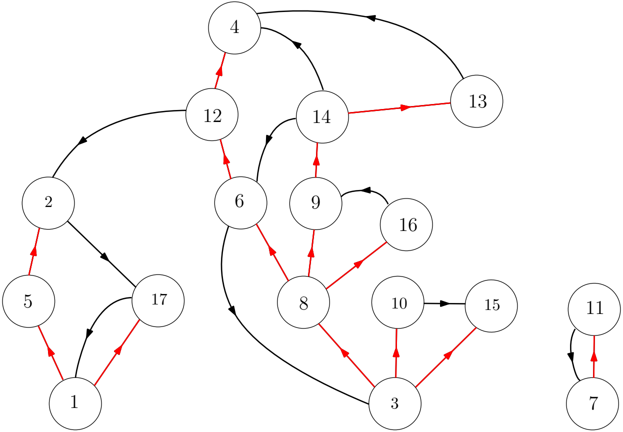
Depth-first exploration



Depth-first exploration



Depth-first exploration

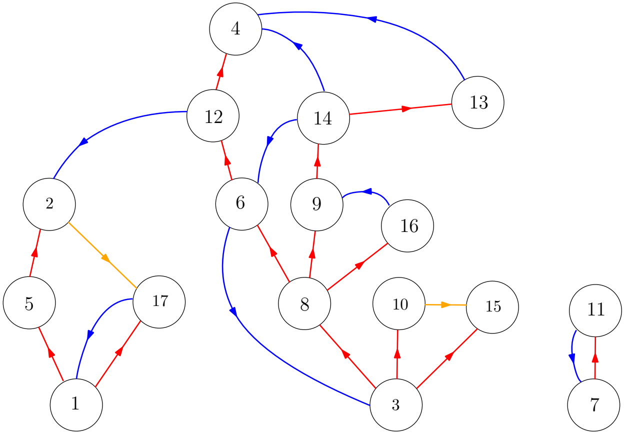


Let $\mathcal{F}_D$ be the directed forest on $[n]$ picked out by the depth-first exploration.

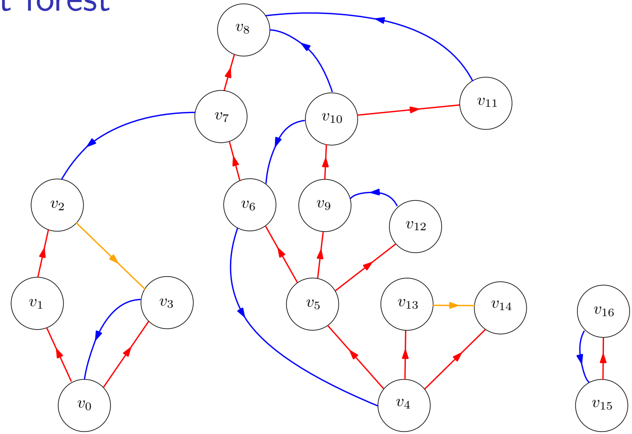
Depth-first forest

Note that this procedure gives an ordering $(v_i)_{0\leq i\leq n-1}$ of the vertices $[n]$, and that the edges of $\mathcal{F}_D$ are all increasing for this ordering. (We will think of it as providing a planar ordering of the forest.)

Depth-first forest



Depth-first forest

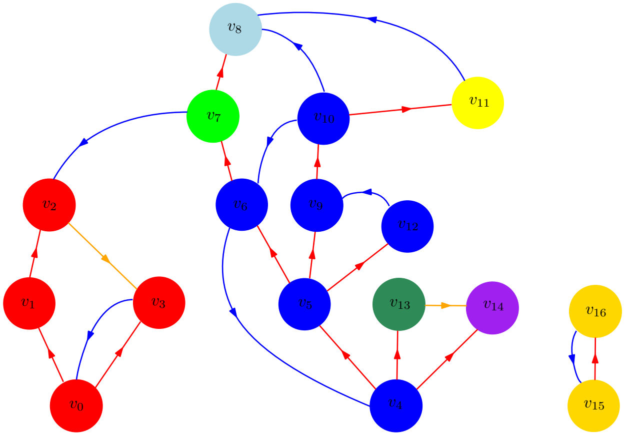


Red edges: increasing for the planar ordering, edges of the directed spanning forest.

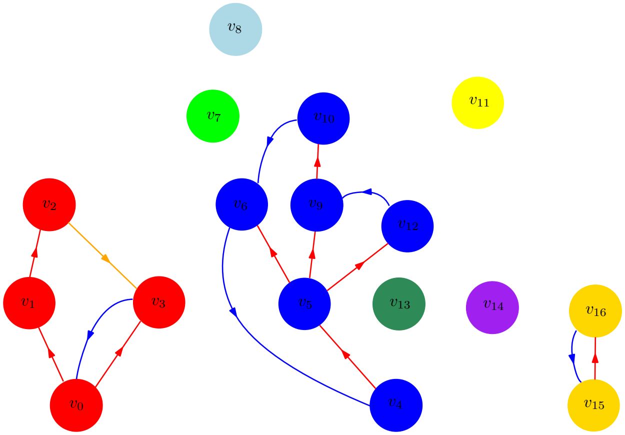
Orange edges: increasing for the planar ordering, but not edges of the forest: "surplus edges".

Blue edges: decreasing for the planar ordering, "back edges".

Strongly connected components



Strongly connected components



Observations

- Any non-singleton strongly connected component must contain at least one forward edge and one back edge.
- Each strongly connected component is contained within a single tree of the directed forest, since there are no forward edges between different trees.
- One such tree can contain multiple strongly connected components.
- Given the directed spanning forest,
  - surplus edges may go from v_j to any vertex on the stack at the time v_j is explored;
  - back edges may go from v_j to v_i for any $i < j$.

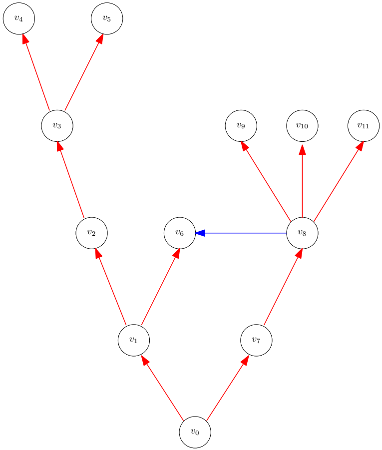
Finding the strongly connected components

To find the strongly connected components, we first note that not all back edges matter.

A particularly important role is played by ancestral back edges, namely those pointing to an ancestor in $\mathcal{F}_D$. Consider a single tree in the directed forest and let us first assume (for simplicity) that there are no surplus edges.

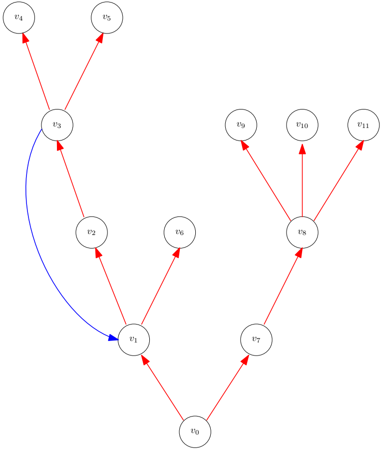
Finding the strongly connected components

If there are only non-ancestral back edges, there are no non-trivial strongly connected components coming from the tree.



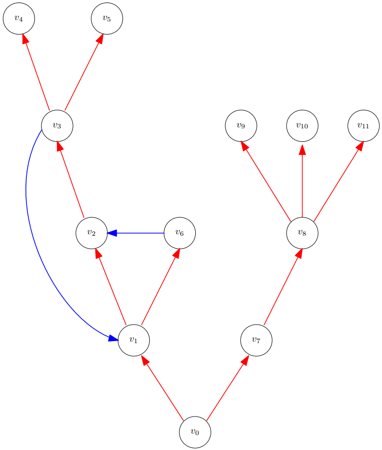
Finding the strongly connected components

On the other hand, as soon as there is one ancestral back edge, we have a non-trivial strongly connected component.



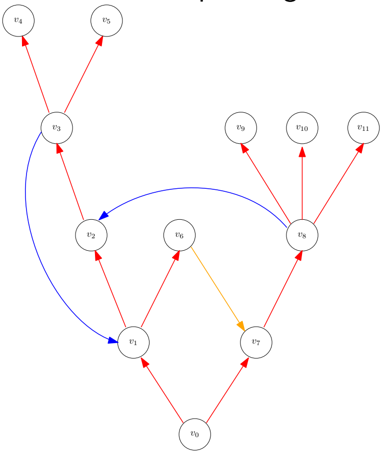
Finding the strongly connected components

Once an ancestral back edge is present, other non-ancestral back edges can "piggy-back" off it.



Finding the strongly connected components

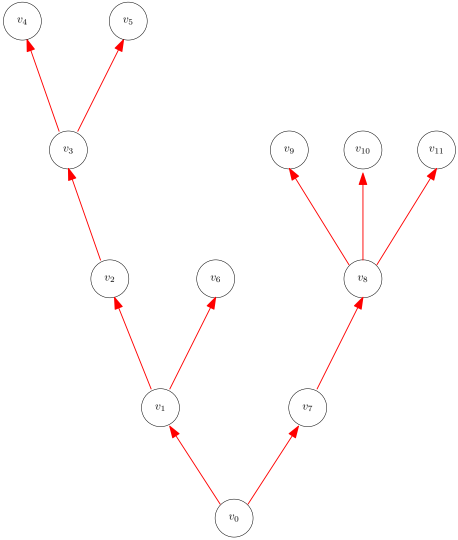
What about surplus edges? Surplus edges go from a vertex to a younger sibling of one its ancestors. This can only help to create a directed cycle if there happens to be a back edge starting in the subtree rooted at that vertex and pointing into a strongly connected part.



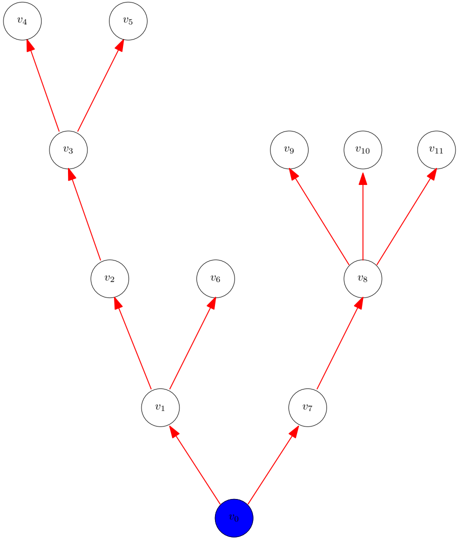
Finding the strongly connected components

Using these ideas, we perform a second exploration of the tree, using the depth-first ordering coming from the first exploration. We now keep track of an active set of vertices and create a list of back edges that matter.

Finding the strongly connected components

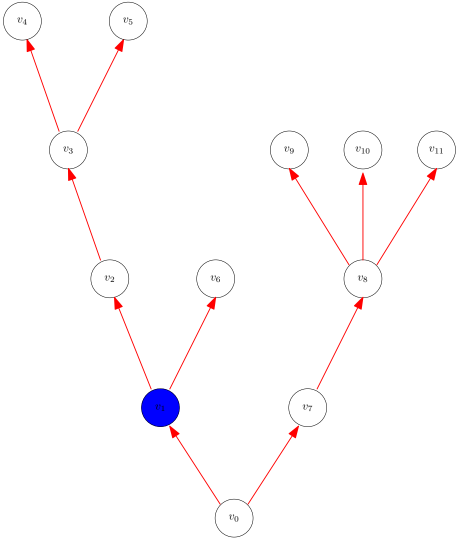


Finding the strongly connected components



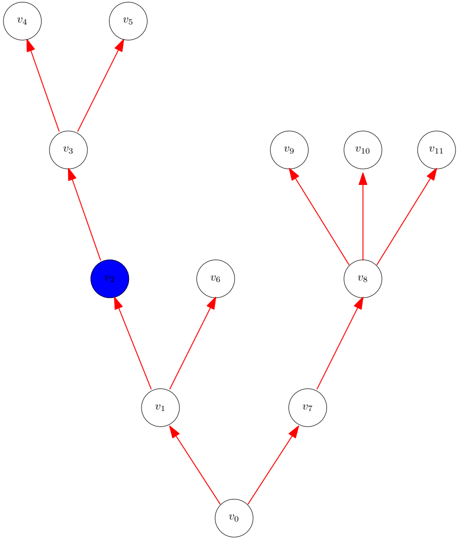
Step 0: active set $A_0 = \emptyset$.

Finding the strongly connected components



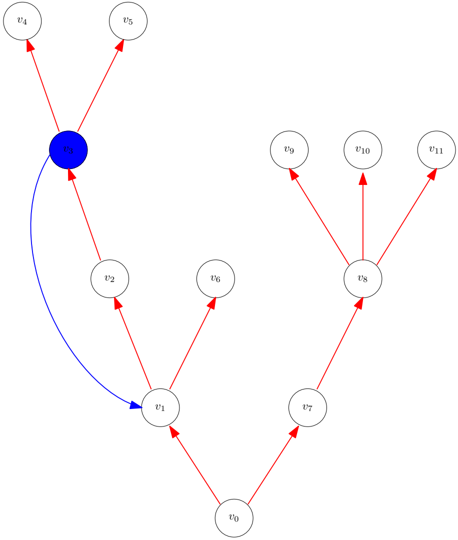
Step 1: active set $A_1 = \{v_0\}$.

Finding the strongly connected components



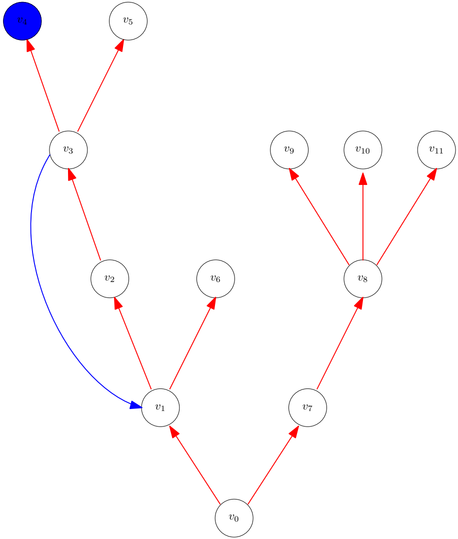
Step 2: active set $A_2 = \{v_0, v_1\}$.

Finding the strongly connected components



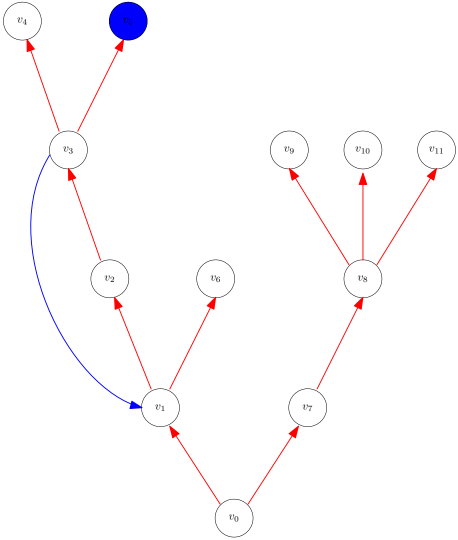
Step 3: active set $A_3 = \{v_0, v_1, v_2\}$.

Finding the strongly connected components



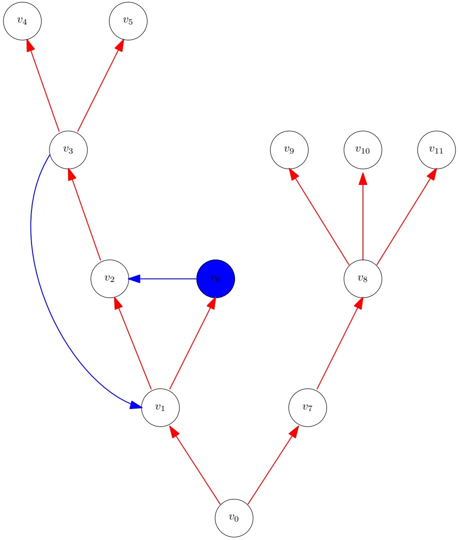
Step 4: active set $A_3 = \{v_0, v_1, v_2, v_3\}$.

Finding the strongly connected components



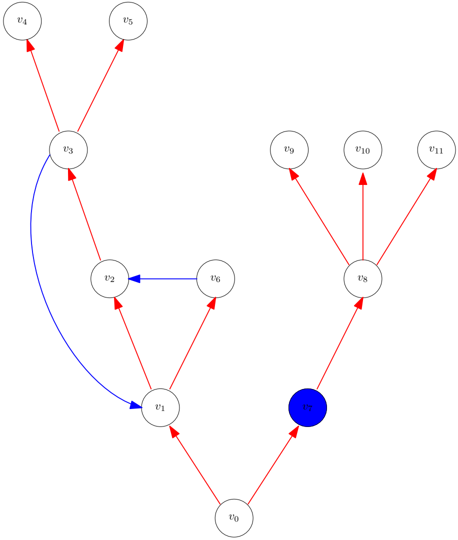
Step 5: active set $A_5 = \{v_0, v_1, v_2, v_3\}$.
Output strongly connected component on $\{v_4\}$.

Finding the strongly connected components



Step 6: active set $A_6 = \{v_0, v_1, v_2, v_3\}$.
Output strongly connected component on $\{v_5\}$.

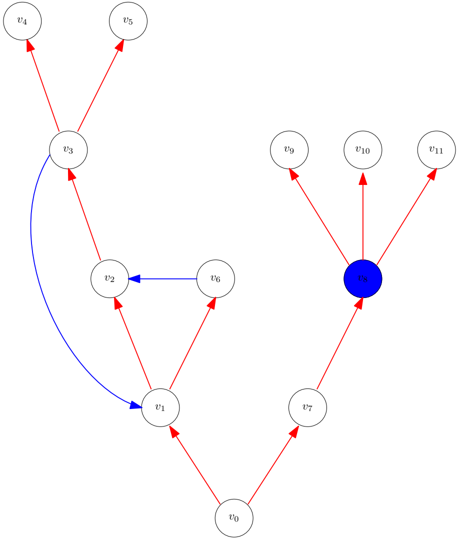
Finding the strongly connected components



Step 7: active set $A_7 = \{v_0\}$.

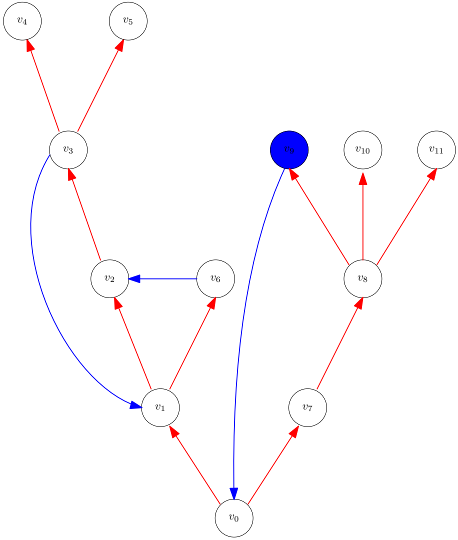
Output strongly connected component on $\{v_1, v_2, v_3, v_6\}$.

Finding the strongly connected components



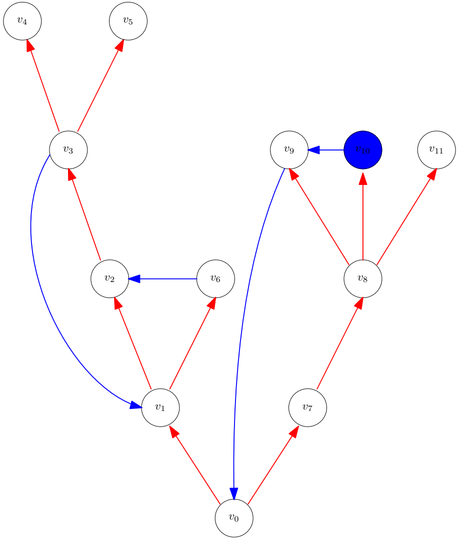
Step 8: active set $A_8 = \{v_0, v_7\}$.

Finding the strongly connected components



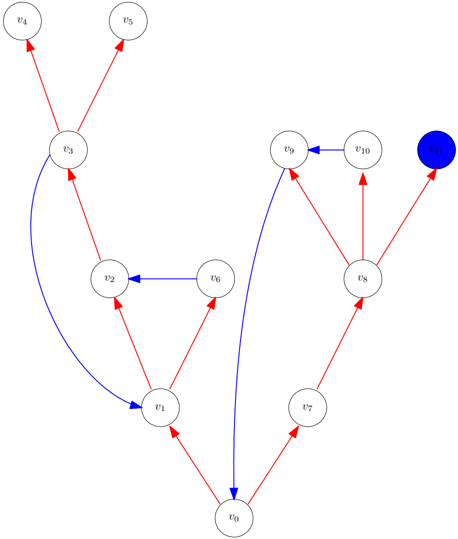
Step 9: active set $A_9 = \{v_0, v_7, v_8\}$.

Finding the strongly connected components



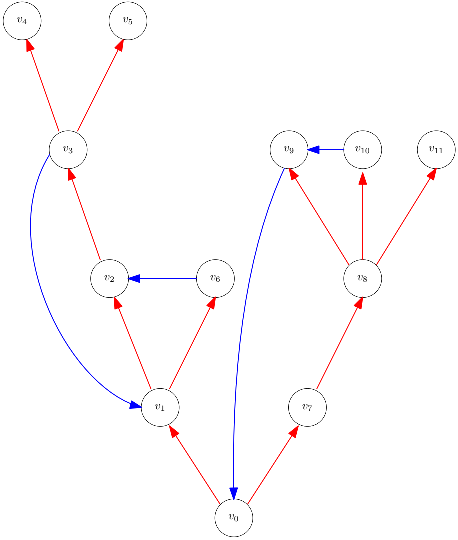
Step 10: active set $A_{10} = \{v_0, v_7, v_8, v_9\}$.

Finding the strongly connected components



Step 11: active set $A_{11} = \{v_0, v_7, v_8, v_9, v_{10}\}$.

Finding the strongly connected components



Output strongly connected components on $\{v_{11}\}$ and $\{v_0, v_7, v_8, v_9, v_{10}\}$.

Observation

Assuming there are no surplus edges, the only back edges which count are those which go from the current vertex at some step i to the current active set A_i.

(If there are surplus edges, we just need to take them into account in the definition of the active set and do something similar.)

Distributional properties

- The distribution of the directed forest is exactly the same as it would be if we did the same exploration on the undirected Erdős–Rényi random graph. This is a well-understood object.
- The possible surplus or back edges are present independently, each with probability p.

Scaling limit of the directed forest

[Aldous, 1997; Addario-Berry, Broutin and G., 2012]

Let $(F_1(n), F_2(n), \ldots)$ be the trees in the directed forest. Then we have the joint convergence in distribution

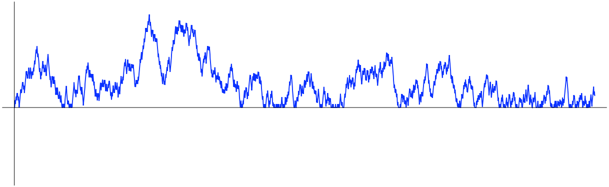
$$\frac{1}{n^{2/3}}(|F_1(n)|, |F_2(n)|, \ldots) \xrightarrow{d} (\sigma_1, \sigma_2, \ldots),$$

$$\frac{1}{n^{1/3}}(F_1(n), F_2(n), \ldots) \xrightarrow{d} (\mathcal{F}_1, \mathcal{F}_2, \ldots),$$

where $(\sigma_1, \sigma_2, \ldots)$ are the ranked lengths of the excursions above 0 of $(2B_t^\lambda, t \geq 0)$ where

$$B_t^\lambda := B_t + \lambda t - t^2/2 - \inf_{s \leq t}(B_s + \lambda s - s^2/2), \quad t \geq 0,$$

and $\mathcal{F}_1, \mathcal{F}_2, \ldots$ are the $\mathbb{R}$-trees encoded by those excursions.

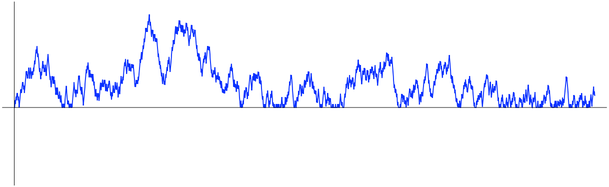


Surplus edges

[Addario-Berry, Broutin and G., 2012]

Consider the depth-first exploration. The process $(2B_t^\lambda, t \geq 0)$ describes the rescaled distance of the current vertex from the root of the subtree being explored. The size of the stack is asymptotically $1/2$ the height of the current vertex.

Surplus edges can go from the current vertex to any of the vertices on the stack, and occur independently with probability $p \sim 1/n$.



In the limit, the surplus edges thus arise as a Poisson point process of intensity $B_t^\lambda dt$ on $\mathbb{R}_+$, and cause the identification of the vertex currently being explored with a uniform vertex along the path to the root. In particular, there are $O(1)$ surplus edges in each tree.

Back edges

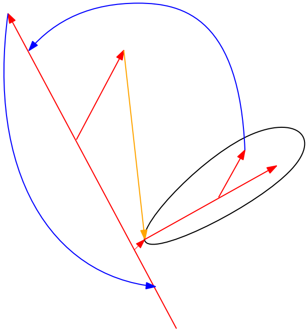
Since a single large tree in the forest is of size $\Theta(n^{2/3})$, there are $\Theta(n^{4/3})$ possible back edges, each of which is present with probability $1/n$. So large trees typically contain $\Theta(n^{1/3})$ back edges, which clearly will not be controllable as $n \to \infty$. Fortunately, we can ignore most of them!

Ancestral back edges can go from the current vertex to any ancestor, and so in the limit arise according to a Poisson point process, this time of intensity $2B_t^\lambda dt$ on $\mathbb{R}_+$.

The full process of back edges which matter is more complicated to describe: one can define the continuum analogue of the active set at each time, which in general consists of a connected subtree. The intensity of back edges is then proportional to the length of the active set. Importantly, this process gives rise to only finitely many back edges in any tree of the directed forest.

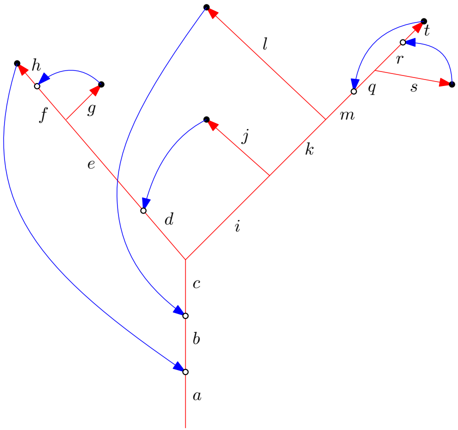
Surplus edges don't contribute

In order for a surplus edge to matter, there needs to be a back edge from the subtree rooted at the head of the surplus edge into the active component.



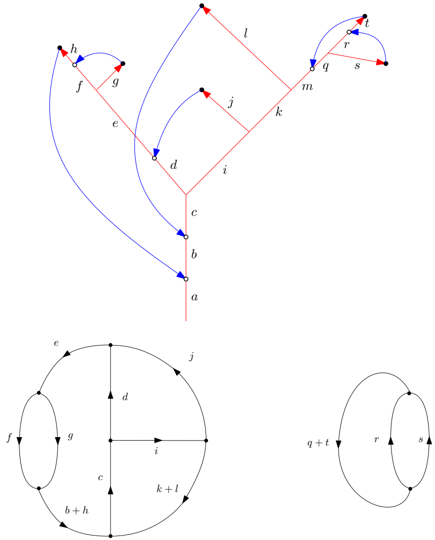
This turns out to have negligible probability in the limit, essentially because the subtree is typically very small. So we may safely ignore the surplus edges.

The scaling limit of the strongly connected components



For each of the trees $\mathcal{F}_i$ consider the subtree $\mathcal{T}_i$ spanned by the root and the back edges that matter. This is a much simpler object than $\mathcal{F}_i$, consisting of finitely many (directed) line-segments and the pairs of points to be identified.

The scaling limit of the strongly connected components



It is then easy to extract the strongly connected components.

The scaling limit of the strongly connected components

The back edges we use a.s. all identify leaves with points of degree 2. Because the underlying trees $\mathcal{F}_i$ are binary a.s., the resulting strongly connected components are 3-regular a.s. (i.e. the vertices either have 2 in-edges and 1 out-edge or 1 in-edge and 2 out-edges.)

Finally, the fact that the total length of the limit object is finite is proved using properties of B^λ.

Thank you!

C. Goldschmidt and R. Stephenson, The scaling limit of a critical random directed graph, arXiv:1905.05397 [math.PR]