Paris-Bath branching structures meeting, June 2014

The scaling limit of the minimum spanning tree of the complete graph

Christina Goldschmidt
University of Oxford

Joint work with **Louigi Addario-Berry** (McGill), **Nicolas Broutin** (INRIA Rocquencourt) and **Grégory Miermont** (ENS Lyon).

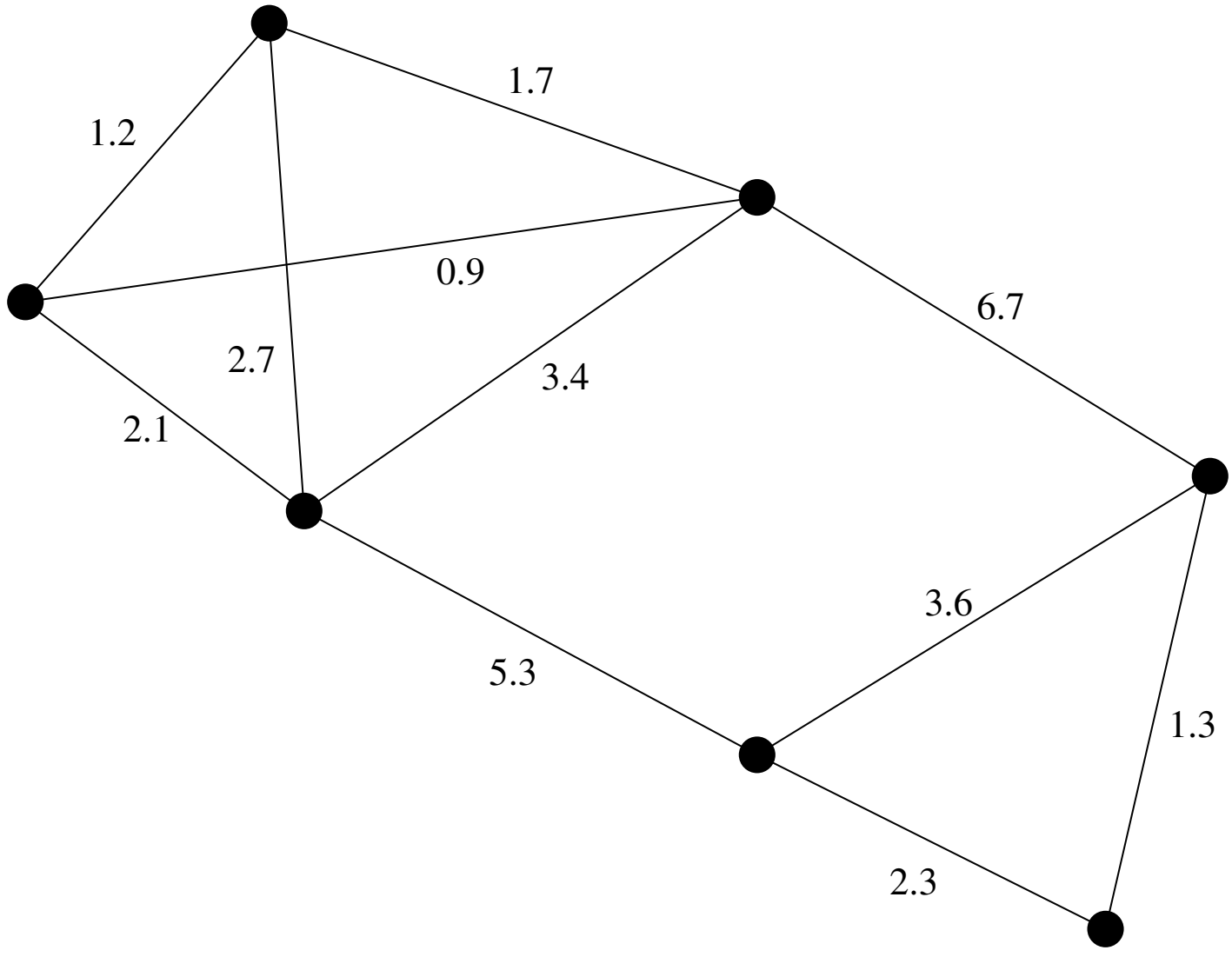
Minimum spanning trees

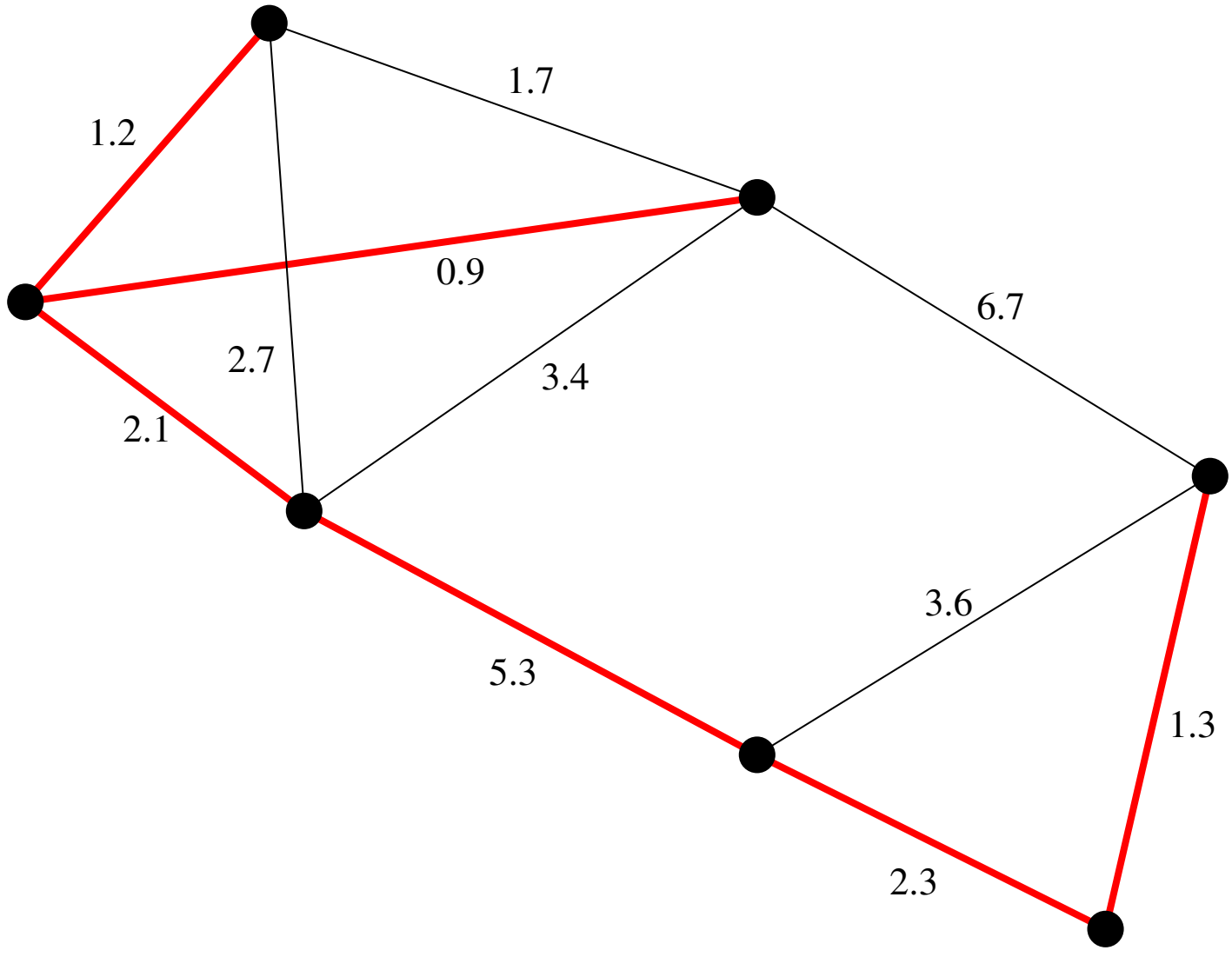
Suppose that $G = (V(G), E(G))$ is a finite connected graph. To each edge $e \in E(G)$, associate a positive weight $w(e)$.

Problem: find the connected subgraph T of G which has the same vertex set as G and minimises

$$\sum_{e \in E(T)} w(e).$$

Since the weights are positive, the minimum is necessarily attained by a tree. If the weights are all distinct, this tree is unique and is the so-called minimum spanning tree (MST).


1.7
1.2
0.9
6.7
2.7
3.4
2.1
3.6
5.3
1.3
2.3


1.7
1.2
0.9
6.7
2.7
3.4
2.1
3.6
5.3
1.3
2.3

Three algorithms

There are three commonly-used algorithms for finding the MST of a graph:

- Kruskal's algorithm
- Prim's algorithm
- Cycle-breaking (reverse Kruskal) algorithm.

Three algorithms

There are three commonly-used algorithms for finding the MST of a graph:

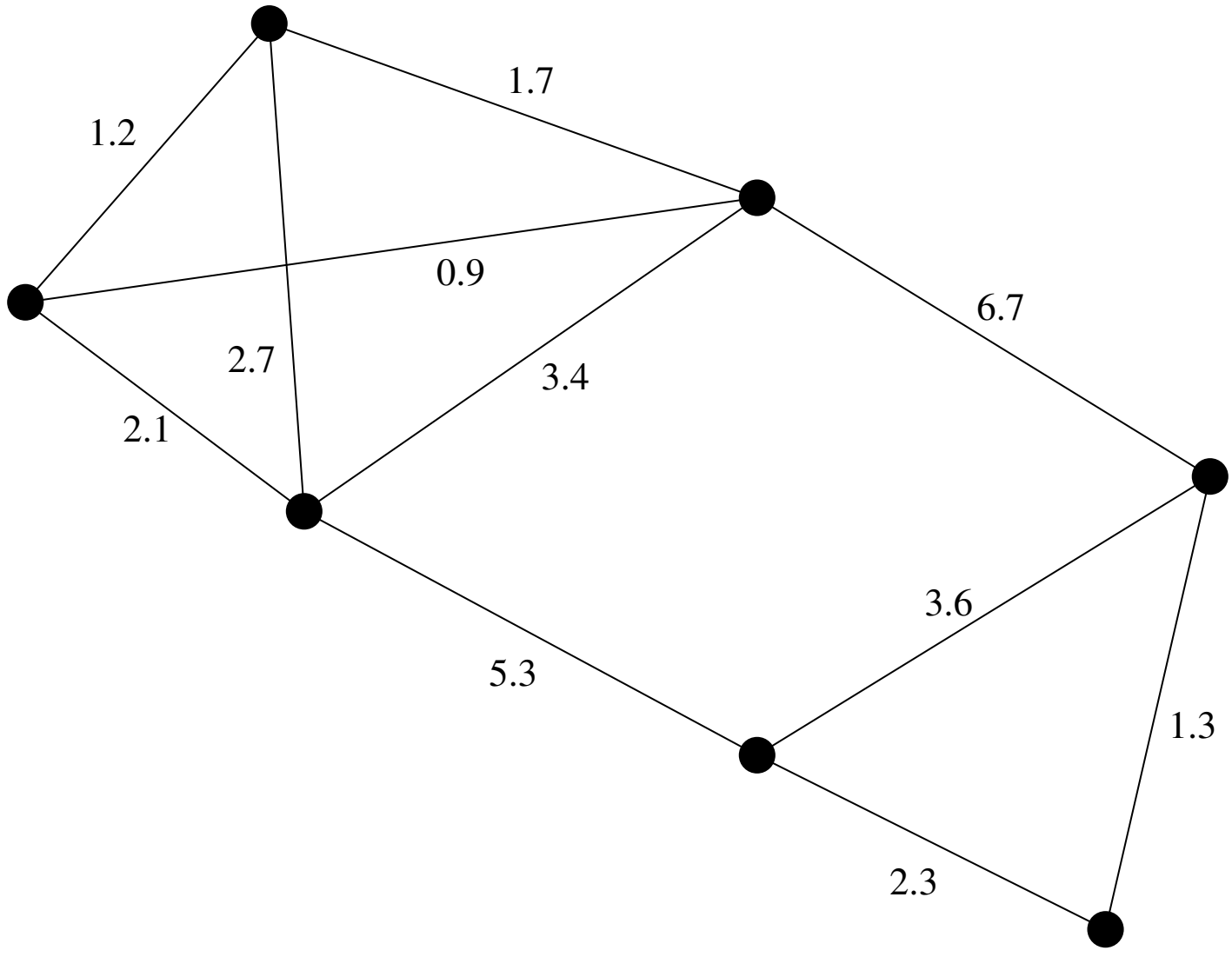
- Kruskal's algorithm
- Prim's algorithm
- Cycle-breaking (reverse Kruskal) algorithm.

Kruskal's algorithm

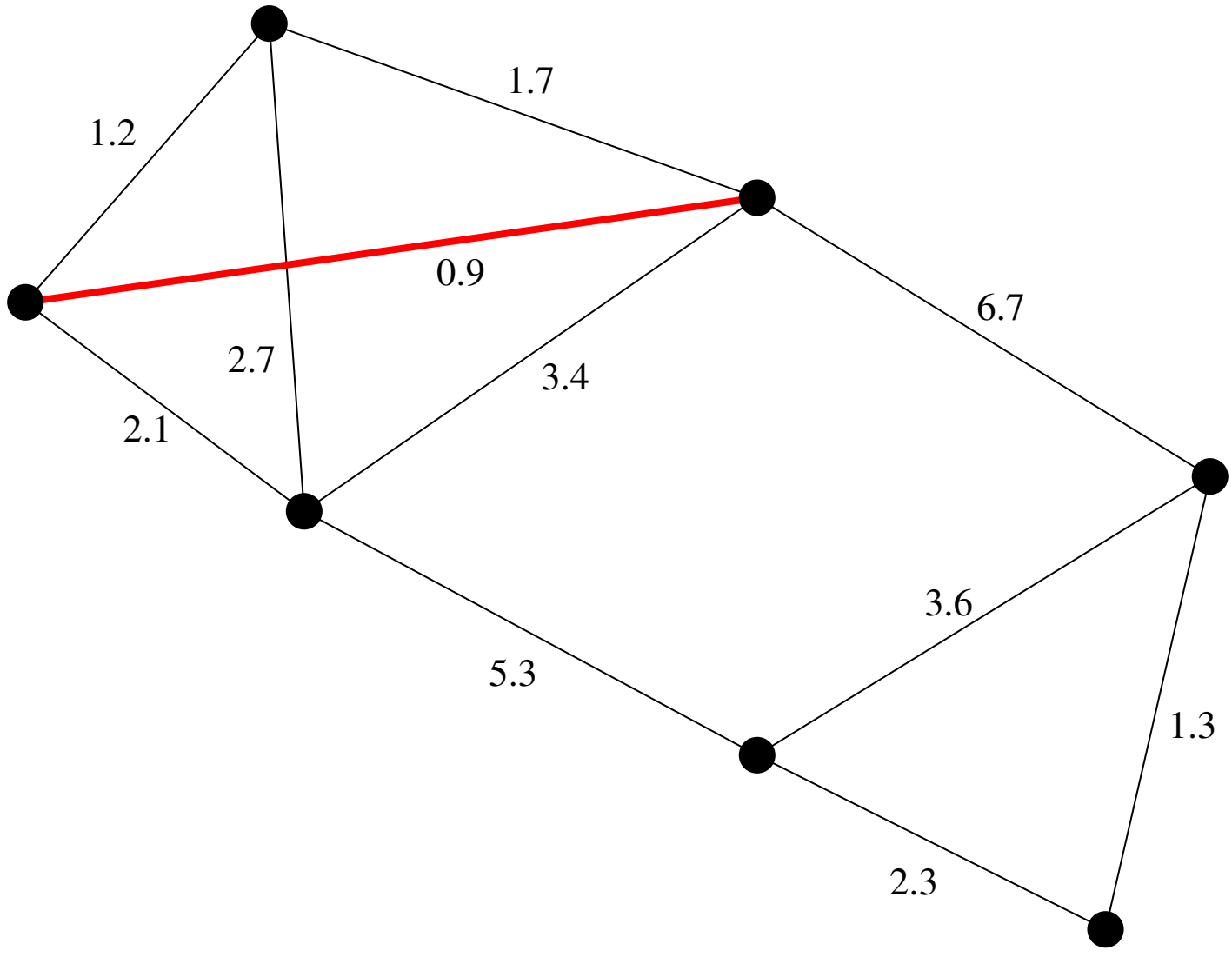
- Start from a forest of isolated vertices. List the edges as $e_1, e_2, \ldots$ in increasing order of weight.
- At step i, add edge e_i as long as it does not create a cycle.
- Stop when all vertices are connected.

At every stage, we have a forest.

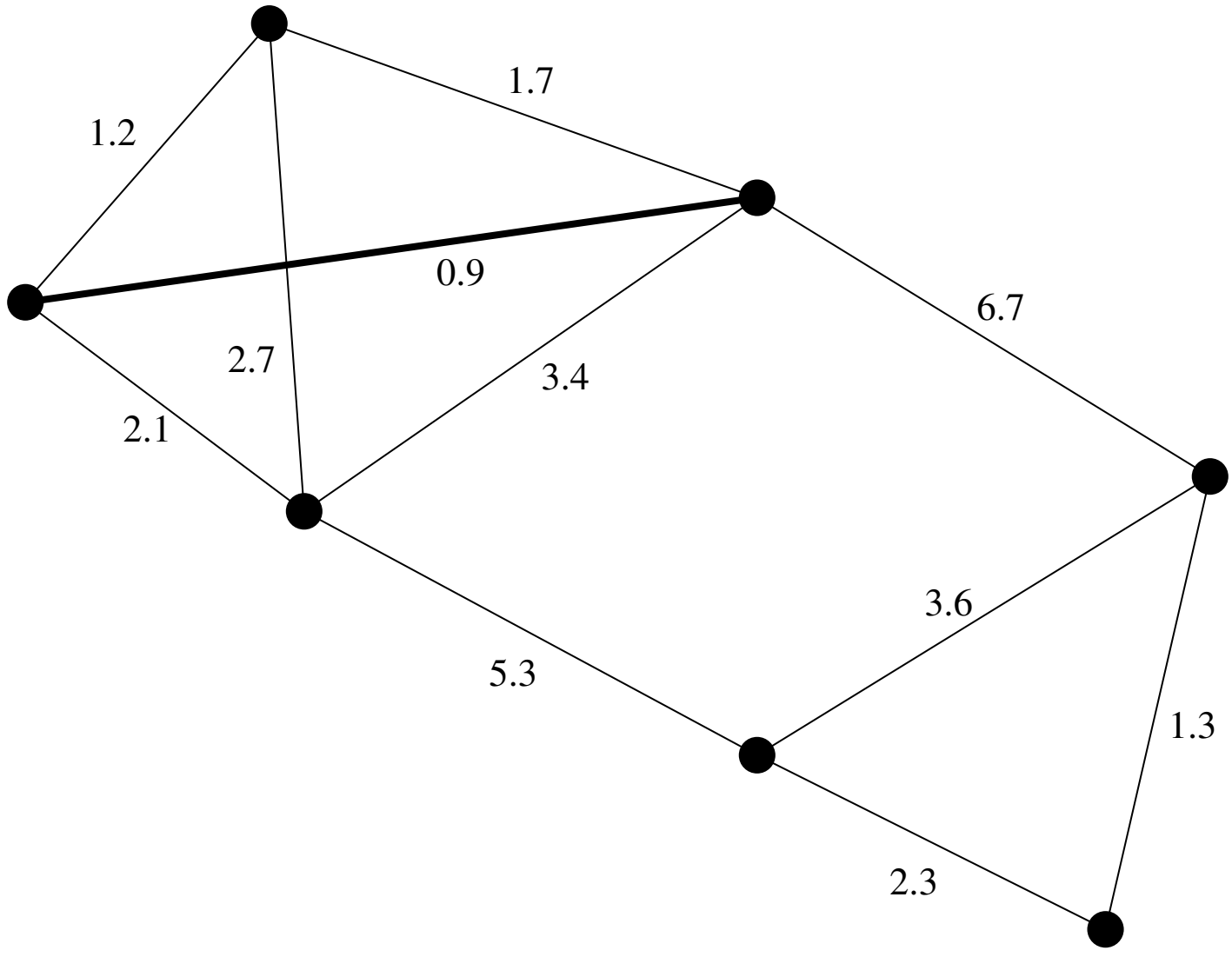
Kruskal's algorithm



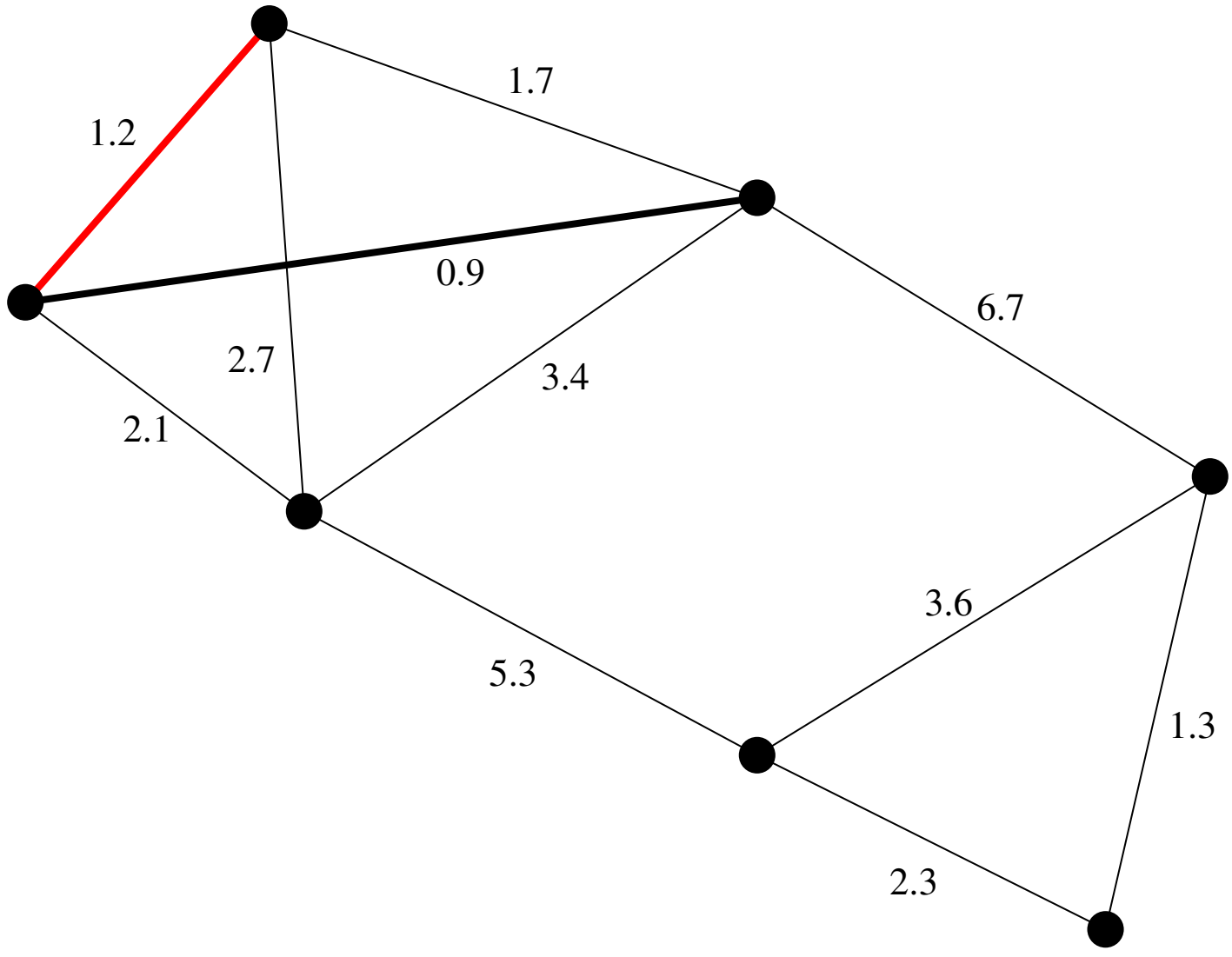
Kruskal's algorithm



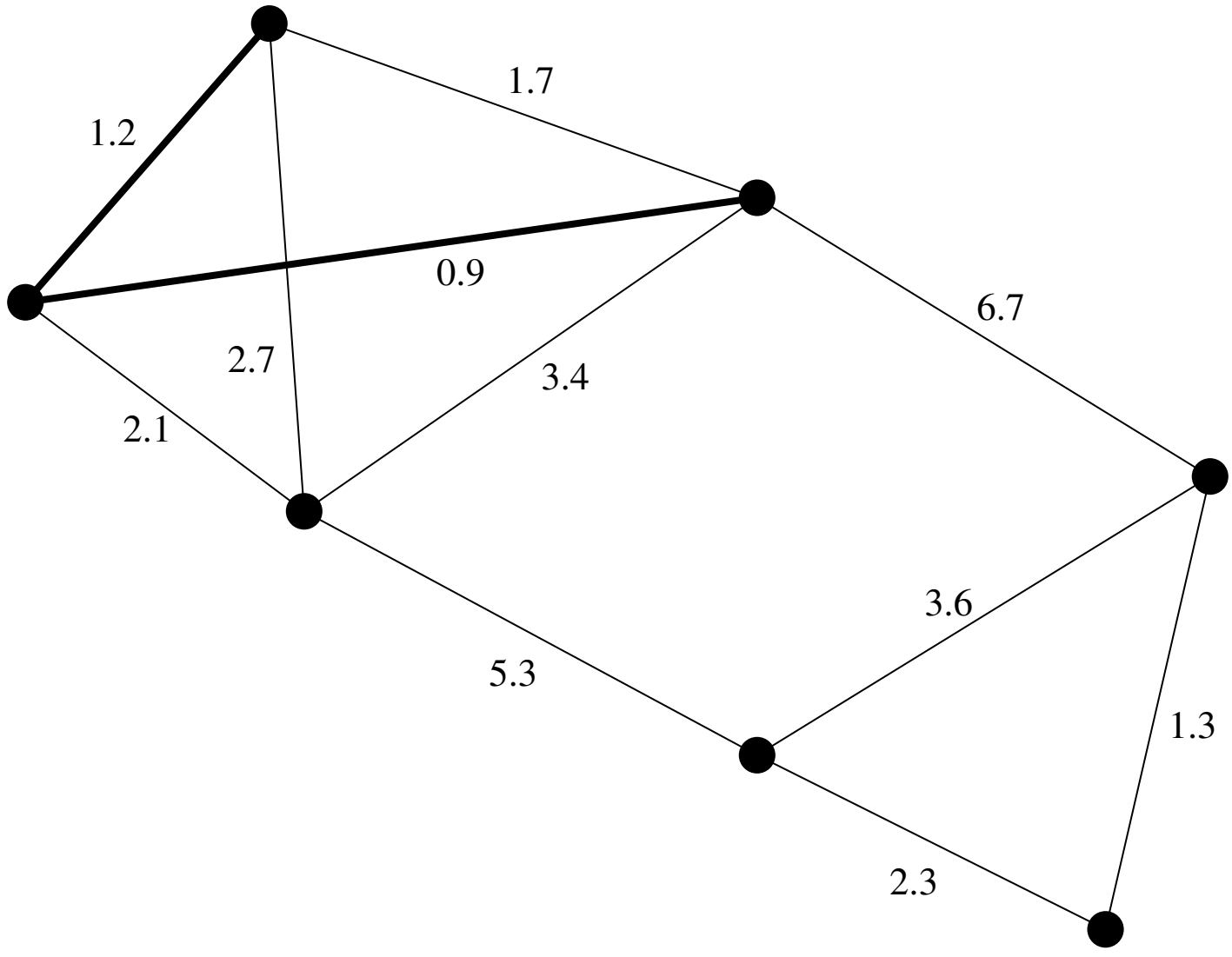
Kruskal's algorithm



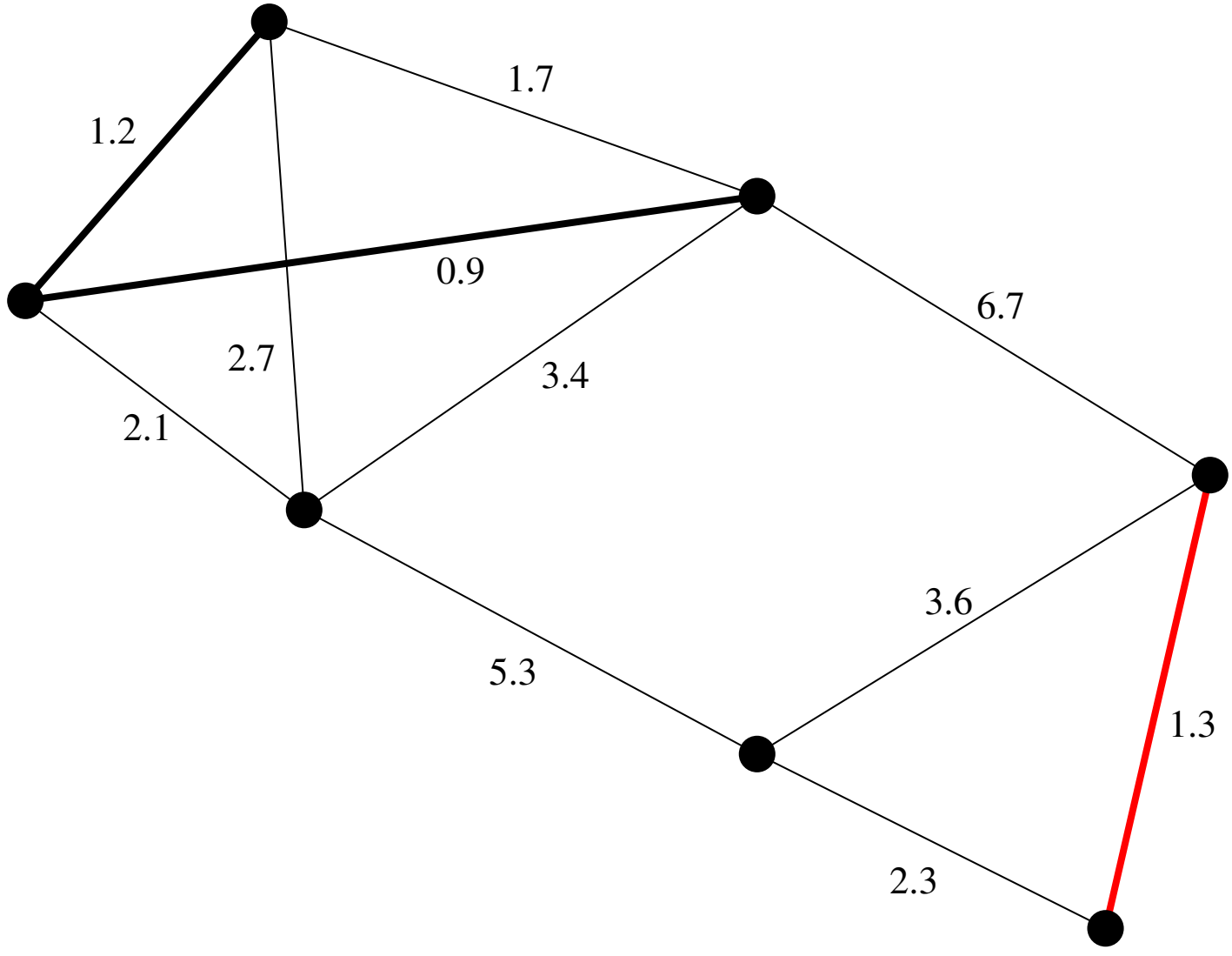
Kruskal's algorithm



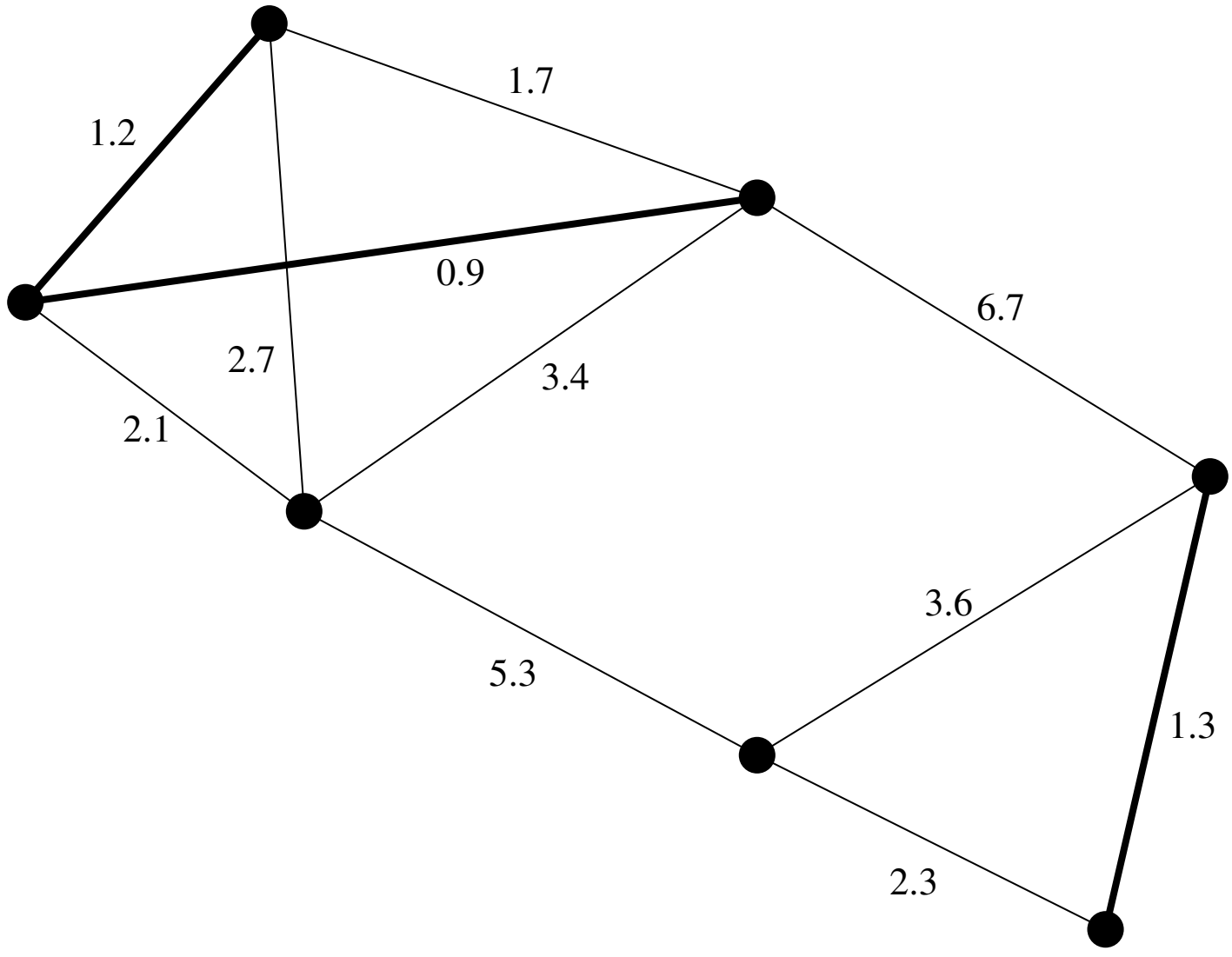
Kruskal's algorithm



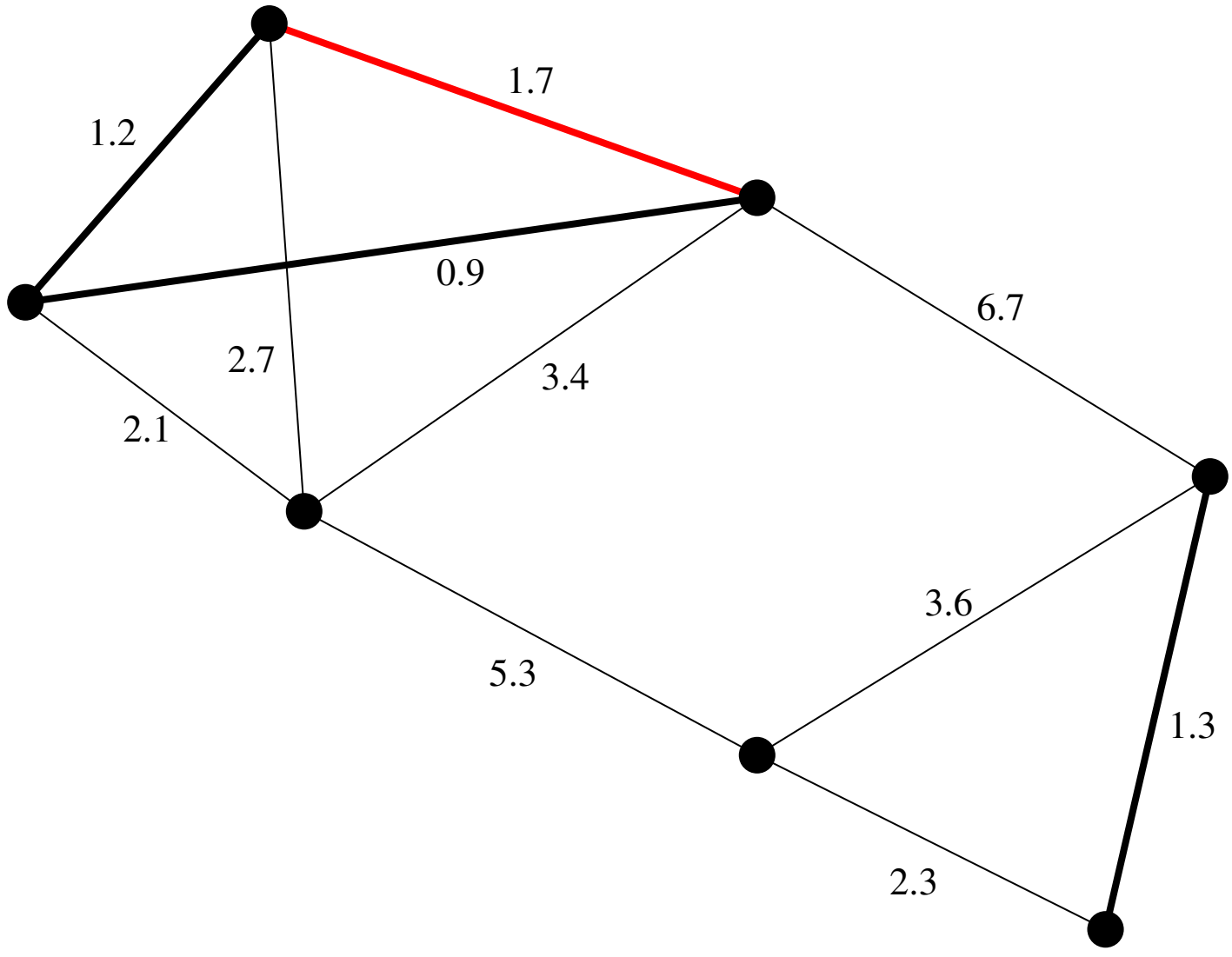
Kruskal's algorithm



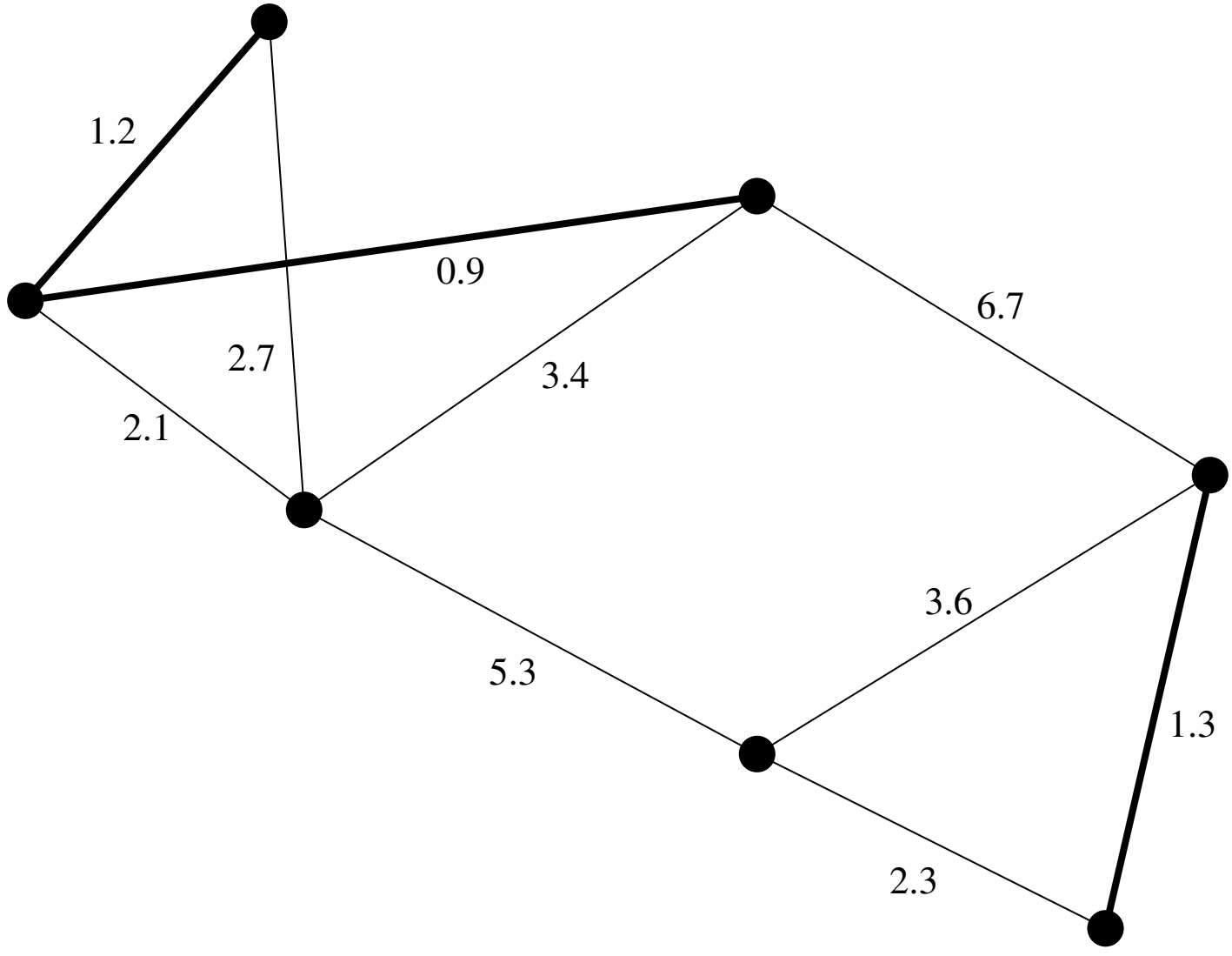
Kruskal's algorithm



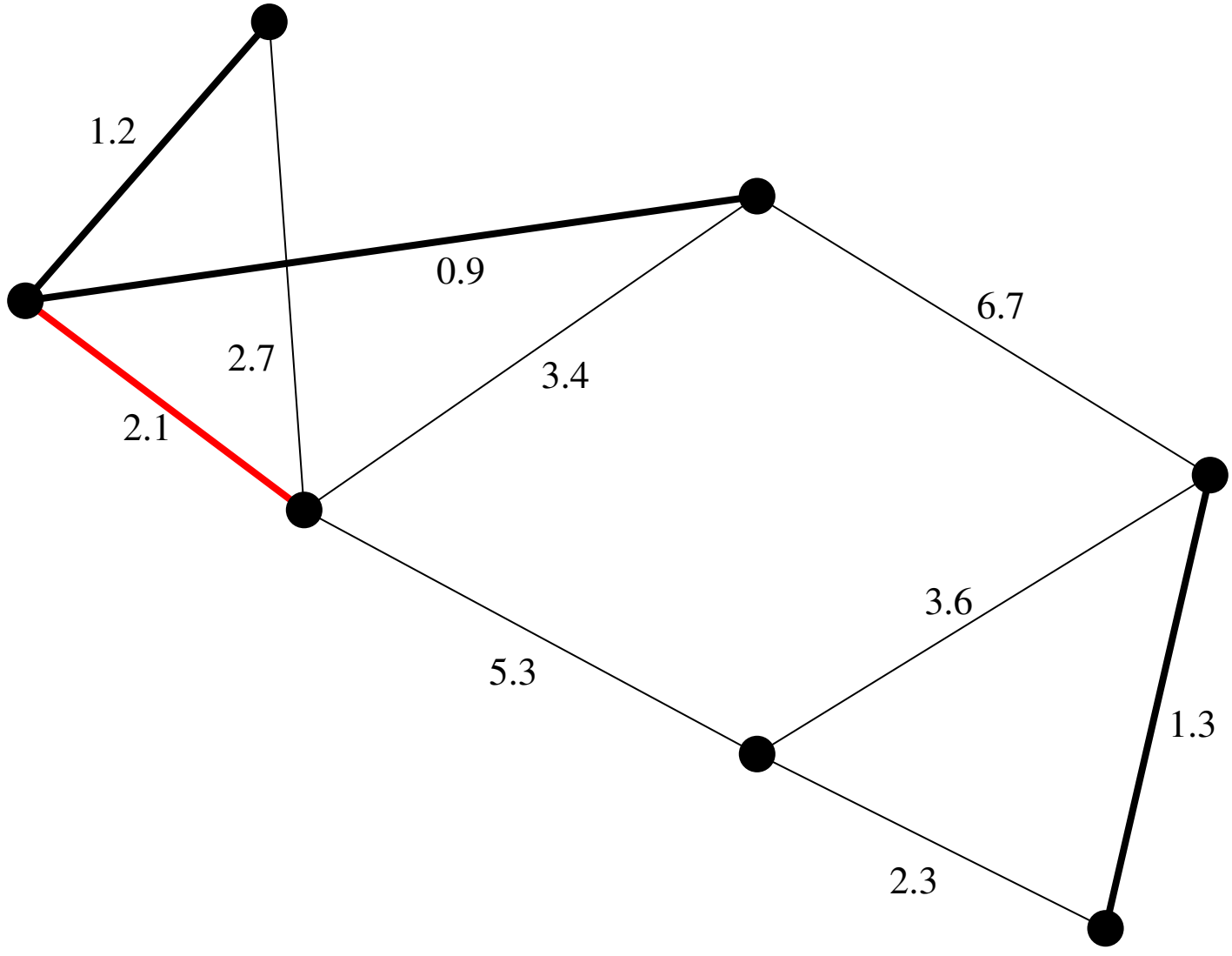
Kruskal's algorithm



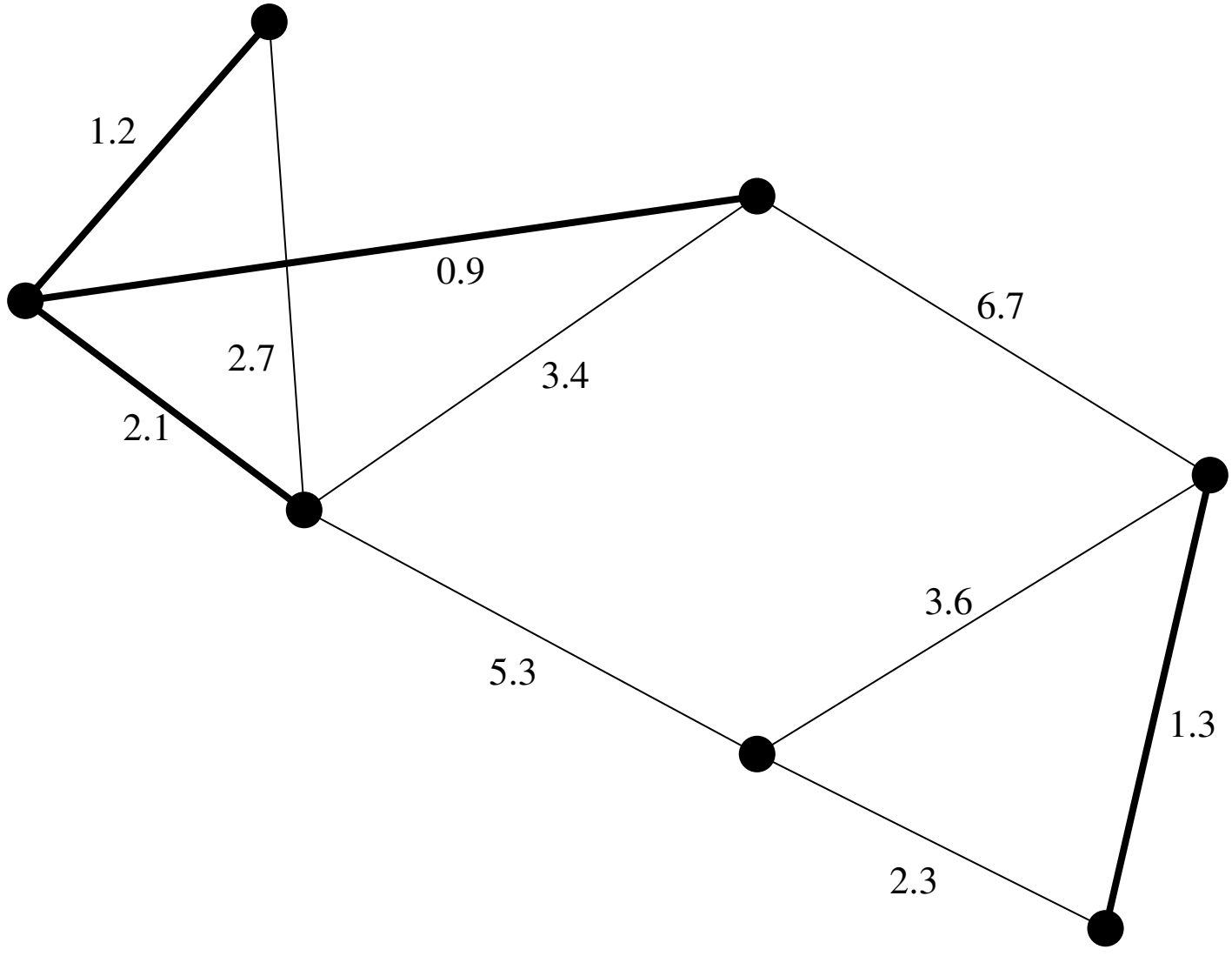
Kruskal's algorithm



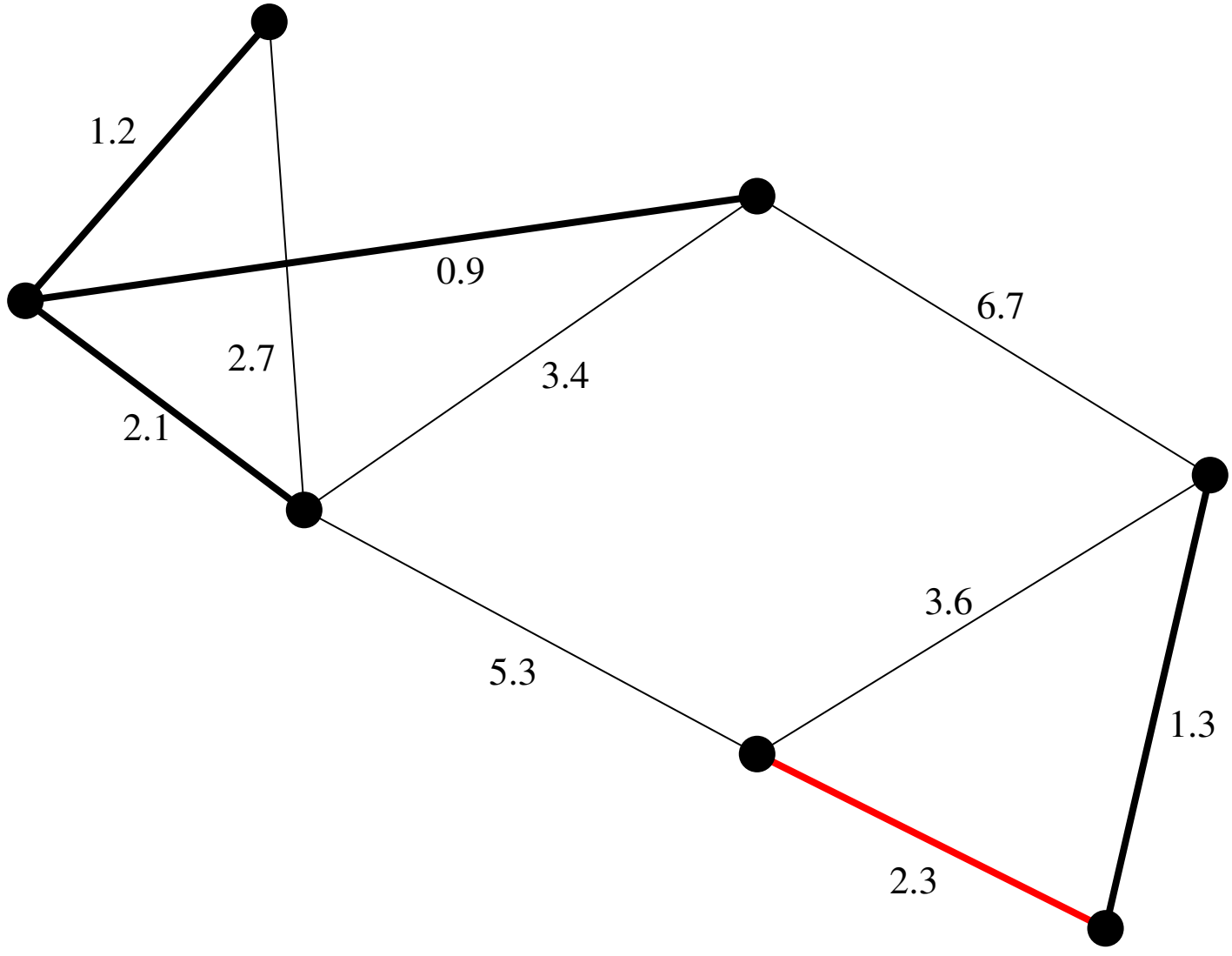
Kruskal's algorithm



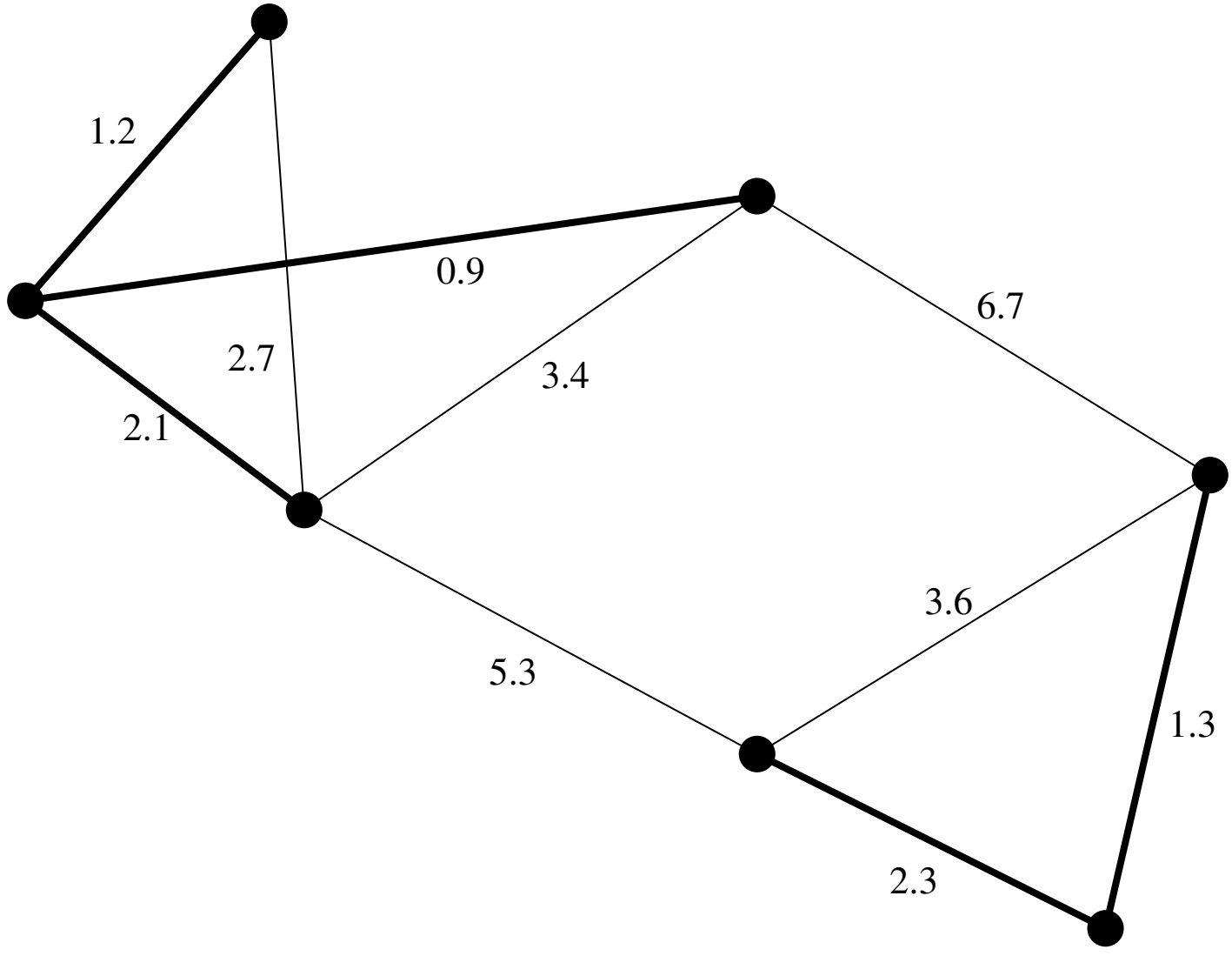
Kruskal's algorithm



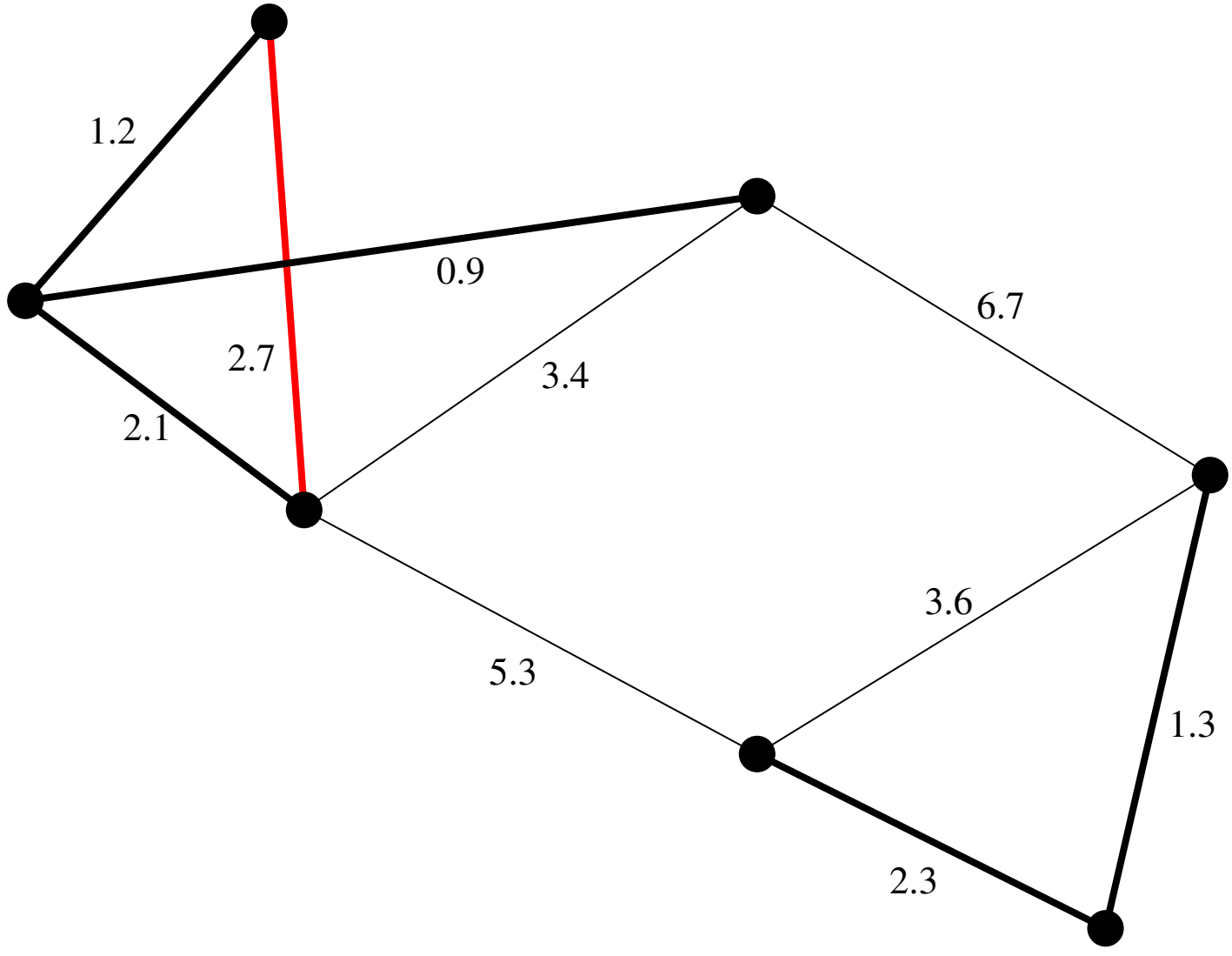
Kruskal's algorithm



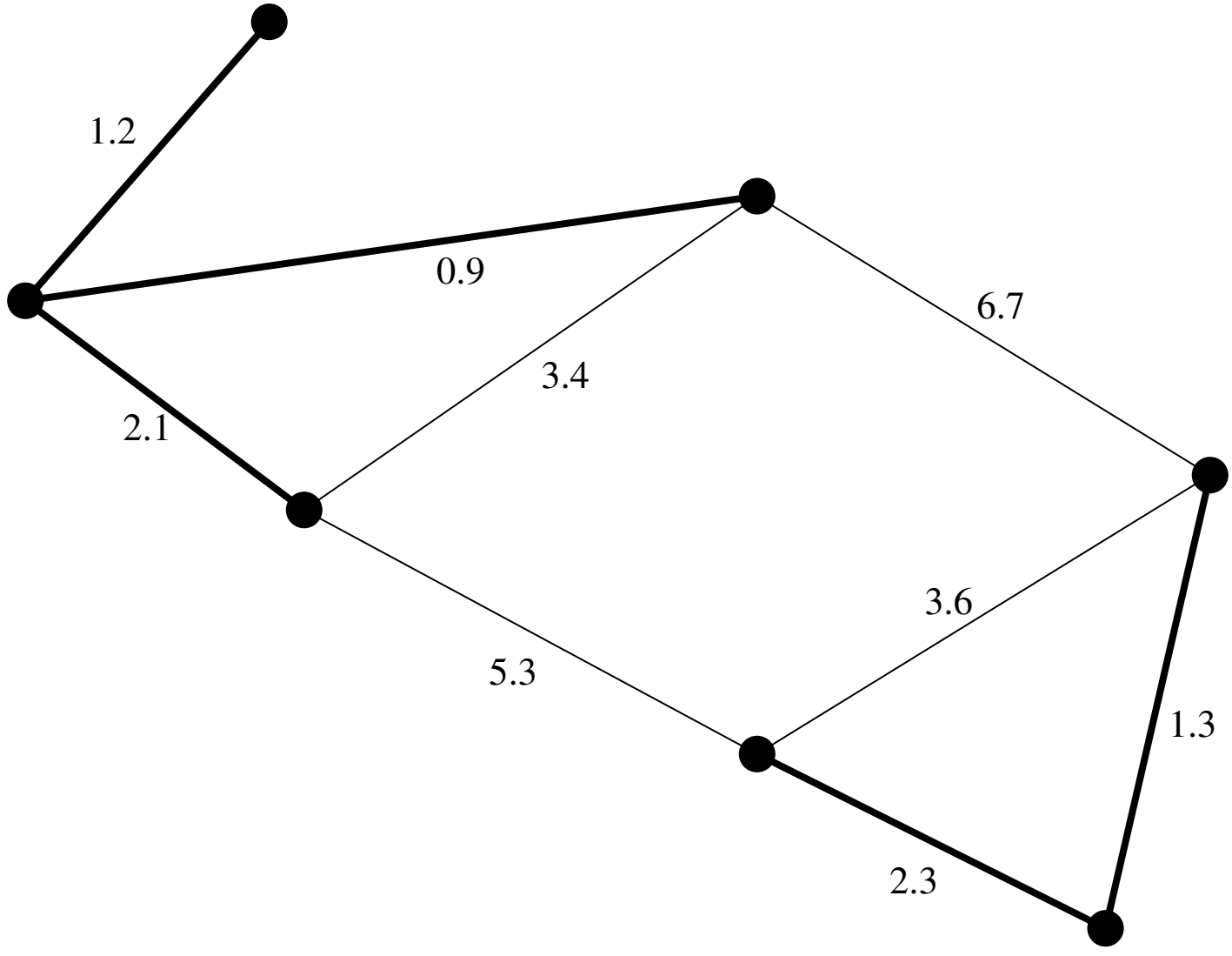
Kruskal's algorithm



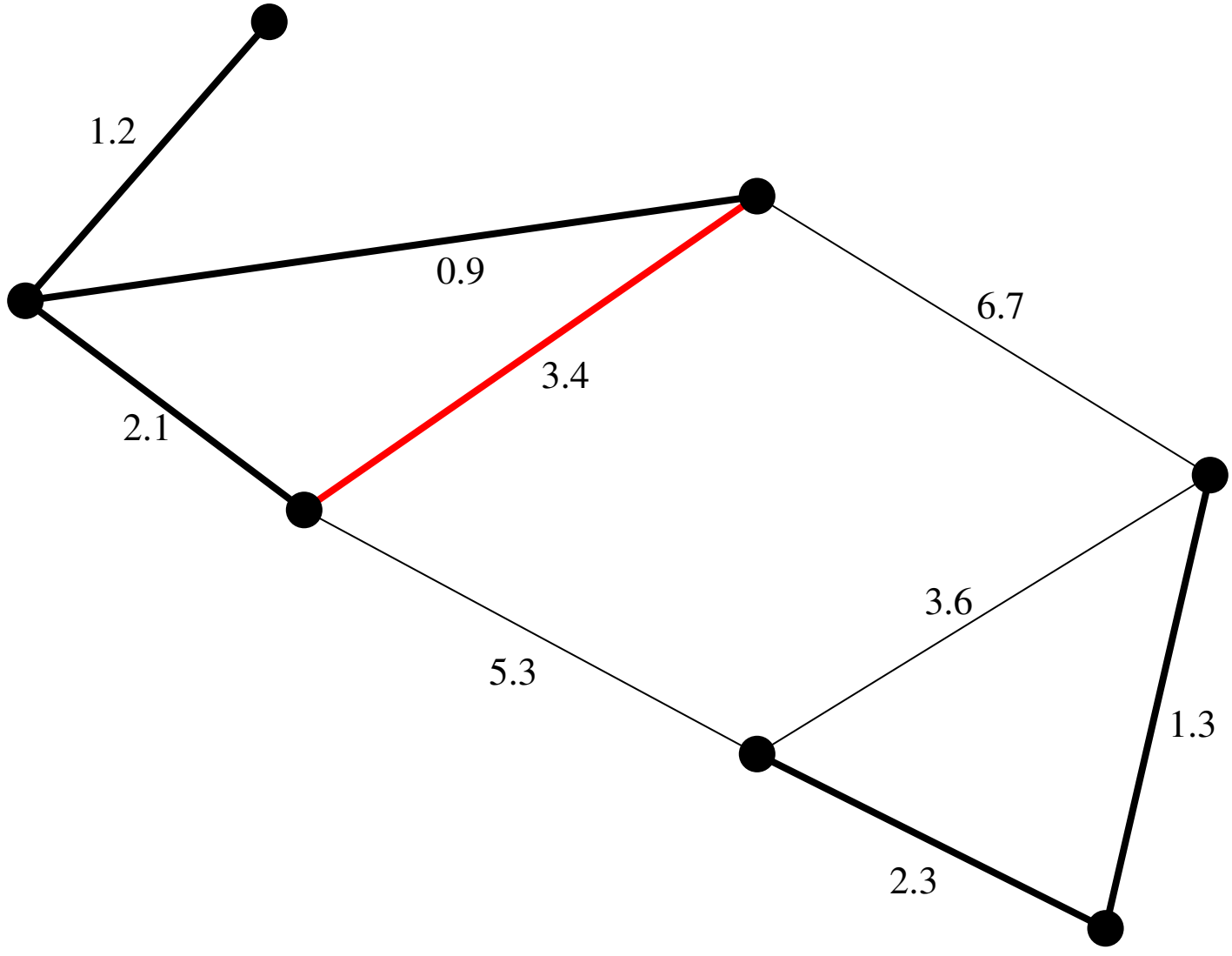
Kruskal's algorithm



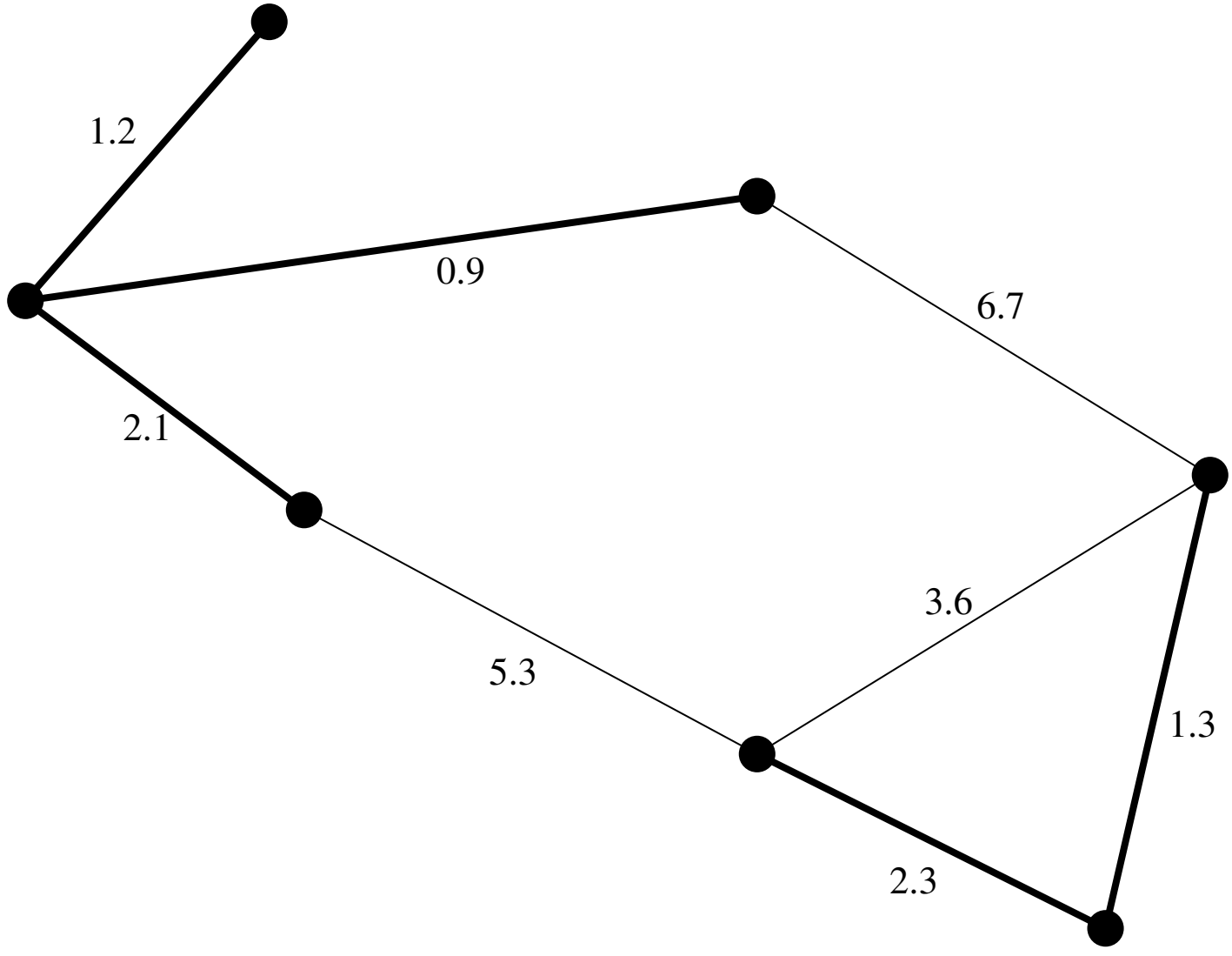
Kruskal's algorithm



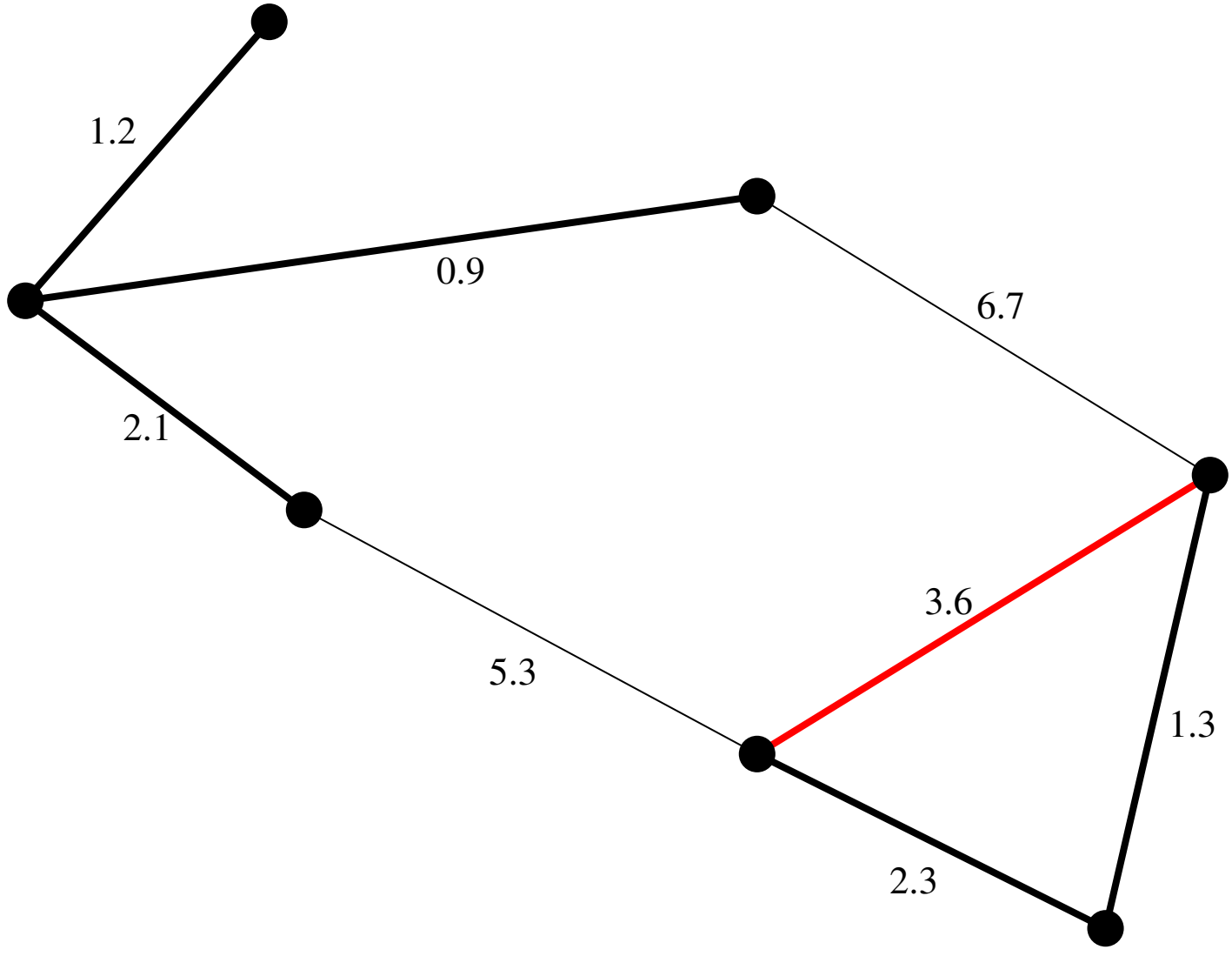
Kruskal's algorithm



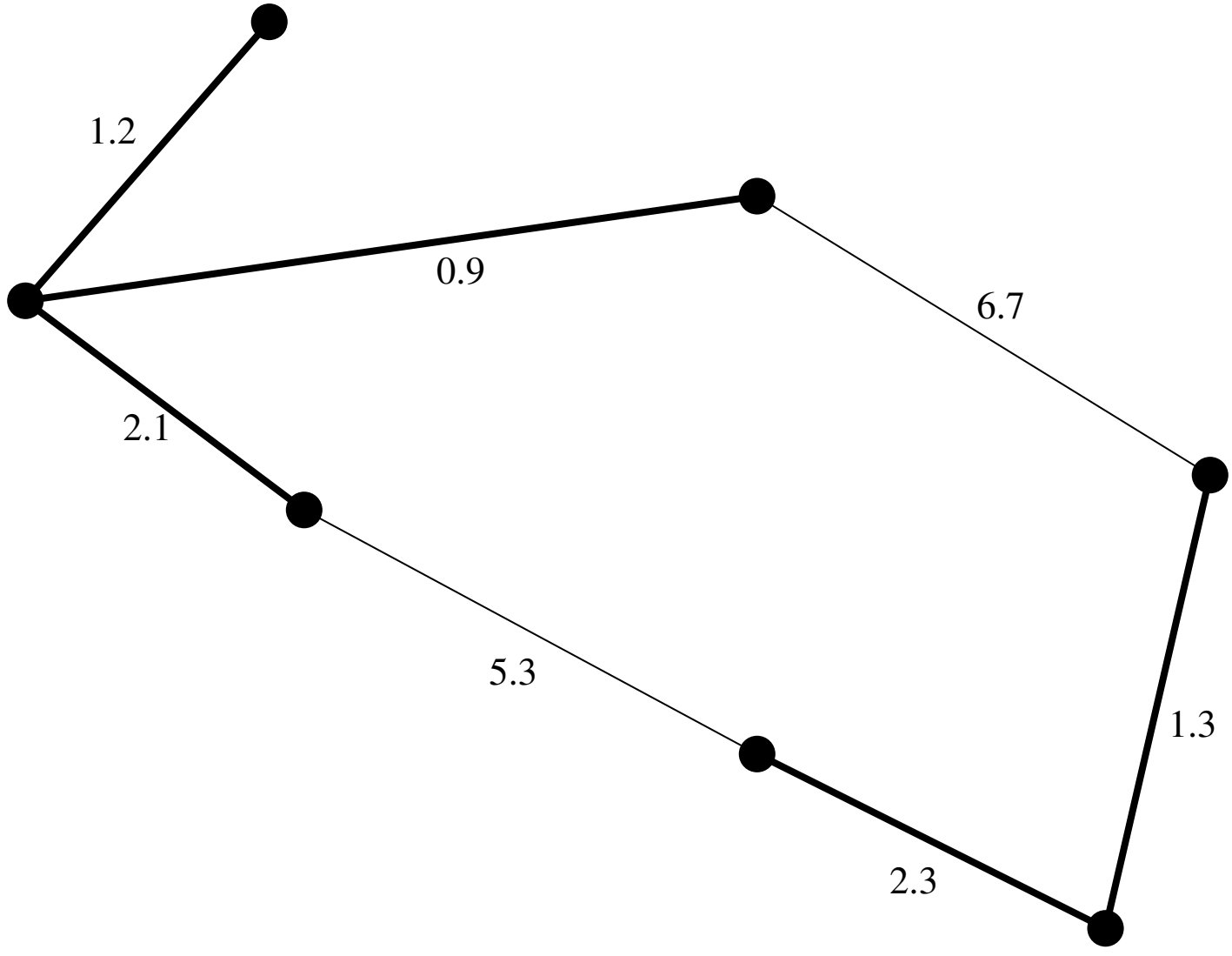
Kruskal's algorithm



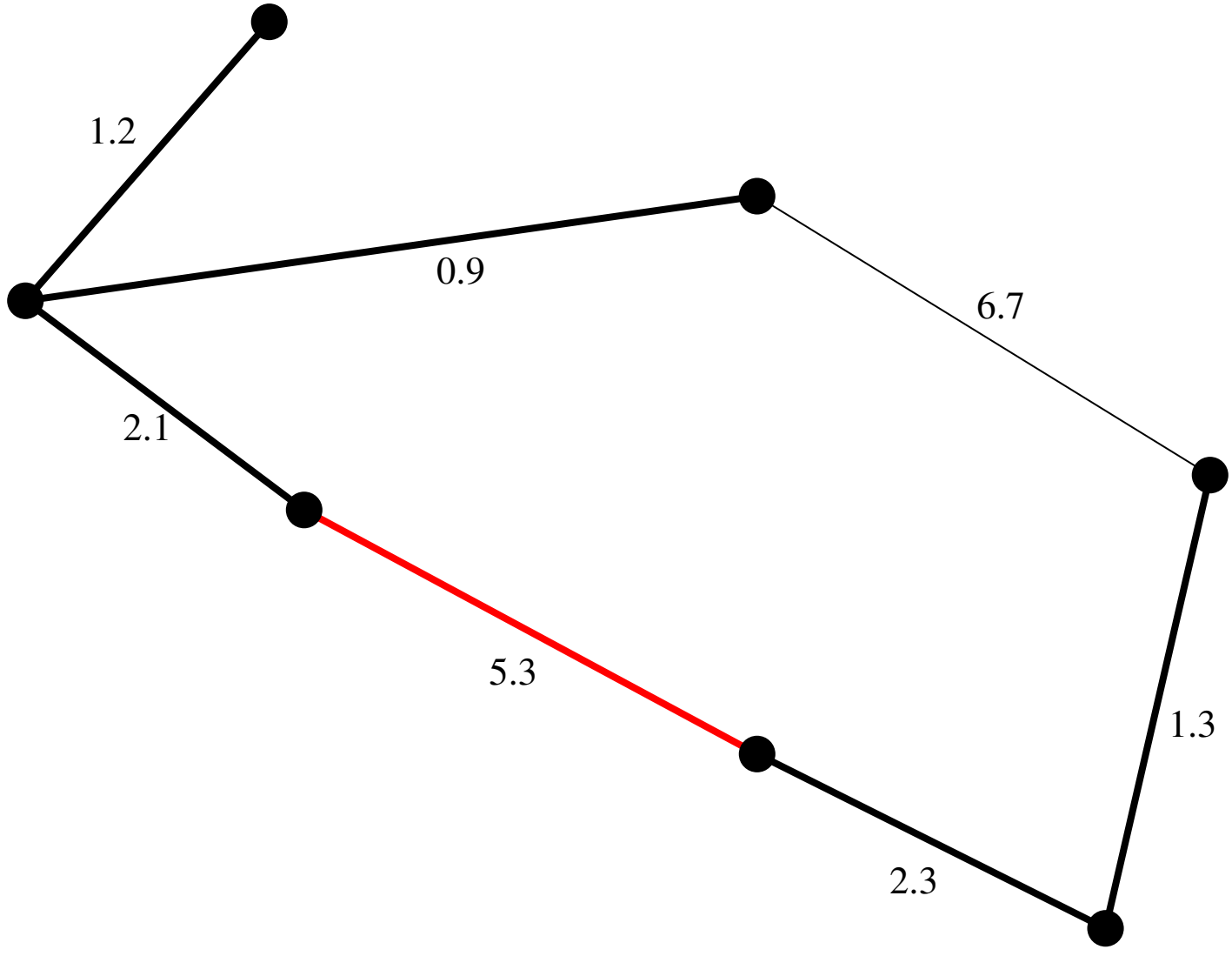
Kruskal's algorithm



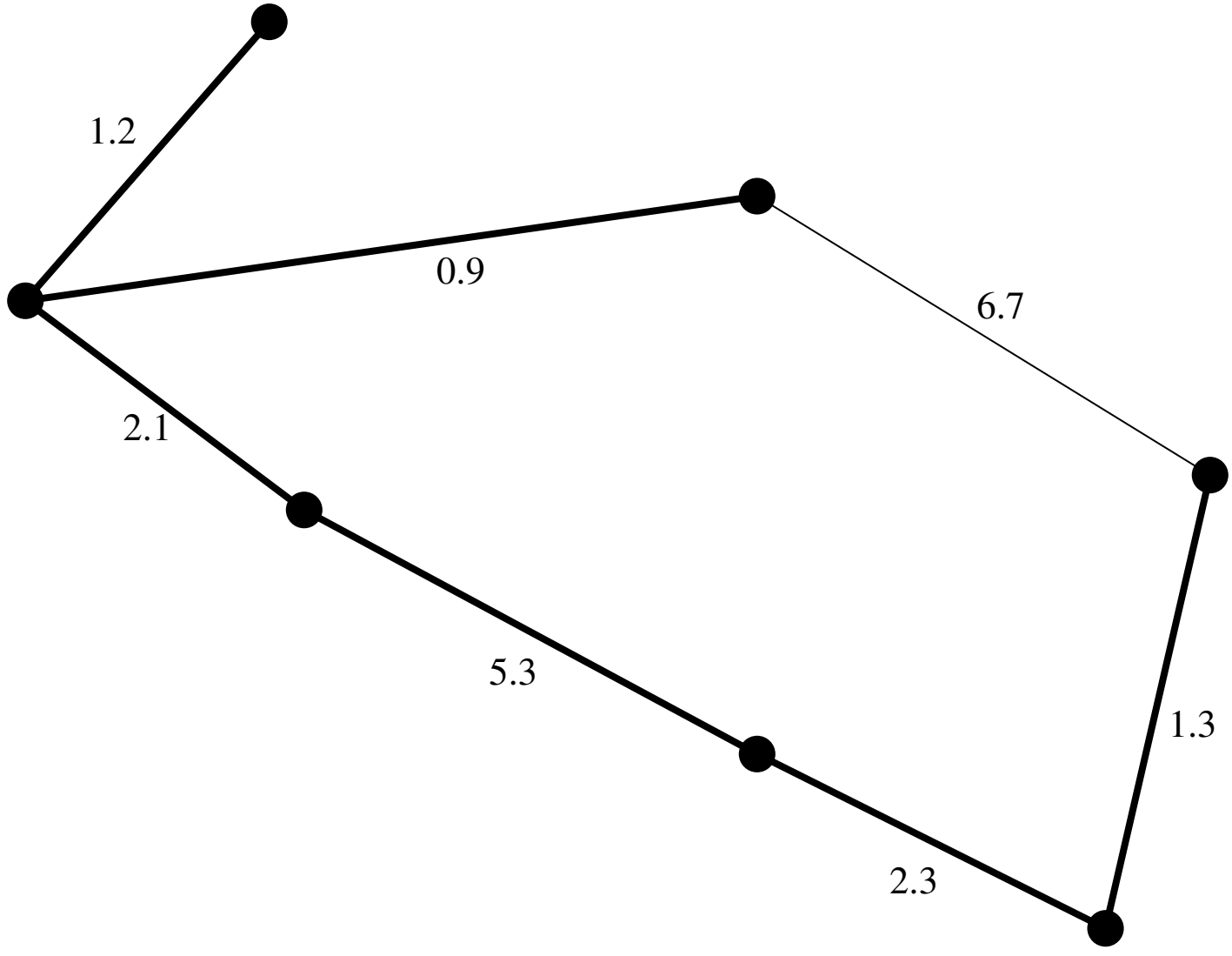
Kruskal's algorithm



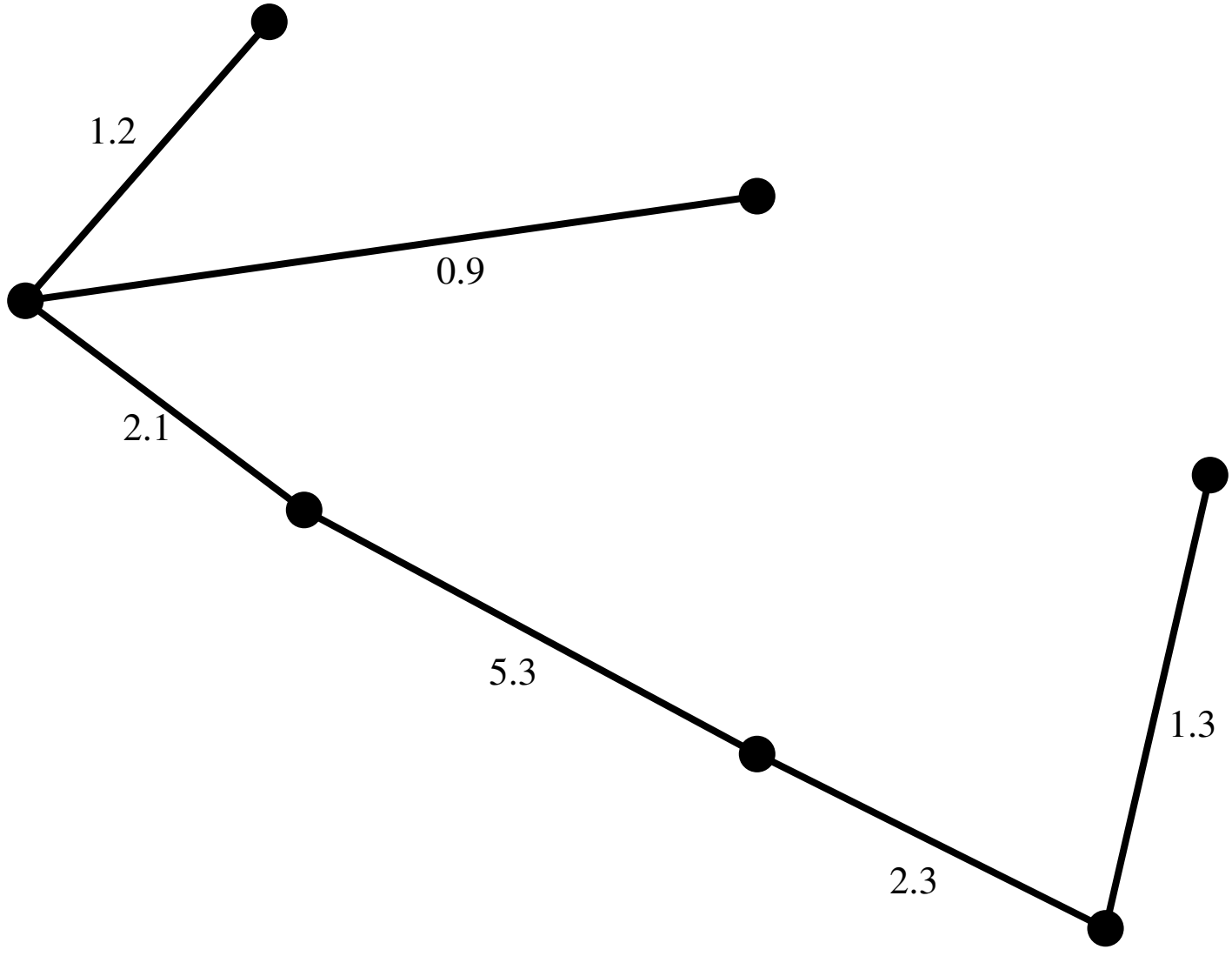
Kruskal's algorithm



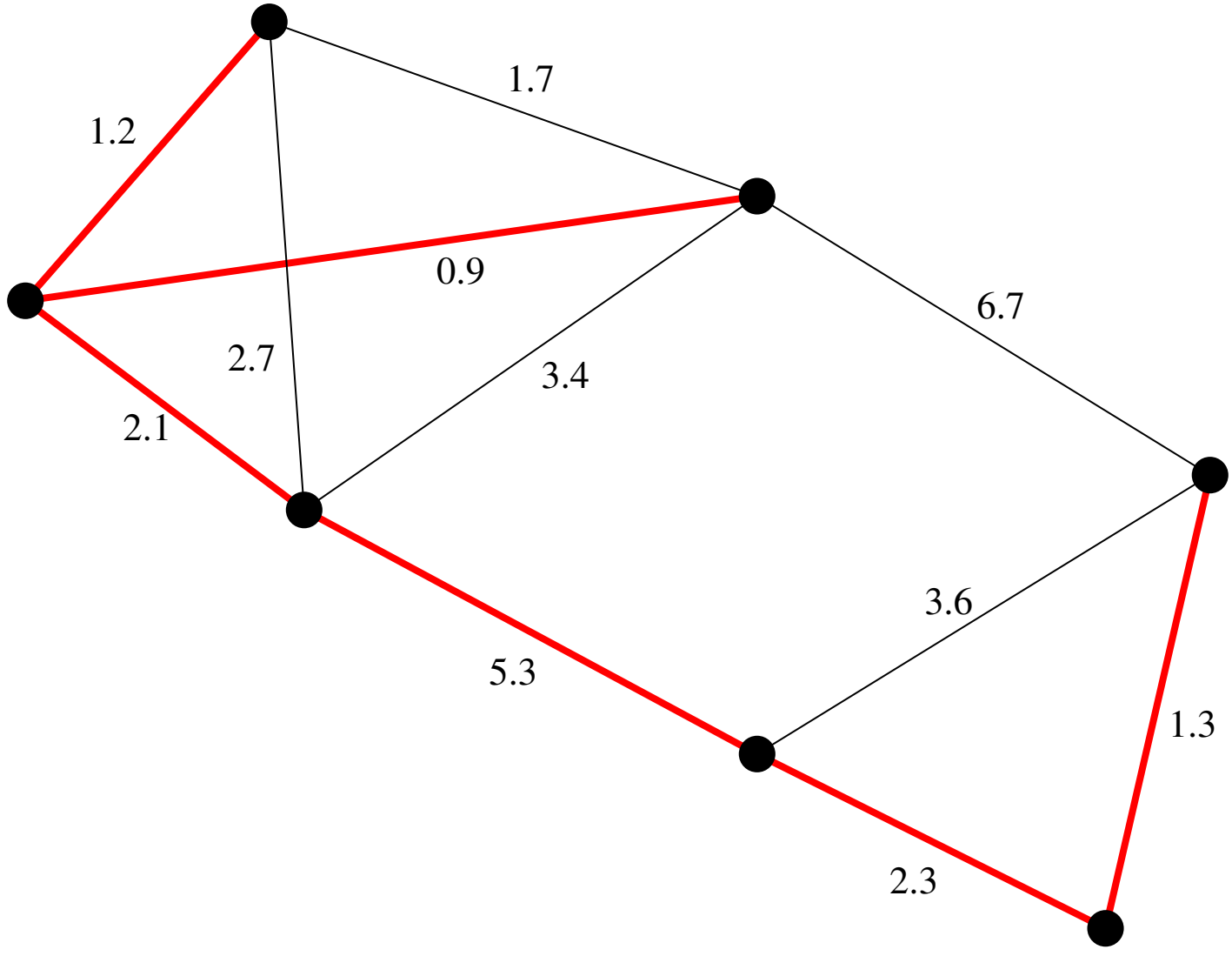
Kruskal's algorithm



Kruskal's algorithm



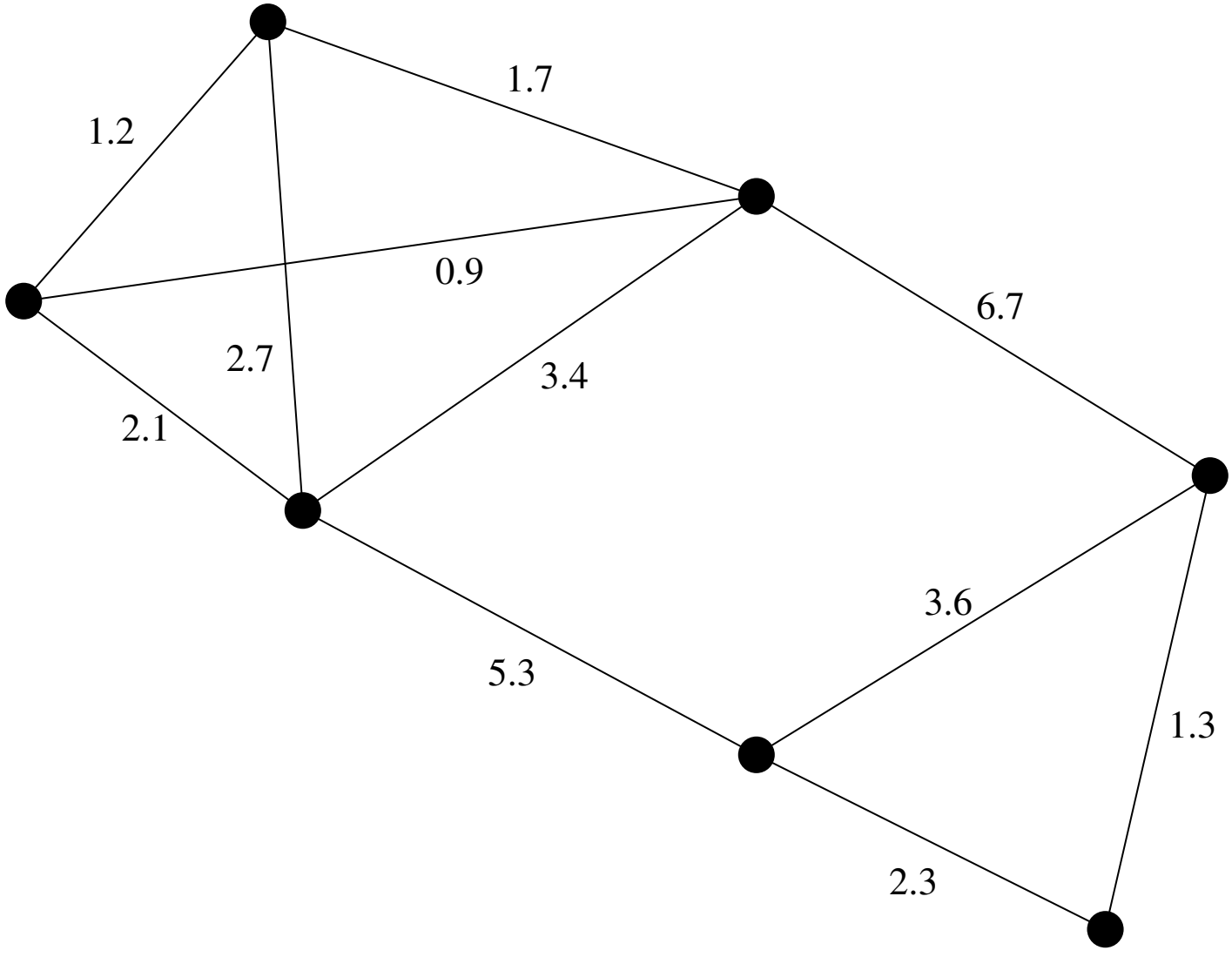
Kruskal's algorithm



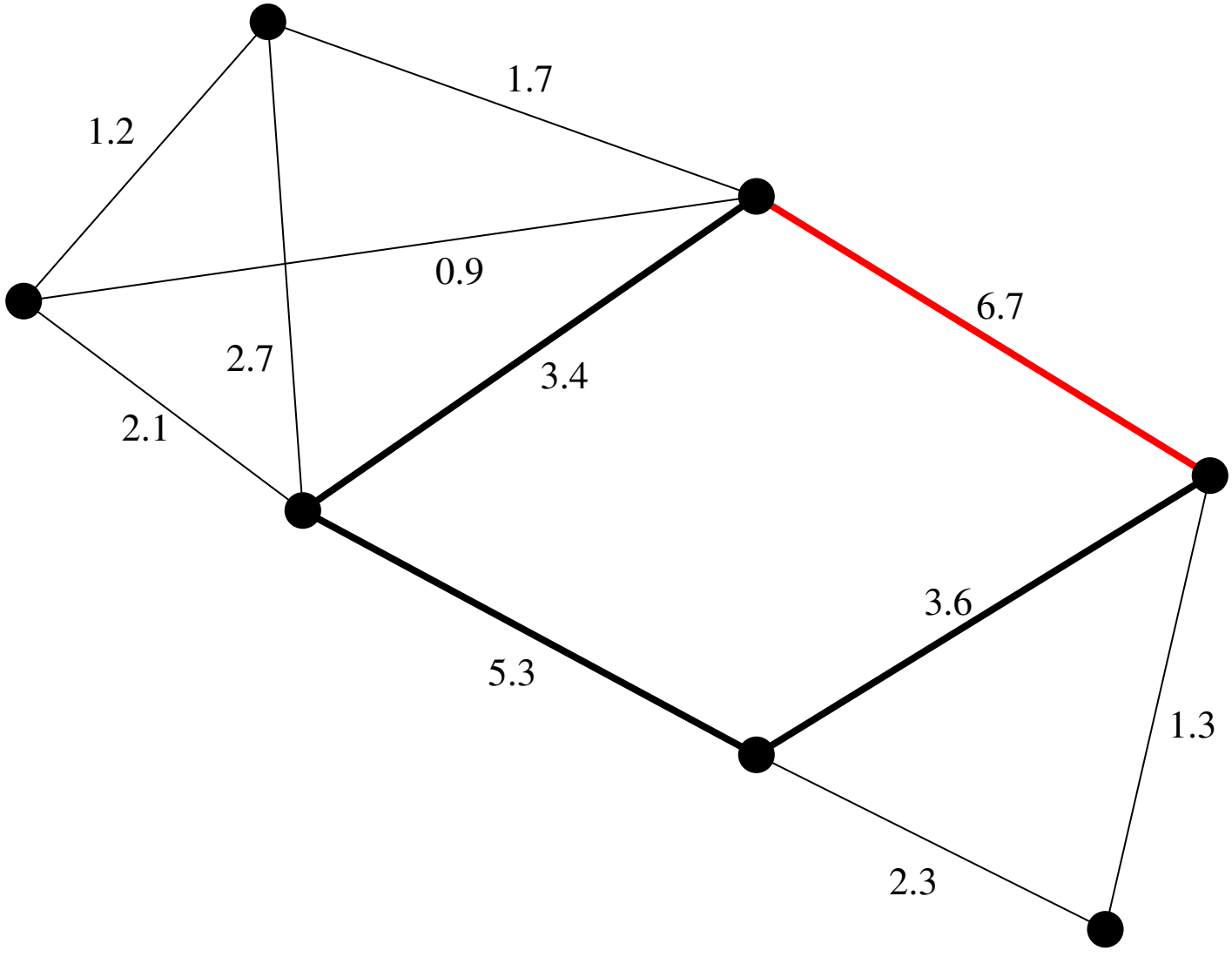
Cycle-breaking algorithm

- Start with all edges present. List the edges as $e_1, e_2, \ldots$ in decreasing order of weight.
- At step i, remove e_i as long as it does not disconnect the graph (i.e. as long as it lies in a cycle).
- Stop when no cycles remain.

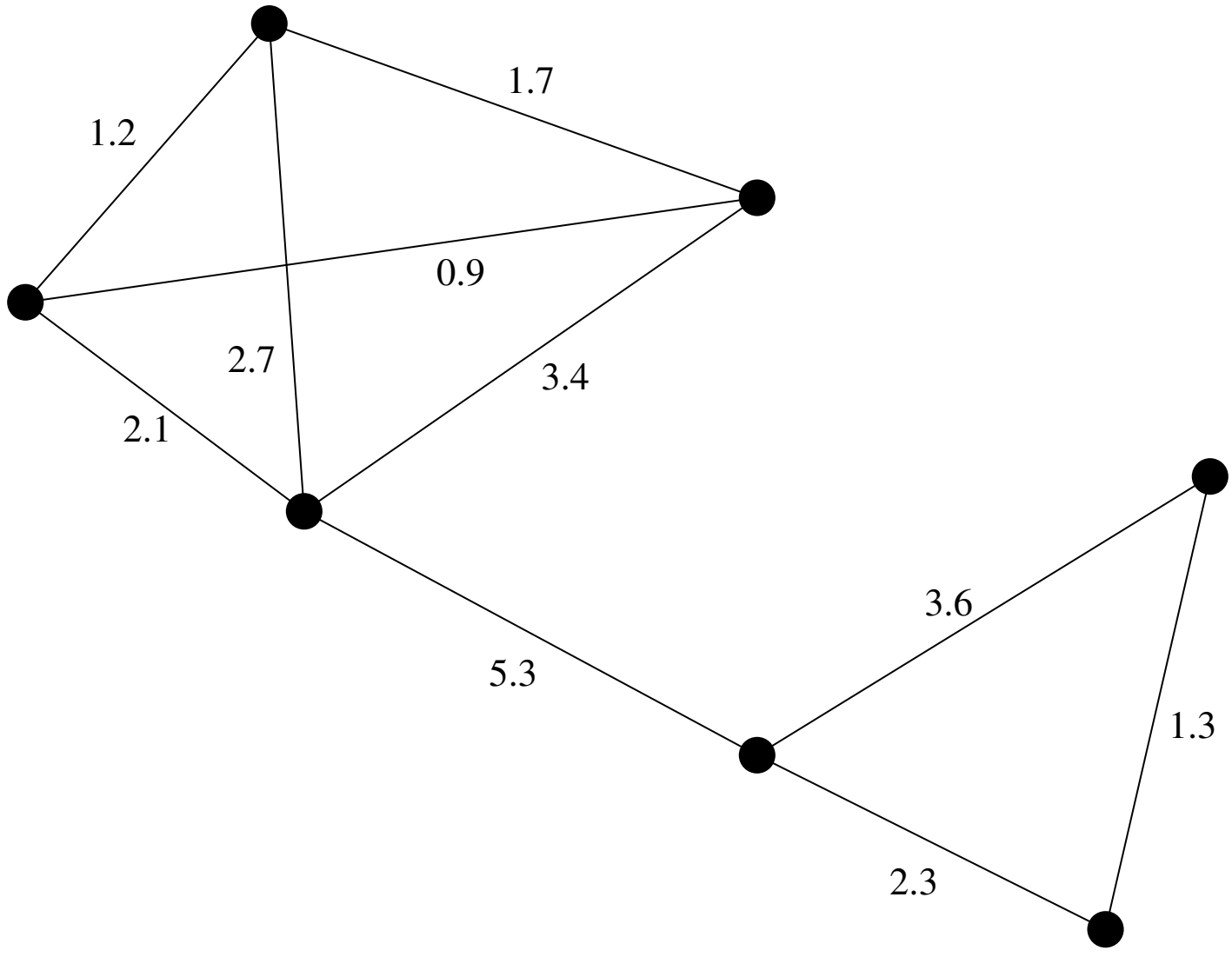
Cycle-breaking algorithm



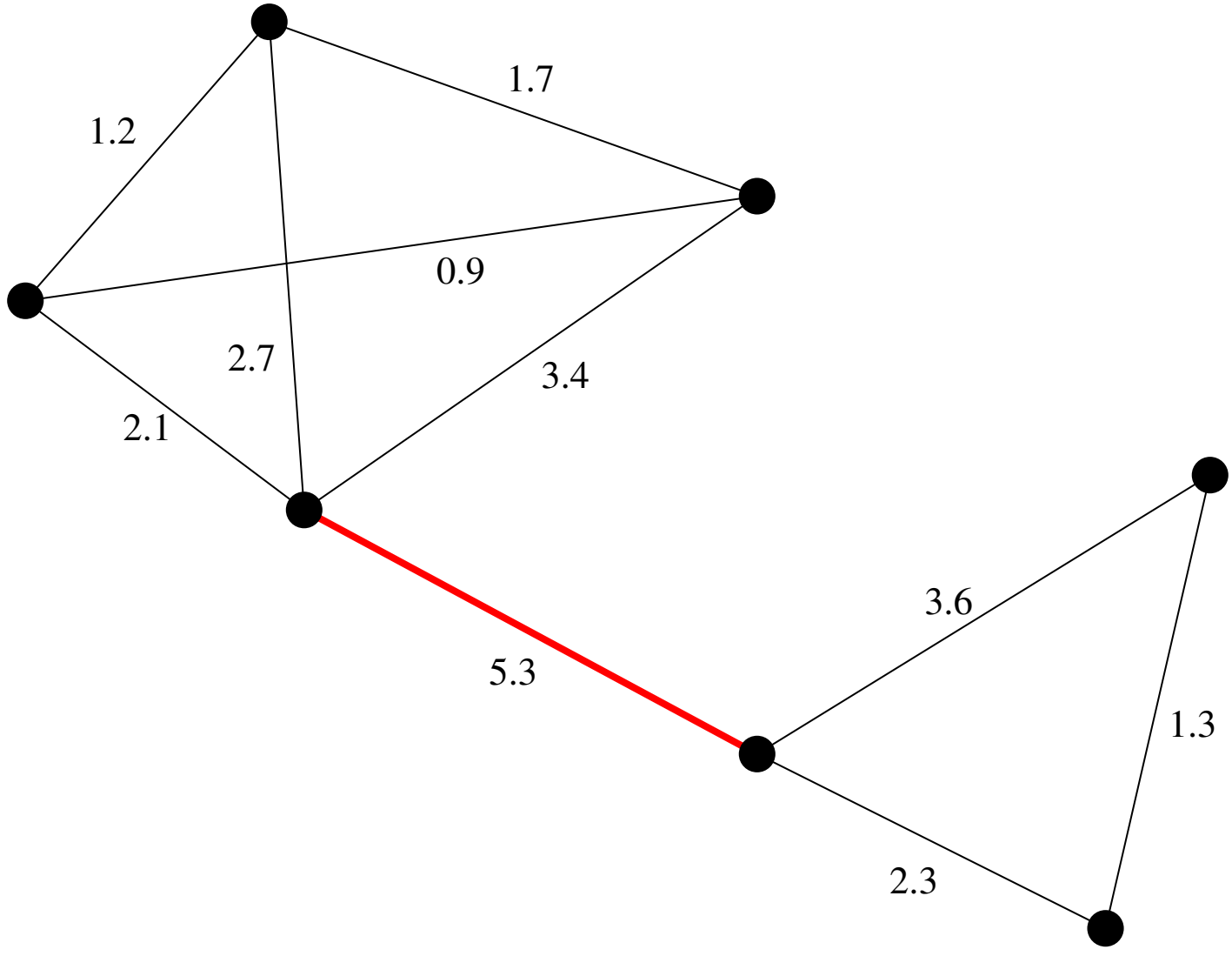
Cycle-breaking algorithm



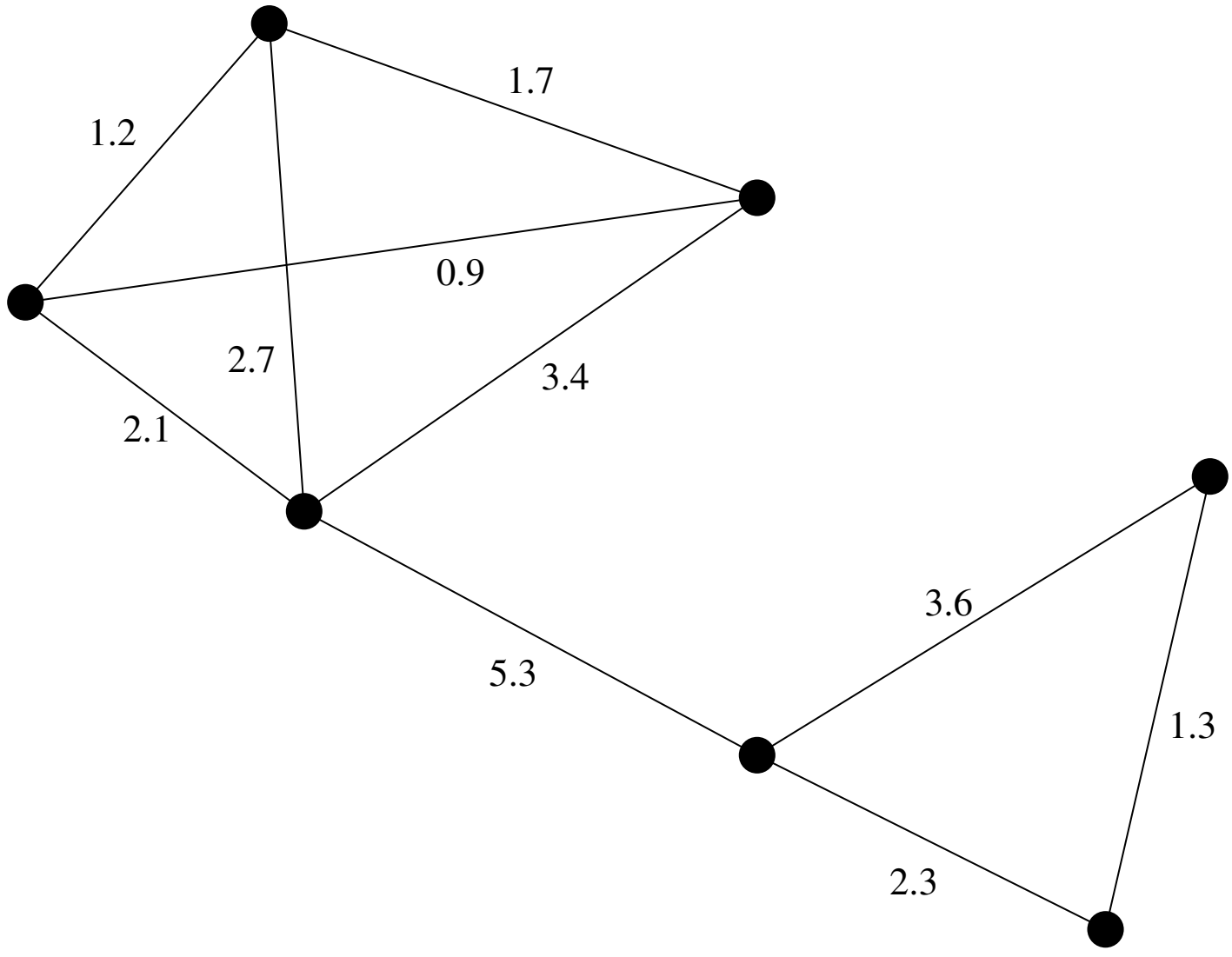
Cycle-breaking algorithm



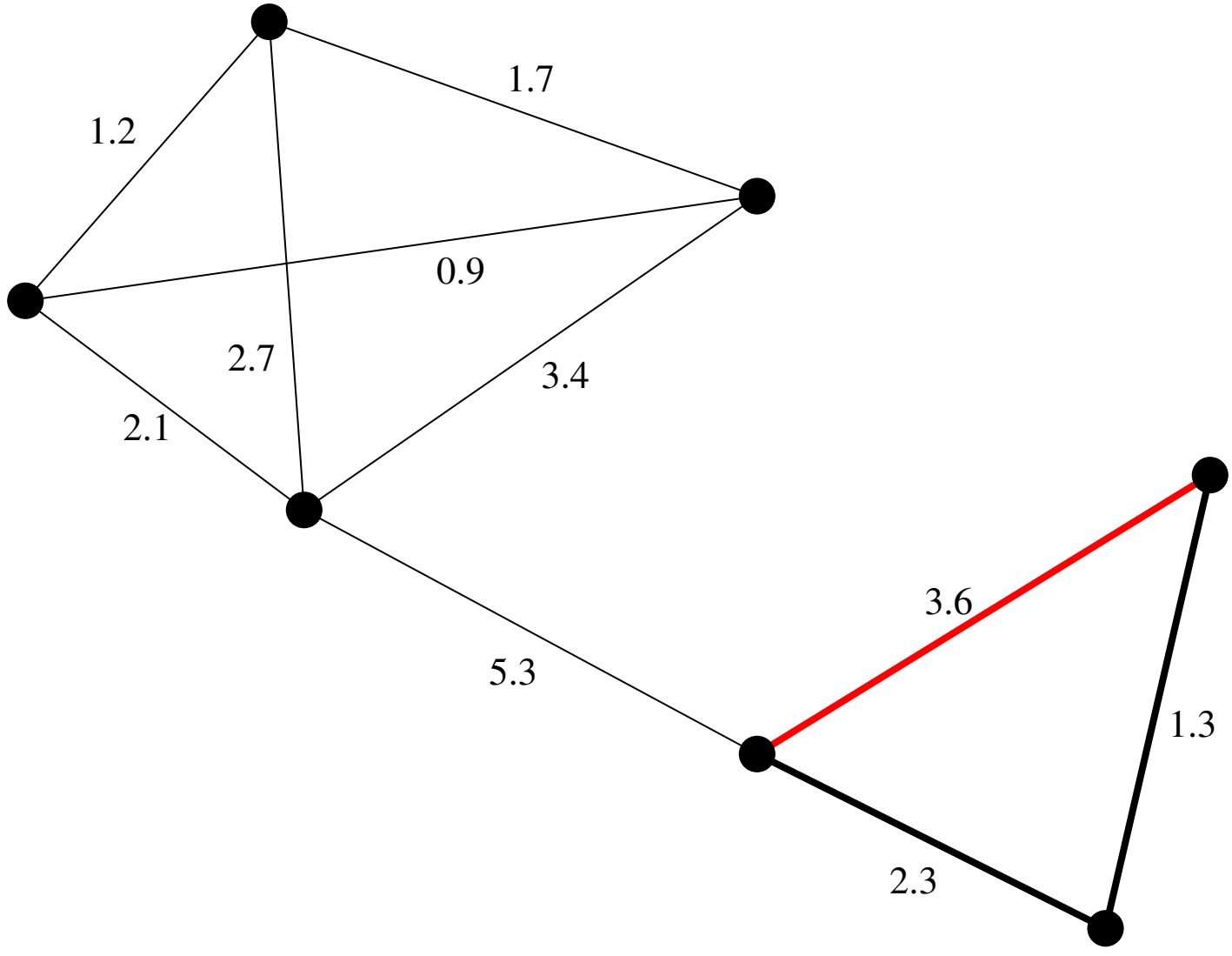
Cycle-breaking algorithm



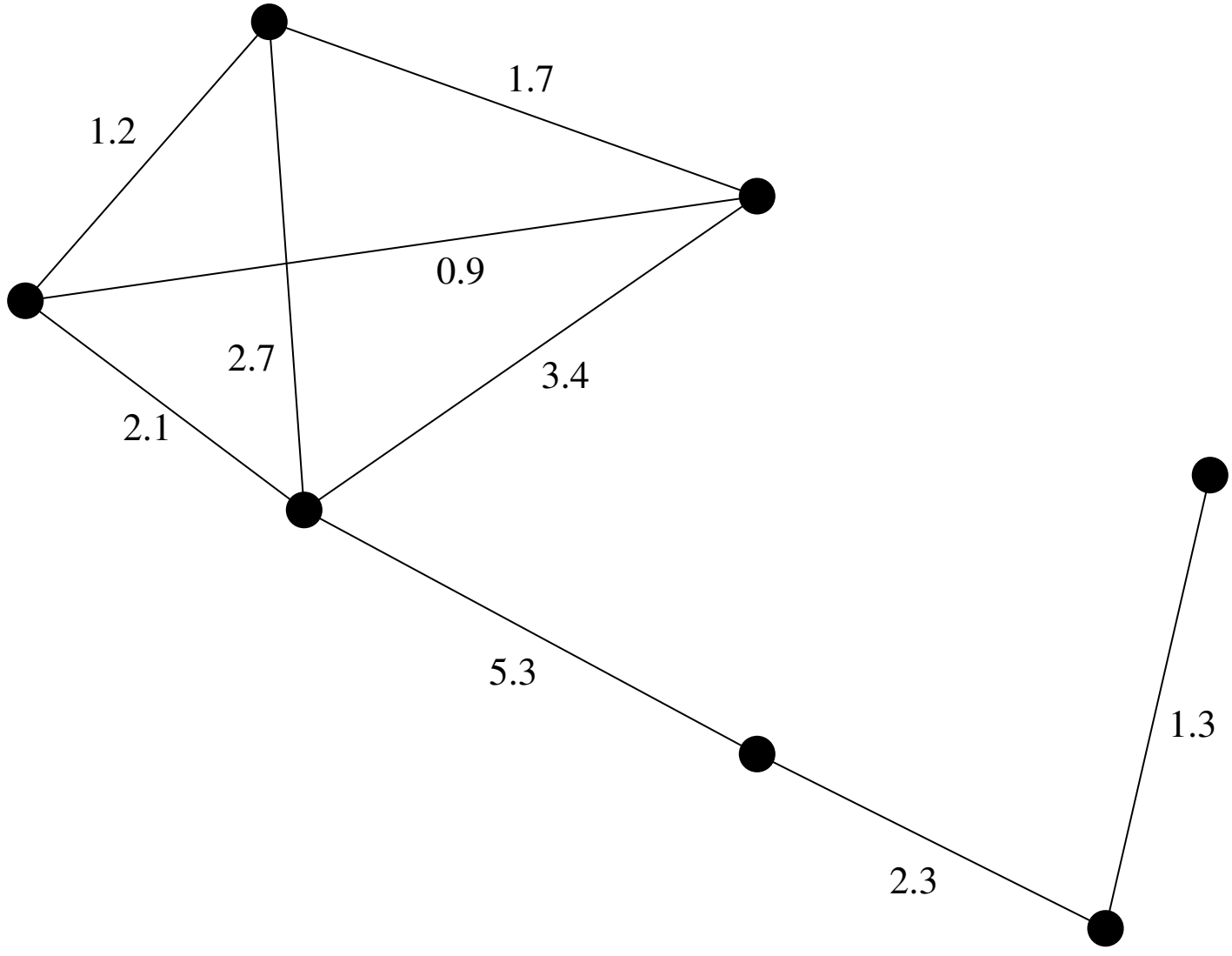
Cycle-breaking algorithm



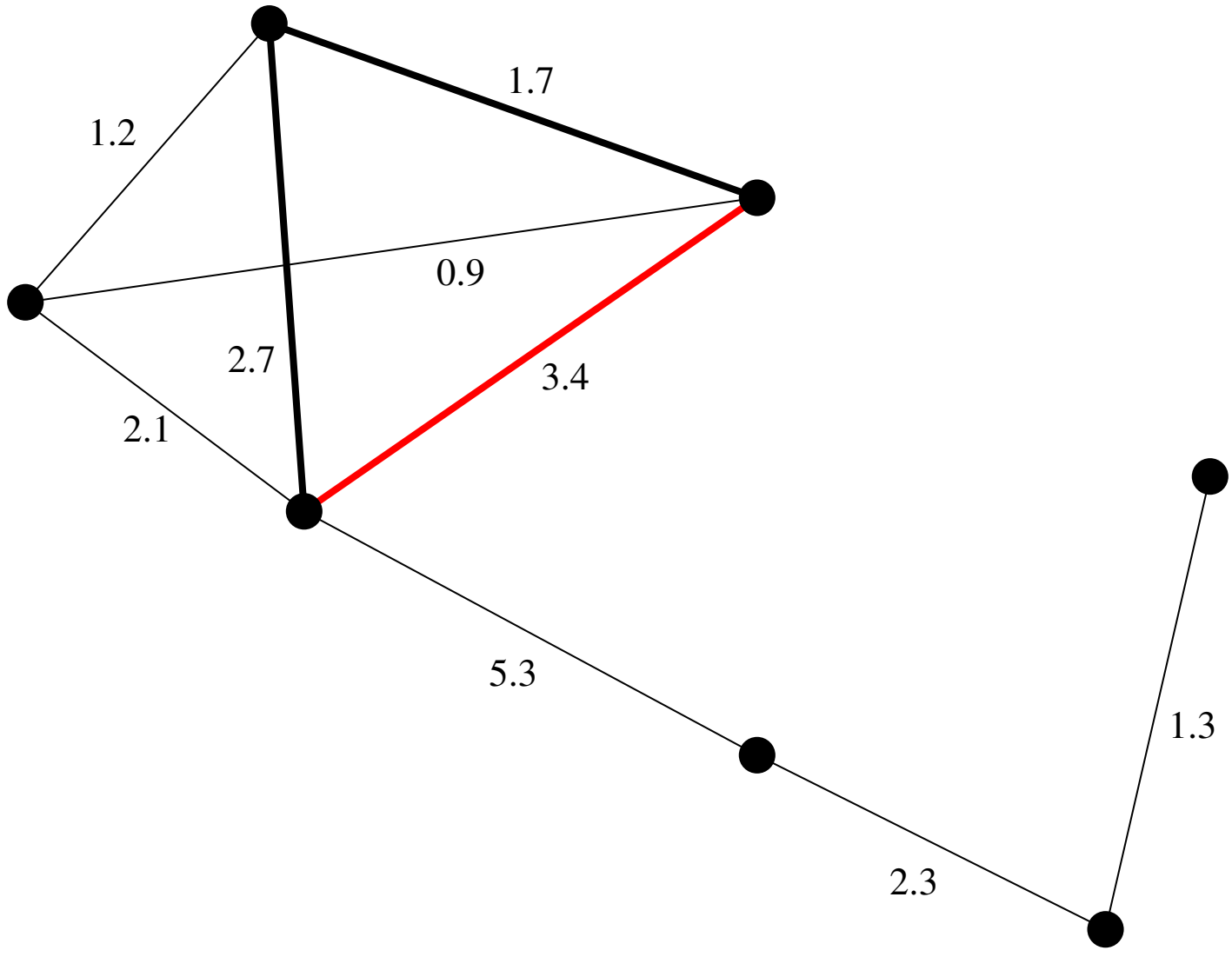
Cycle-breaking algorithm



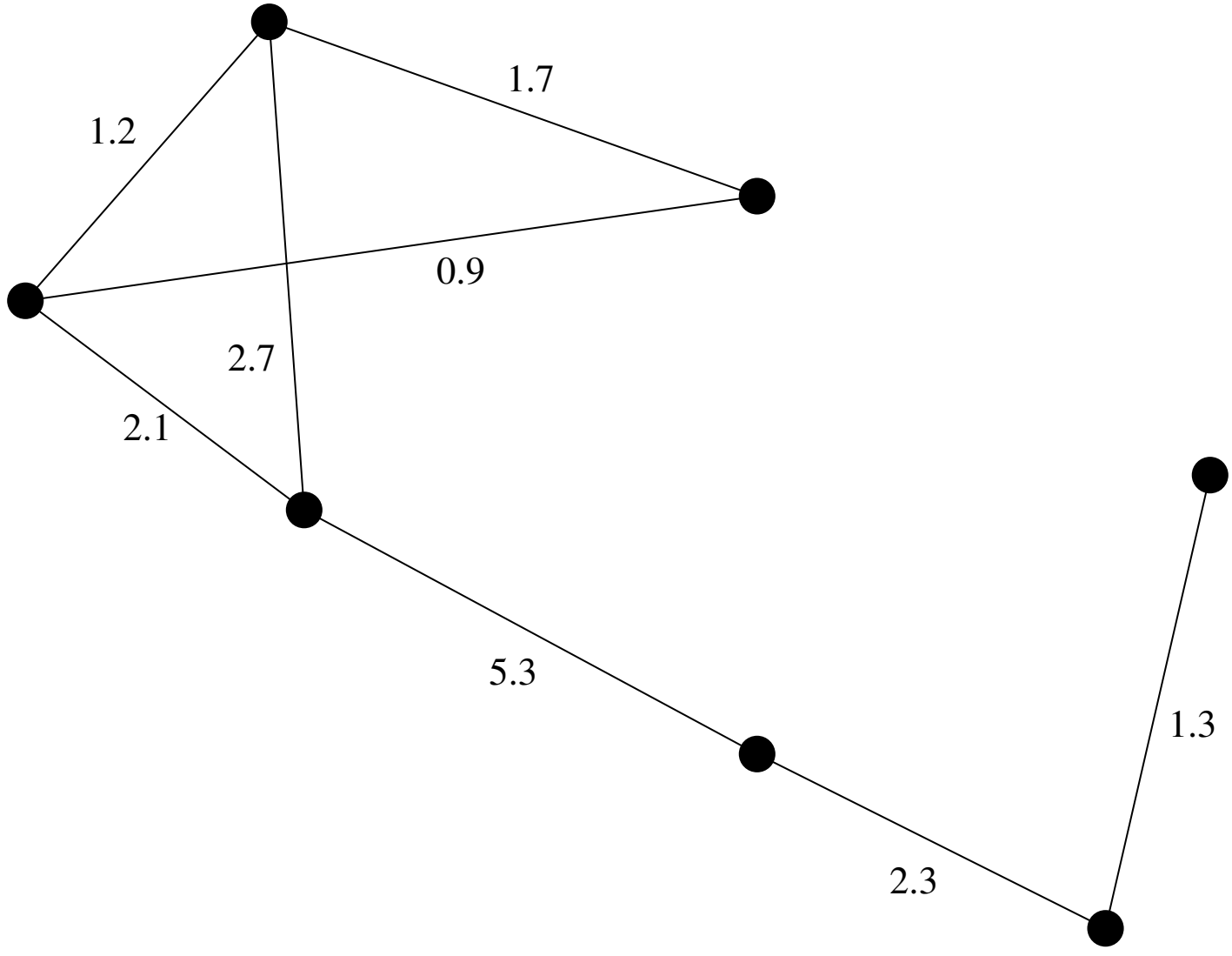
Cycle-breaking algorithm



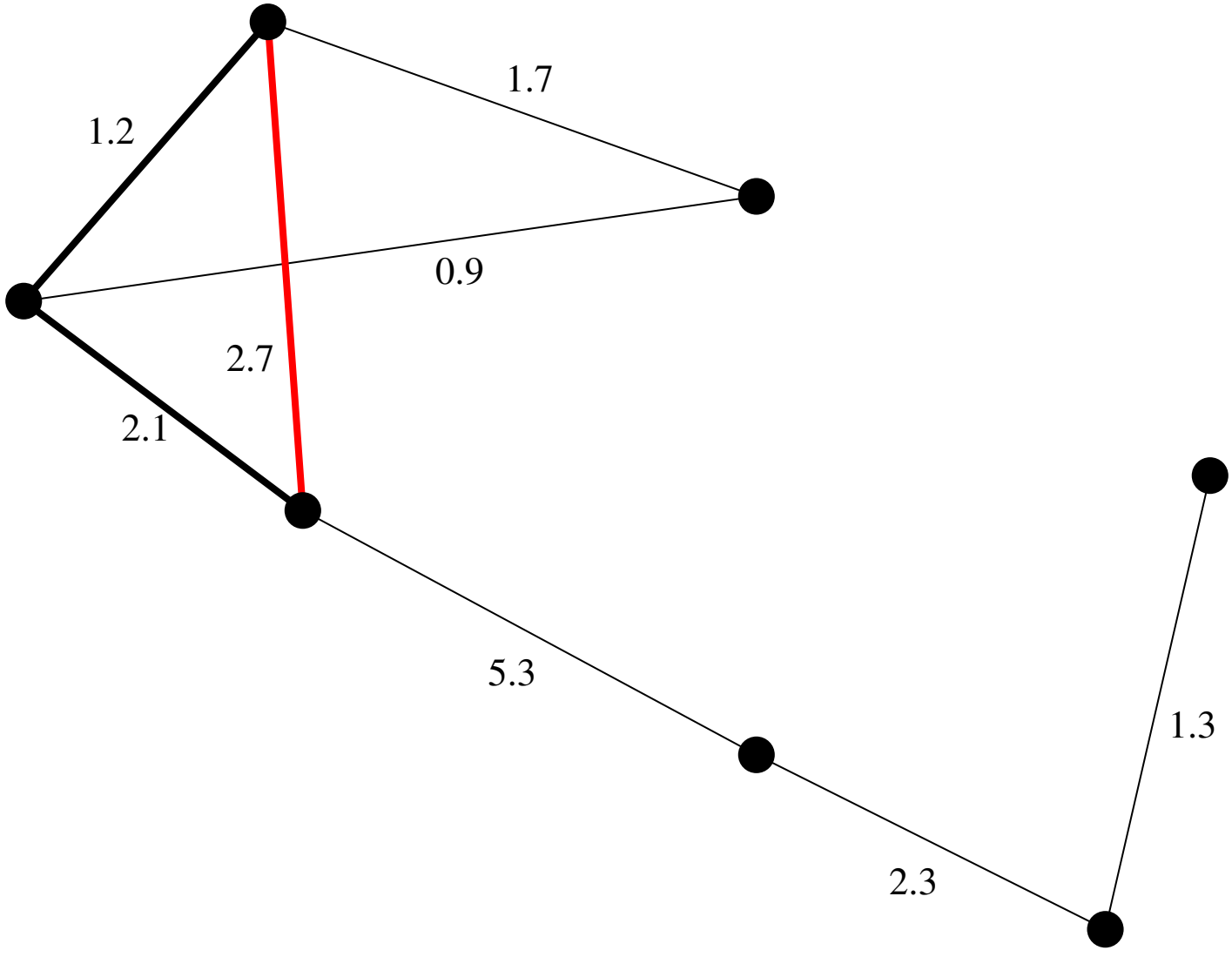
Cycle-breaking algorithm



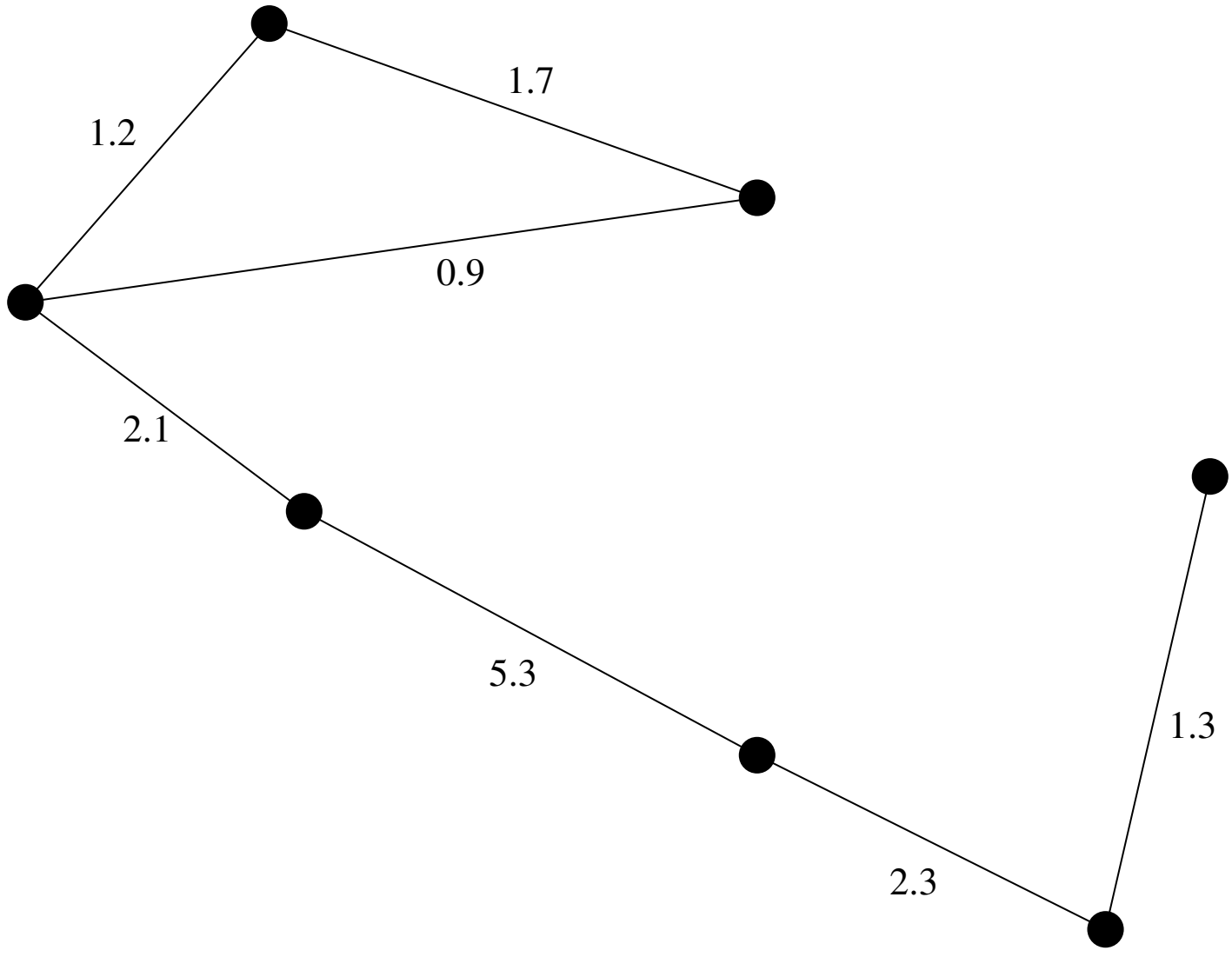
Cycle-breaking algorithm



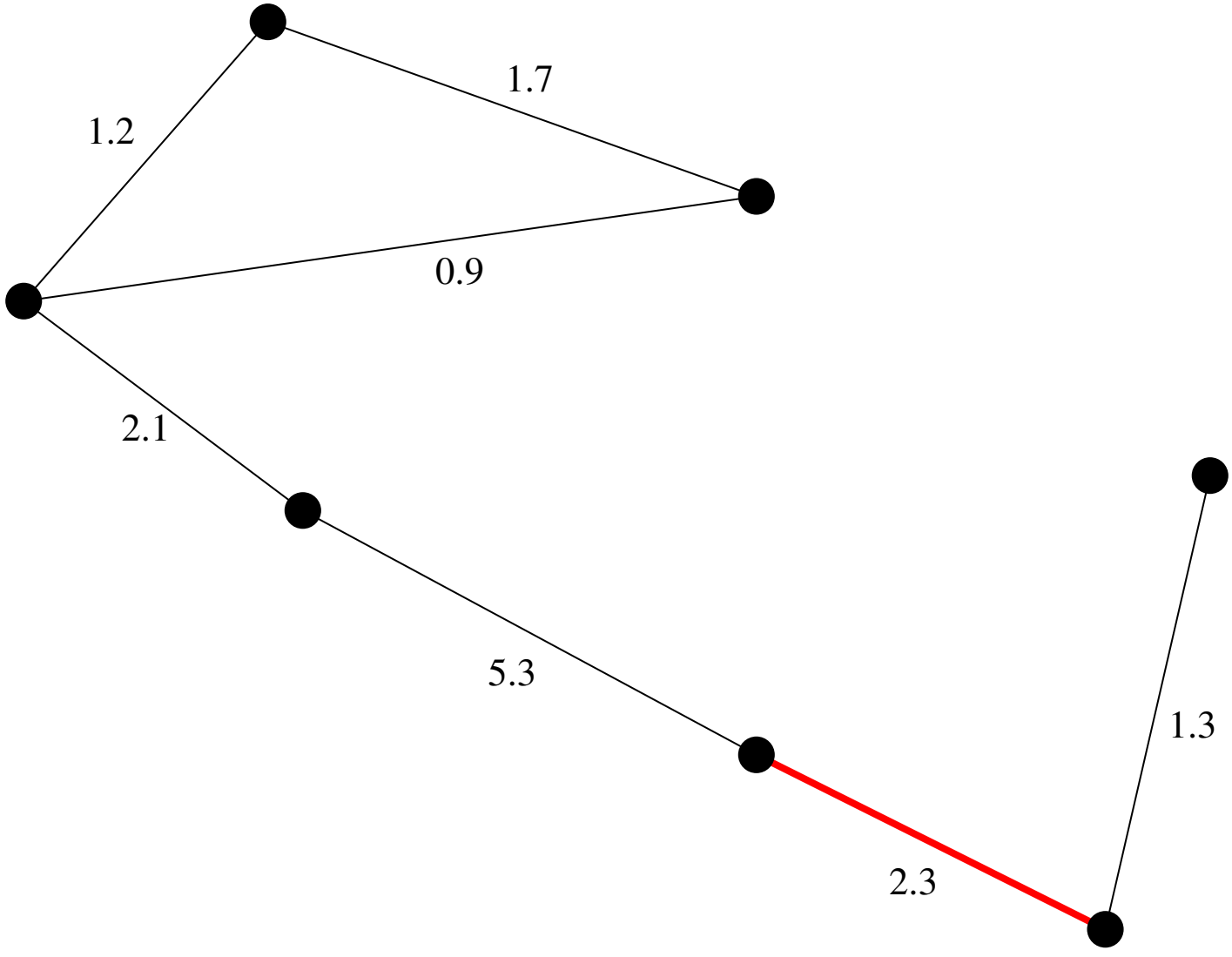
Cycle-breaking algorithm



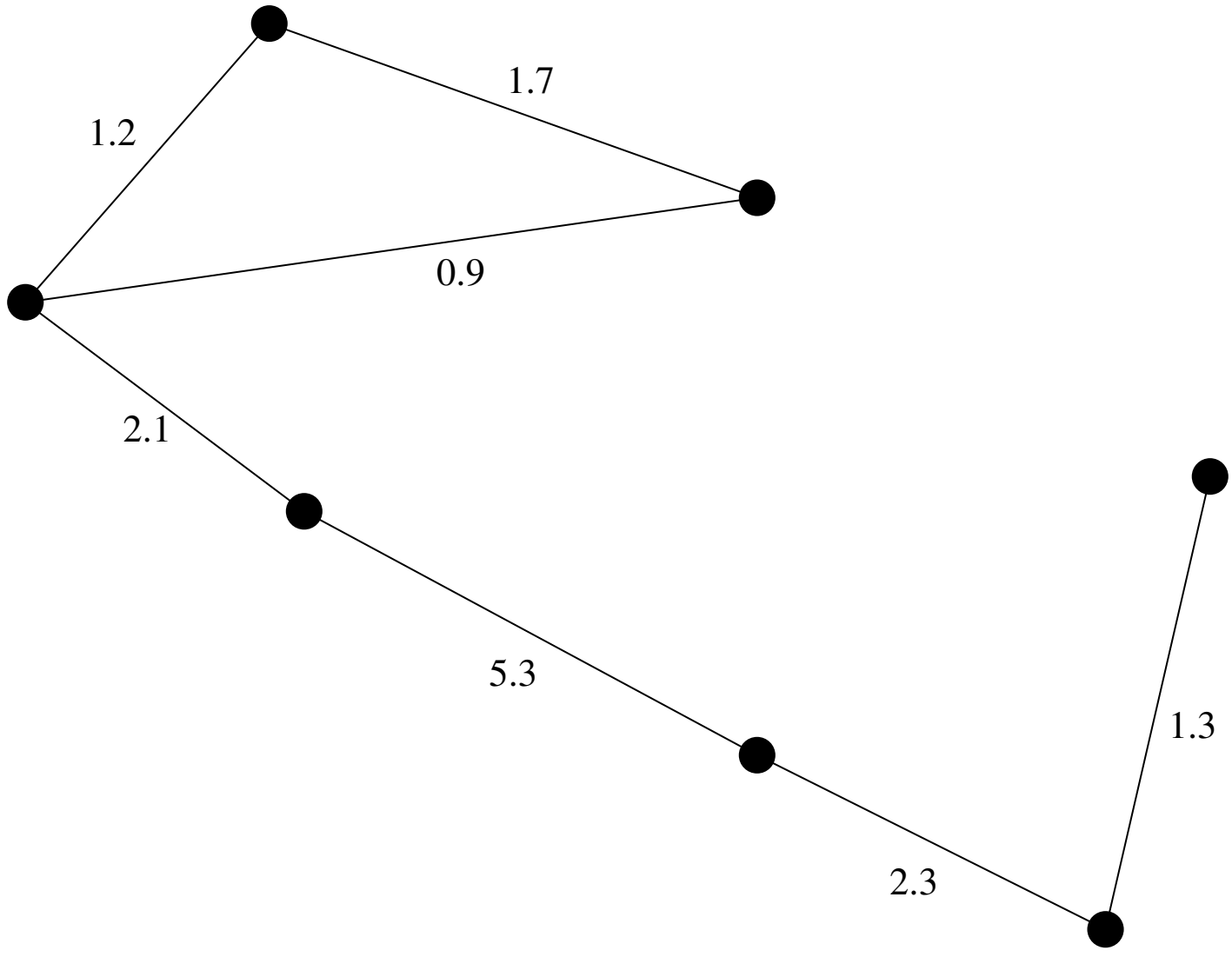
Cycle-breaking algorithm



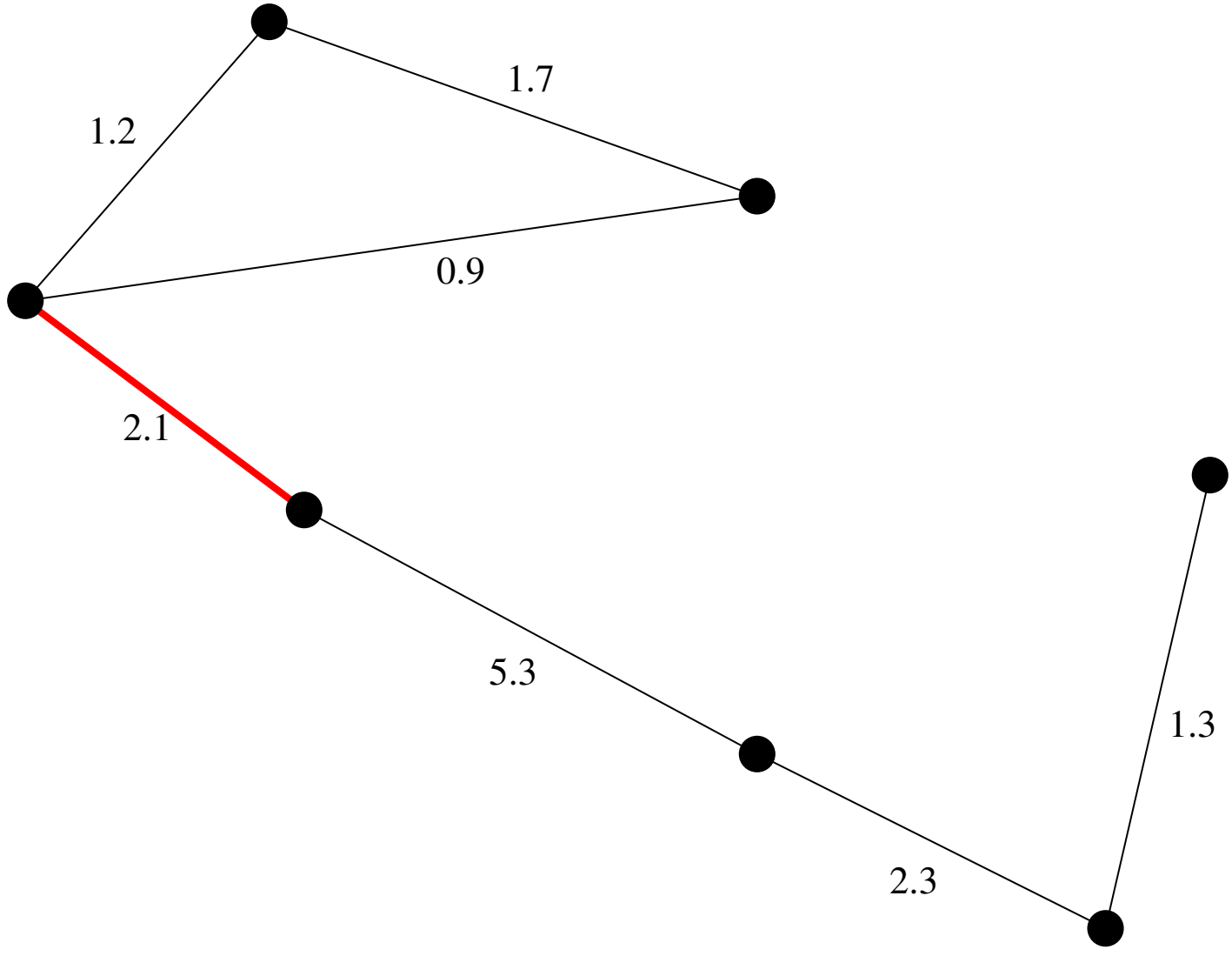
Cycle-breaking algorithm



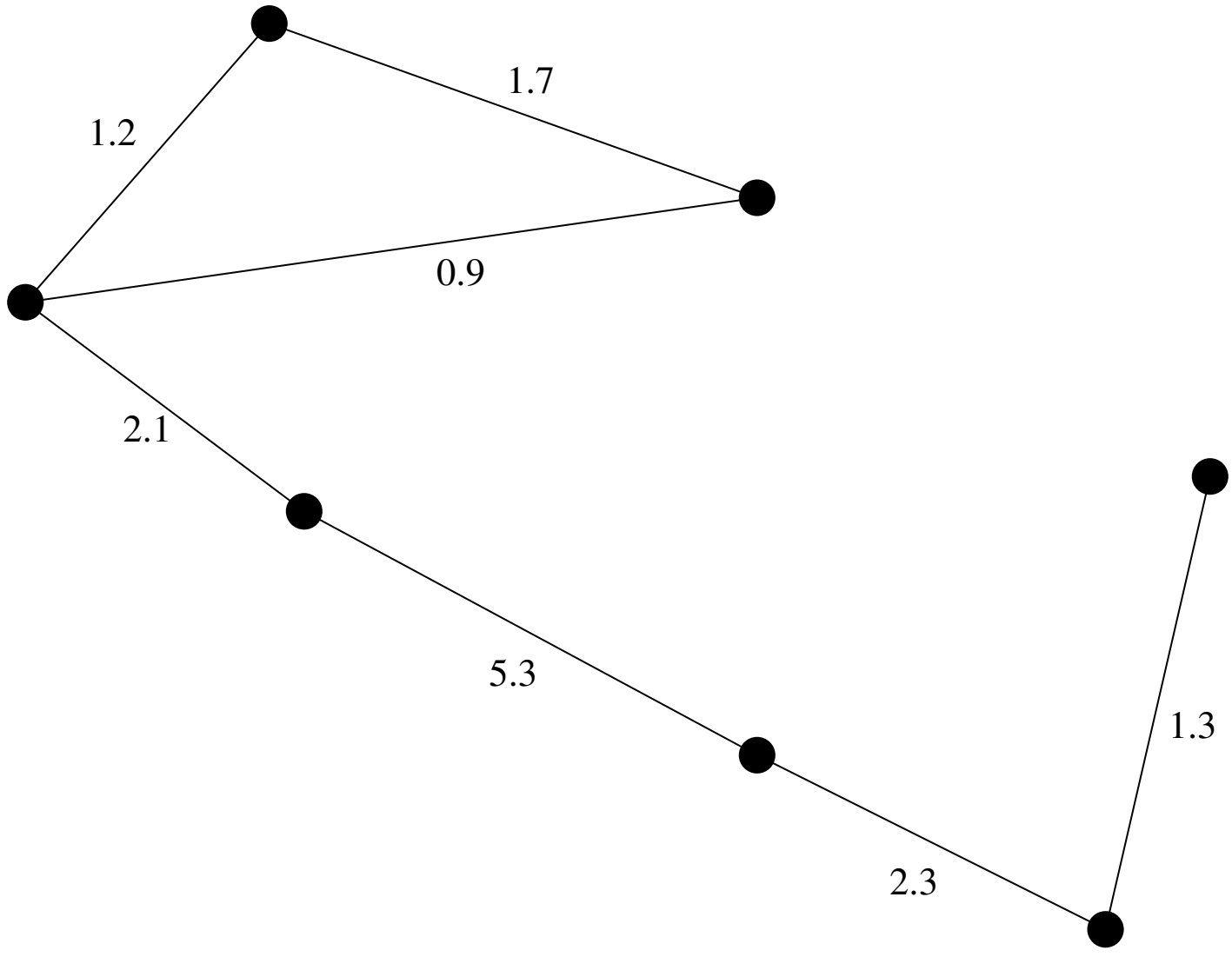
Cycle-breaking algorithm



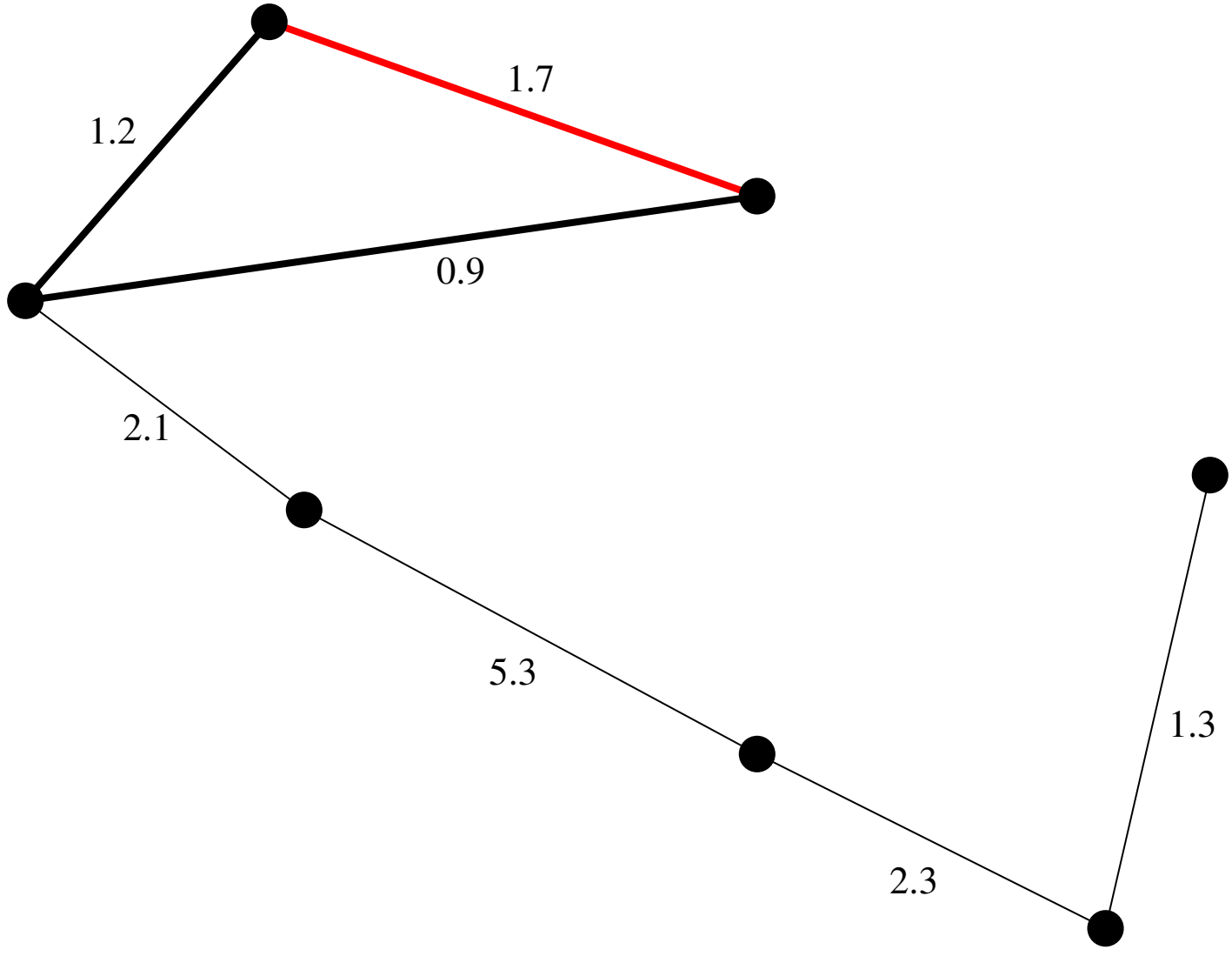
Cycle-breaking algorithm



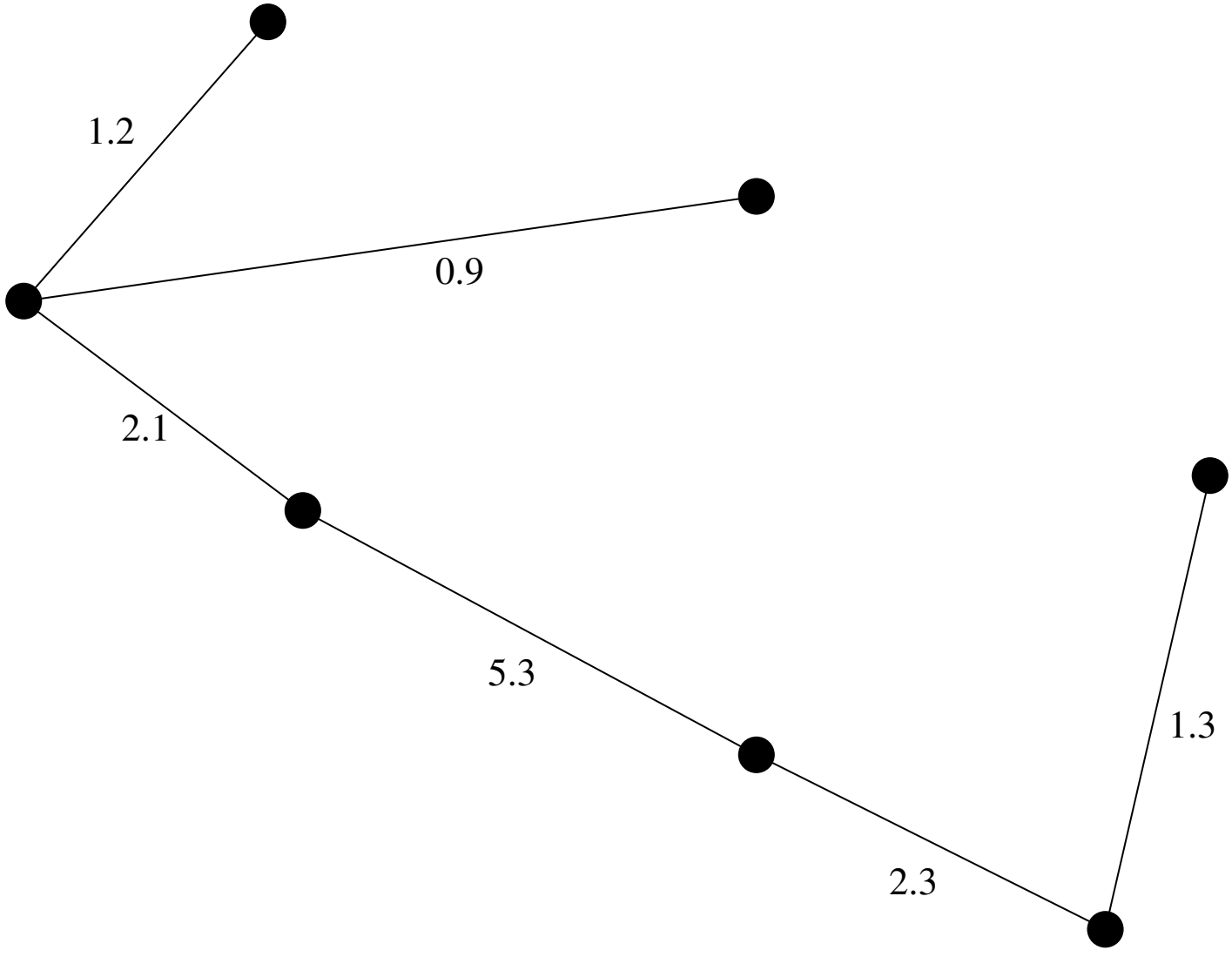
Cycle-breaking algorithm



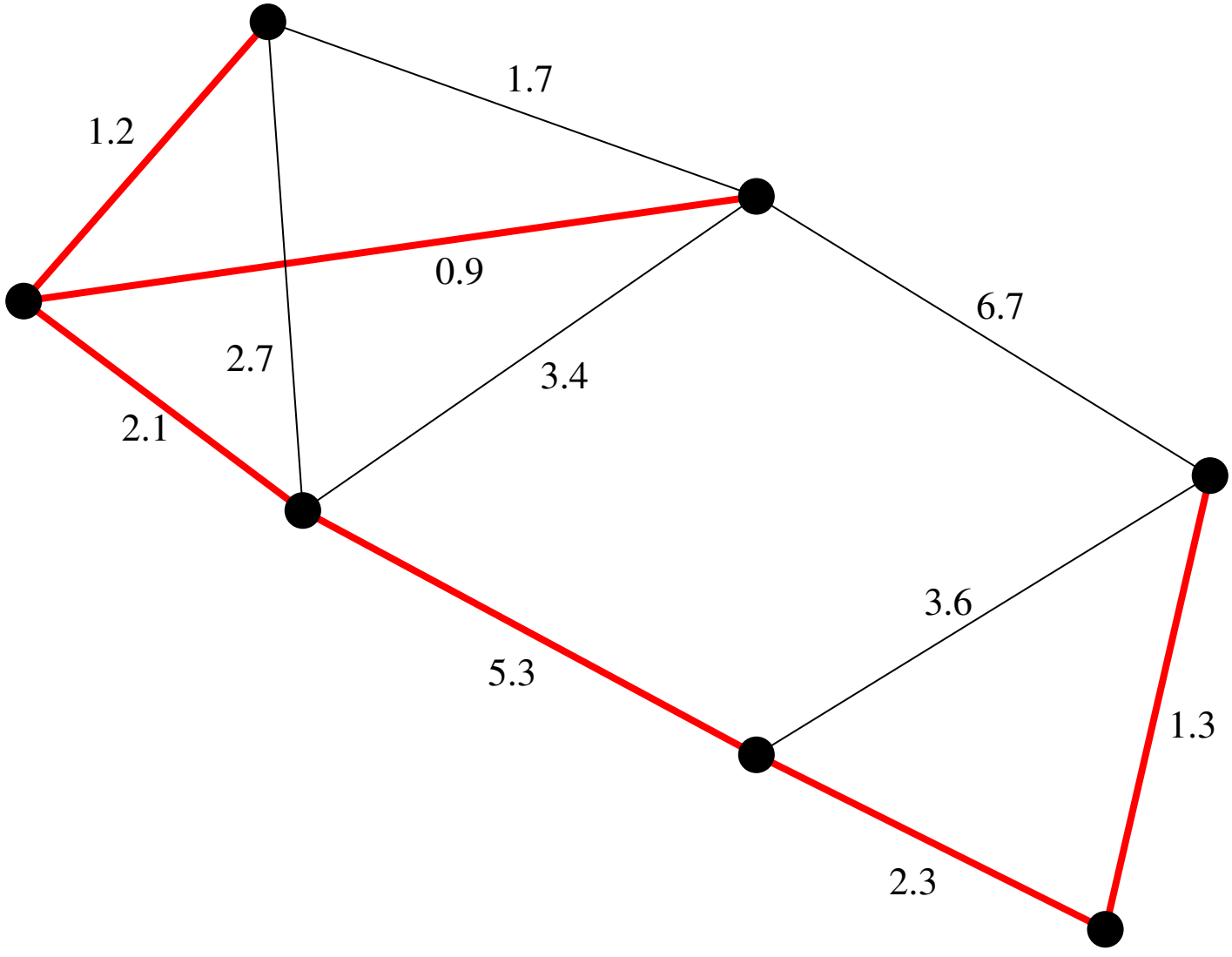
Cycle-breaking algorithm



Cycle-breaking algorithm

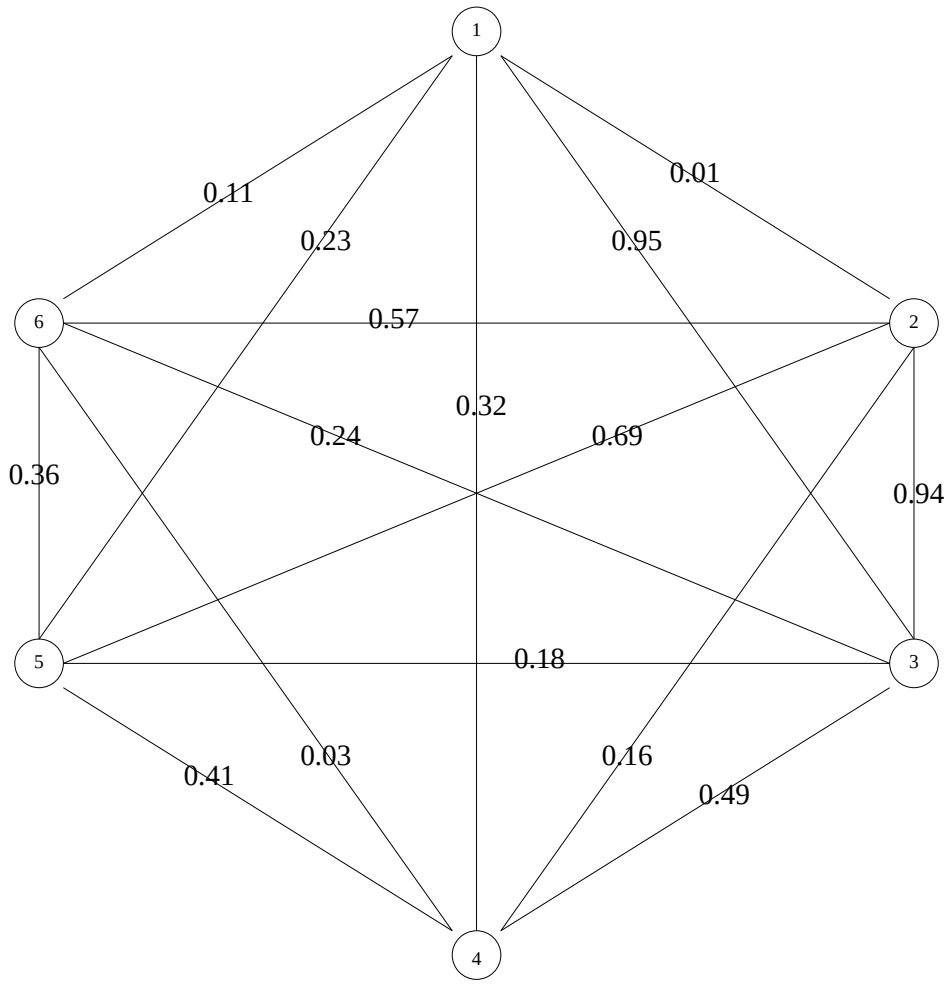


Cycle-breaking algorithm



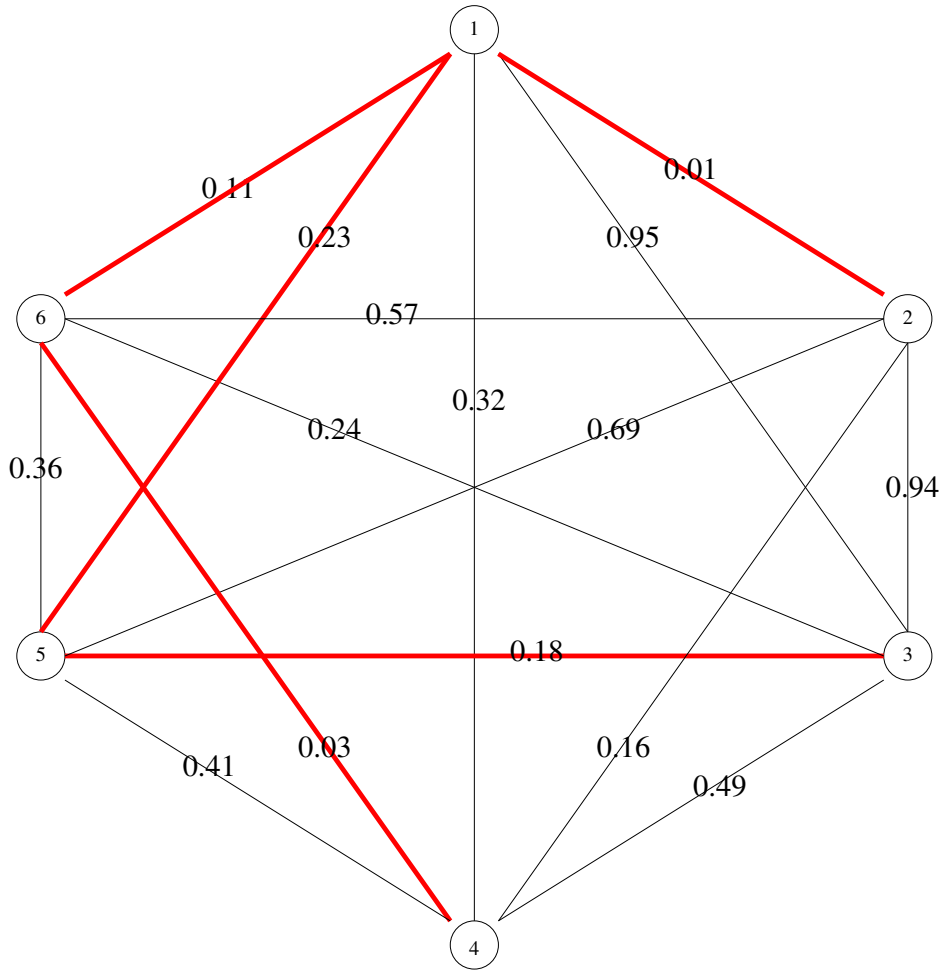
The minimum spanning tree of the complete graph

Consider the complete graph on n vertices with independent edge-weights which are uniformly distributed on $[0, 1]$.



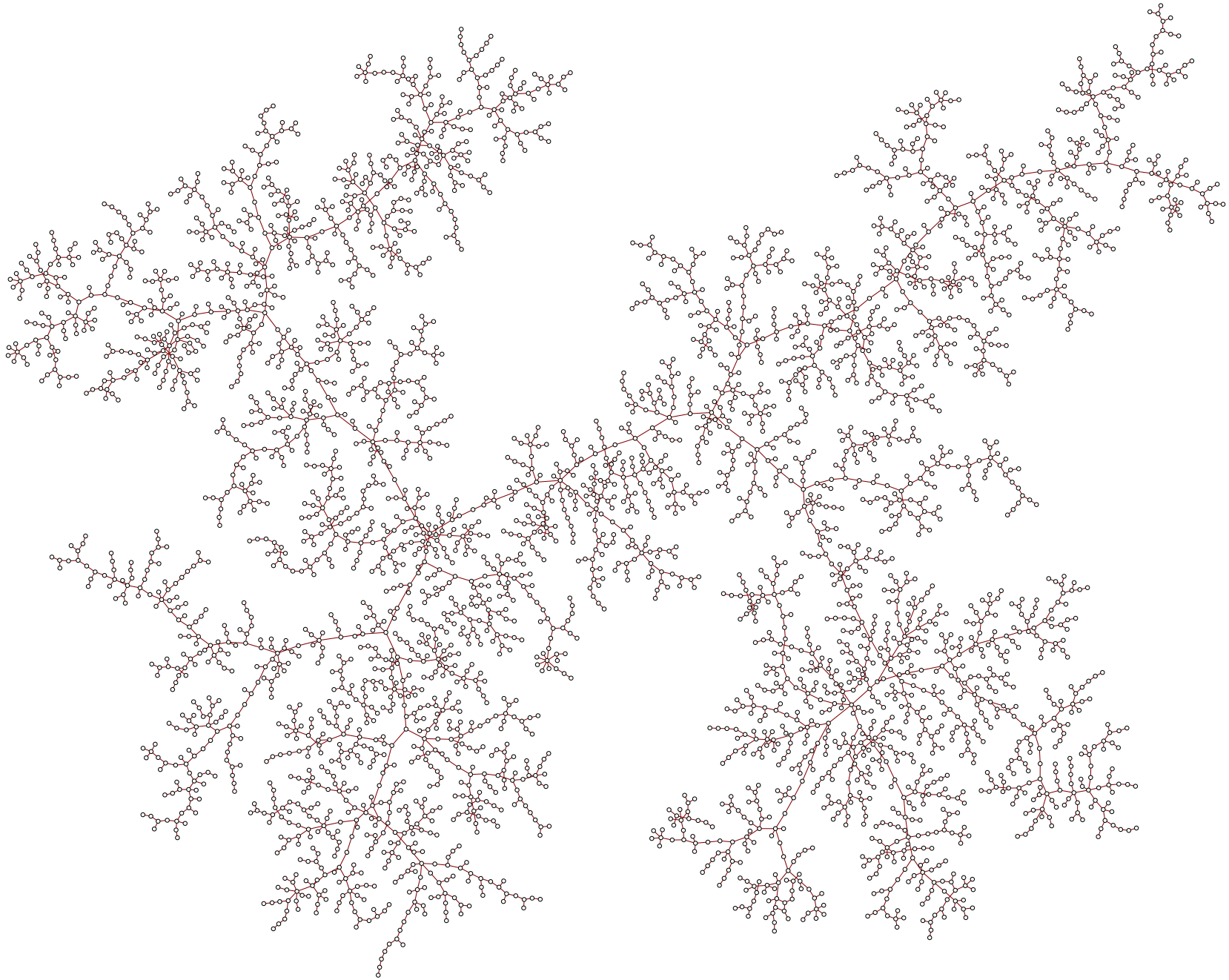
The minimum spanning tree of the complete graph

Consider the complete graph on n vertices with independent edge-weights which are uniformly distributed on $[0, 1]$.



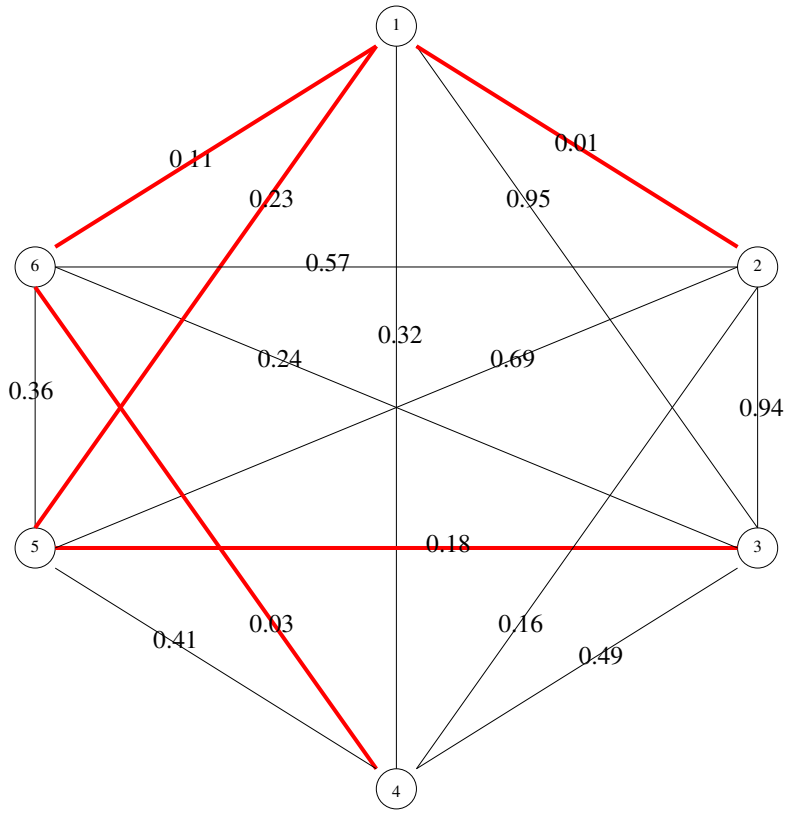
Question (Aldous (1990))

Does the MST of the complete graph on n vertices possess a scaling limit?



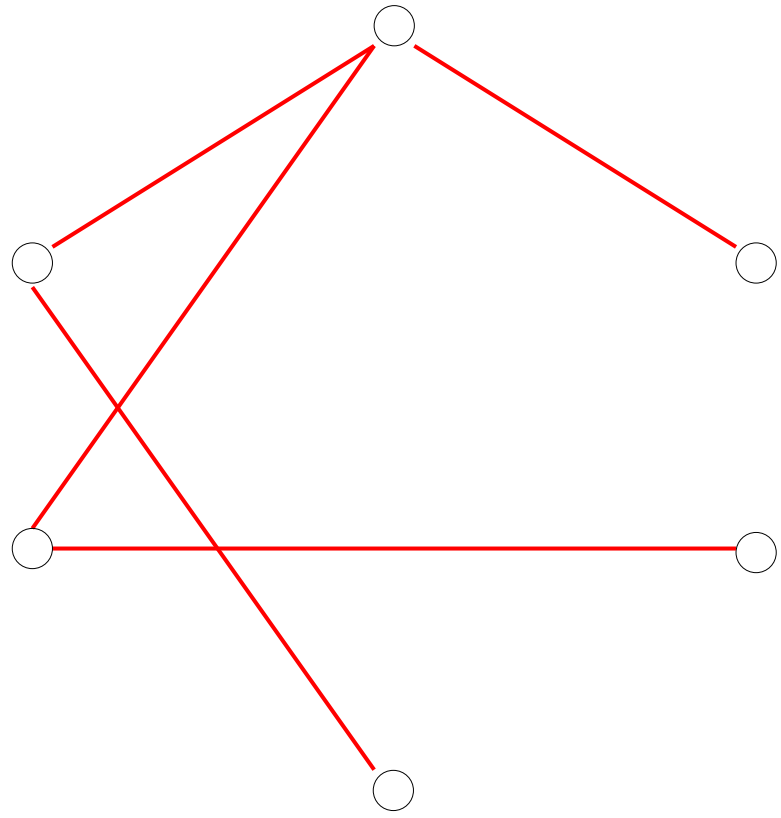
[Picture by Louigi Addario-Berry]

Graphs as metric spaces



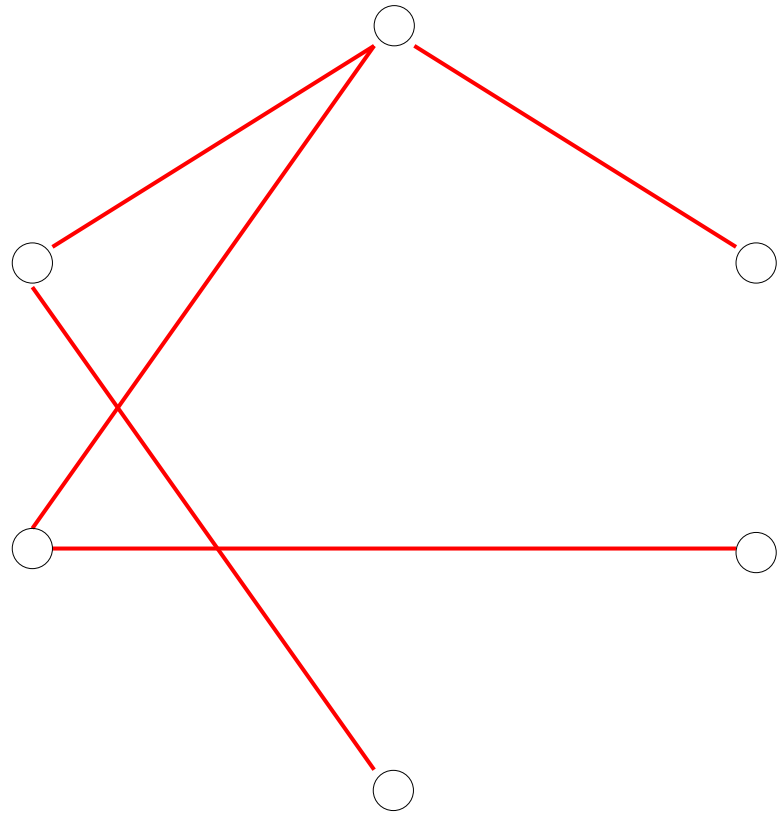
We don't really care about the vertex labels or the weights.

Graphs as metric spaces



So we will think of the tree as a metric space using the graph distance for the metric.

Graphs as metric spaces



So we will think of the tree as a metric space using the graph distance for the metric. We will also want to be able to pick points uniformly at random from the tree, so we equip it with the uniform measure (call this a measured metric space).

An easier scaling limit: the uniform spanning tree

Suppose that instead we consider T_n, the uniform spanning tree (UST) of the complete graph on n vertices.

An easier scaling limit: the uniform spanning tree

Suppose that instead we consider T_n, the uniform spanning tree (UST) of the complete graph on n vertices. (This is simply the uniform random tree on n vertices.)

An easier scaling limit: the uniform spanning tree

Suppose that instead we consider T_n, the uniform spanning tree (UST) of the complete graph on n vertices. (This is simply the uniform random tree on n vertices.)

Again, think of T_n as a measured metric space by using the graph distance and assigning each vertex weight $1/n$.

An easier scaling limit: the uniform spanning tree

Suppose that instead we consider T_n, the uniform spanning tree (UST) of the complete graph on n vertices. (This is simply the uniform random tree on n vertices.)

Again, think of T_n as a measured metric space by using the graph distance and assigning each vertex weight $1/n$.

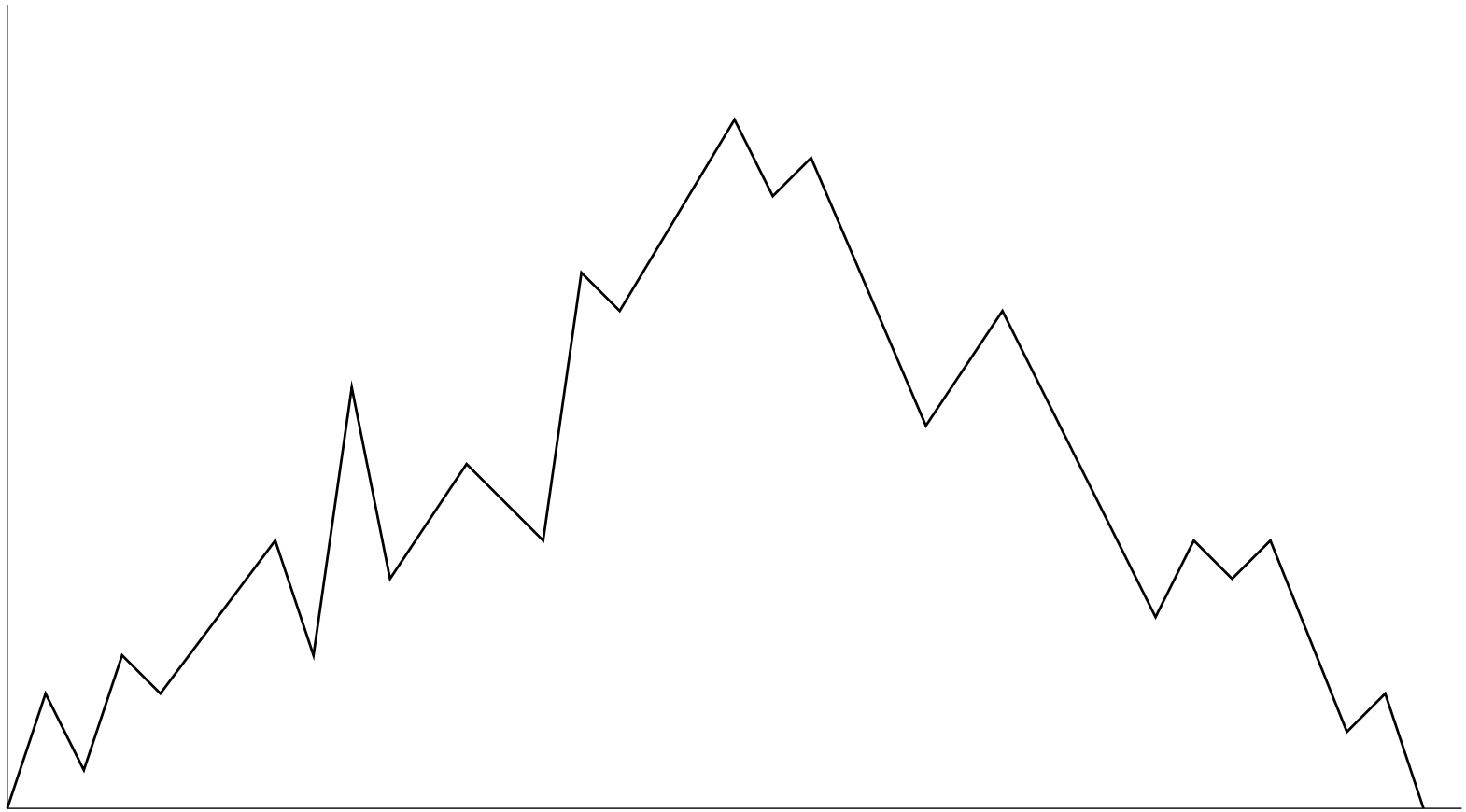
Theorem. (Aldous (1993); Le Gall (2006)) As $n \to \infty$,

$$\frac{1}{\sqrt{n}} T_n \xrightarrow{d} \mathcal{T},$$

in the sense of the Gromov-Hausdorff-Prokohorov distance $\mathrm{d_{GHP}}$ between compact measured metric spaces, where $\mathcal{T}$ is the Brownian continuum random tree.

Real trees from excursions

Let $h : [0,1] \to \mathbb{R}^+$ be an excursion, that is a continuous function such that $h(0) = h(1) = 0$ and $h(x) > 0$ for $x \in (0,1)$.



Real trees from excursions

Now put glue on the underside of the excursion and push the two sides together...

Real trees from excursions

Now put glue on the underside of the excursion and push the two sides together...

Real trees from excursions

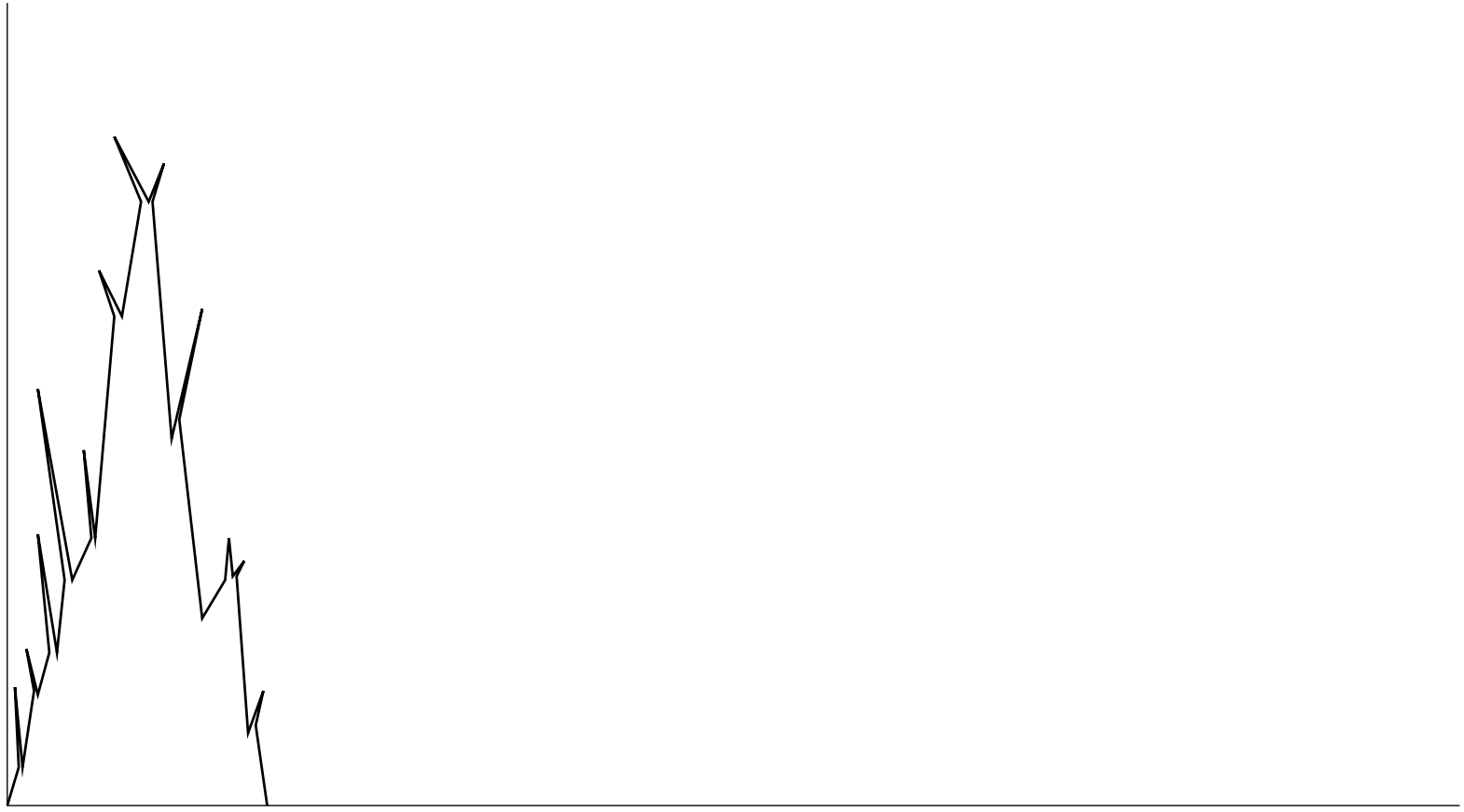
Now put glue on the underside of the excursion and push the two sides together...

Real trees from excursions

Now put glue on the underside of the excursion and push the two sides together...

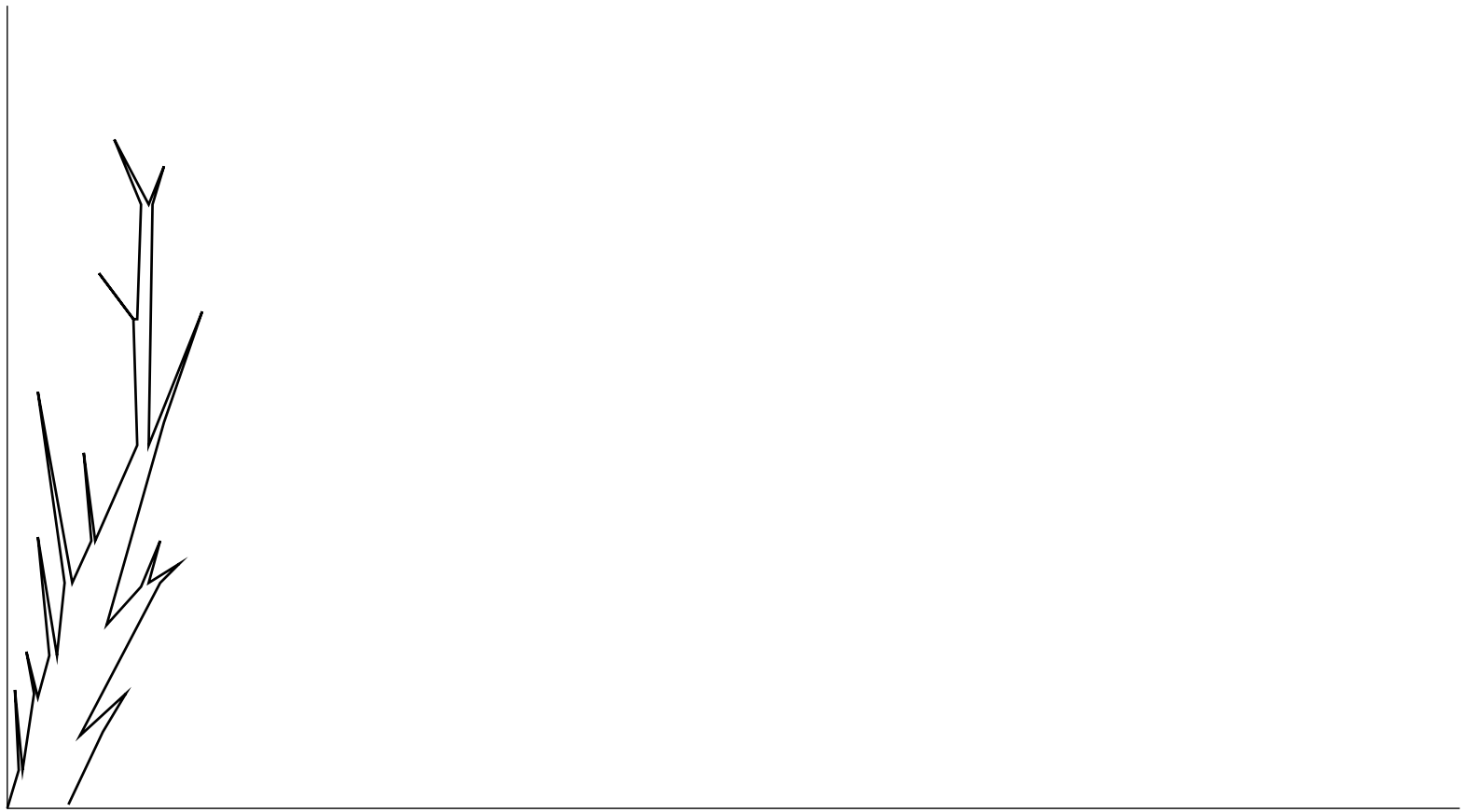
Real trees from excursions

Now put glue on the underside of the excursion and push the two sides together...



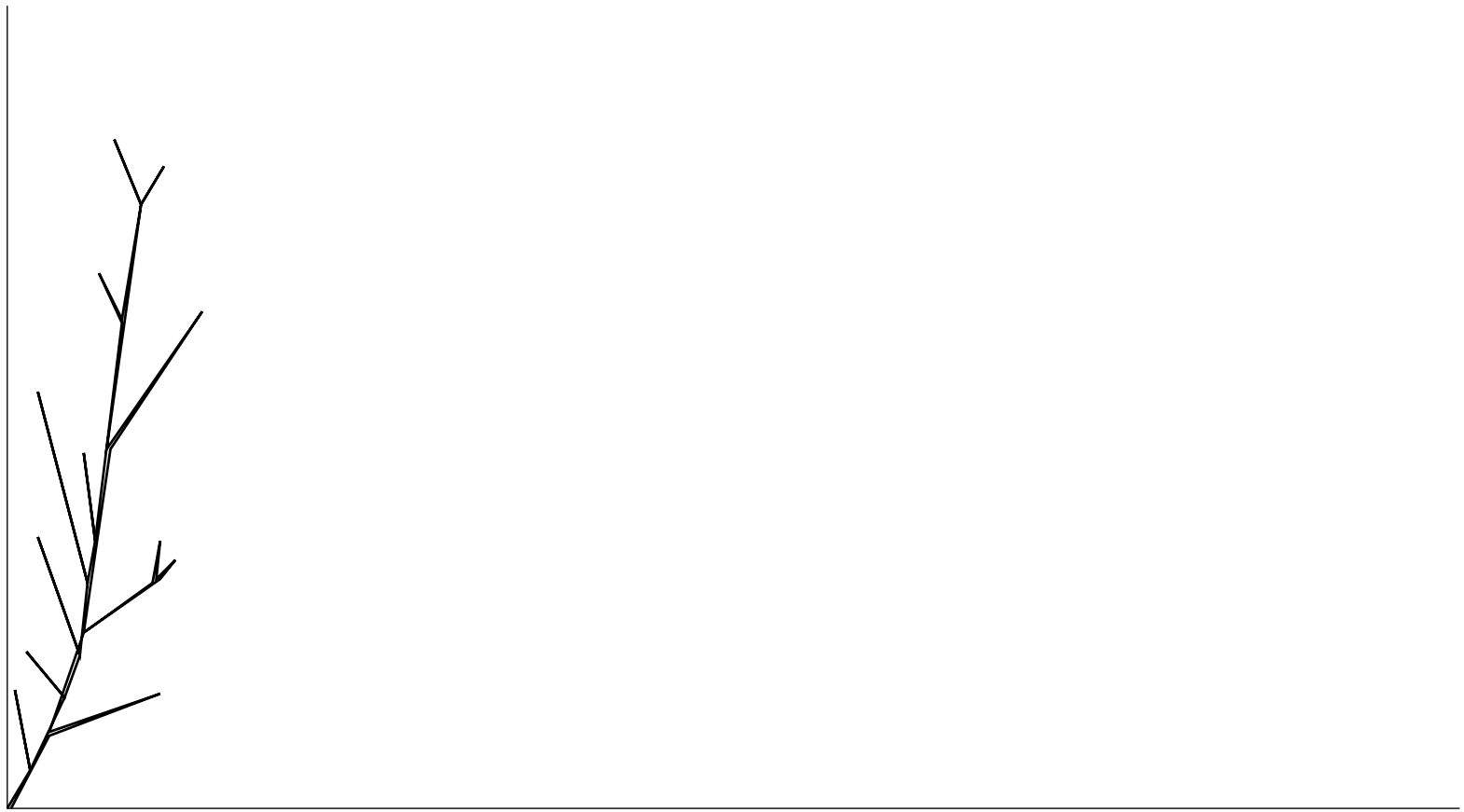
Real trees from excursions

Now put glue on the underside of the excursion and push the two sides together...



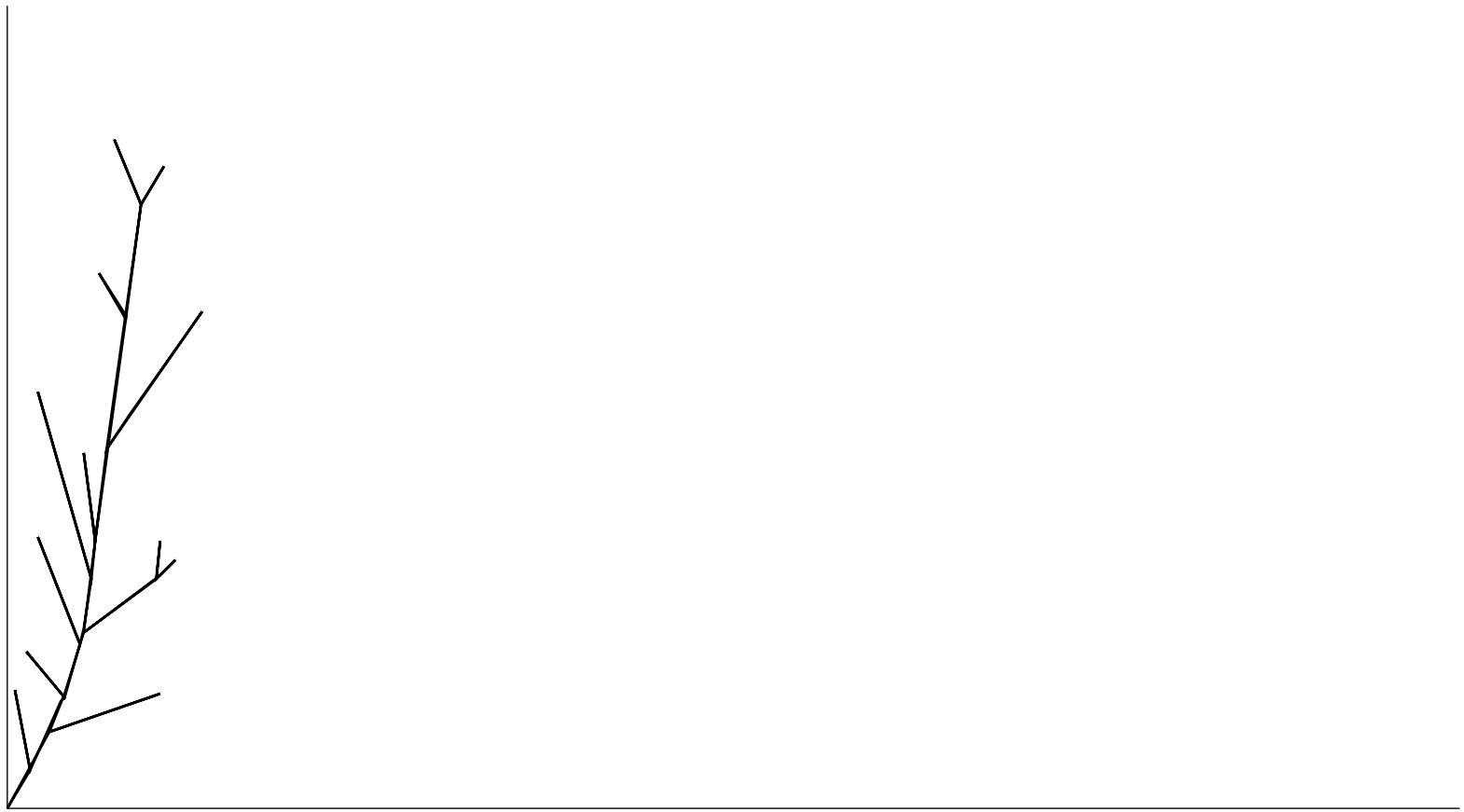
Real trees from excursions

Now put glue on the underside of the excursion and push the two sides together...



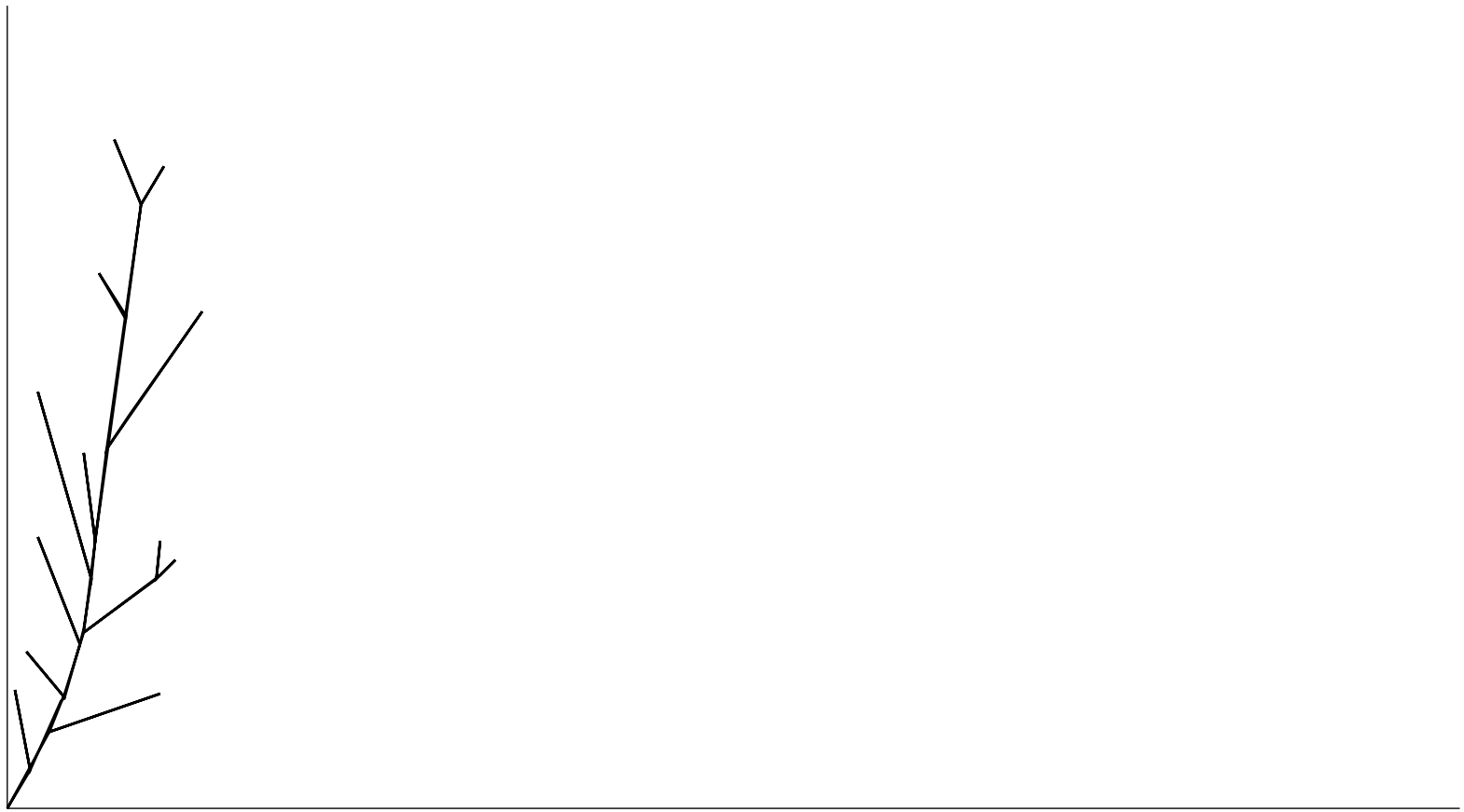
Real trees from excursions

Now put glue on the underside of the excursion and push the two sides together to get a tree.

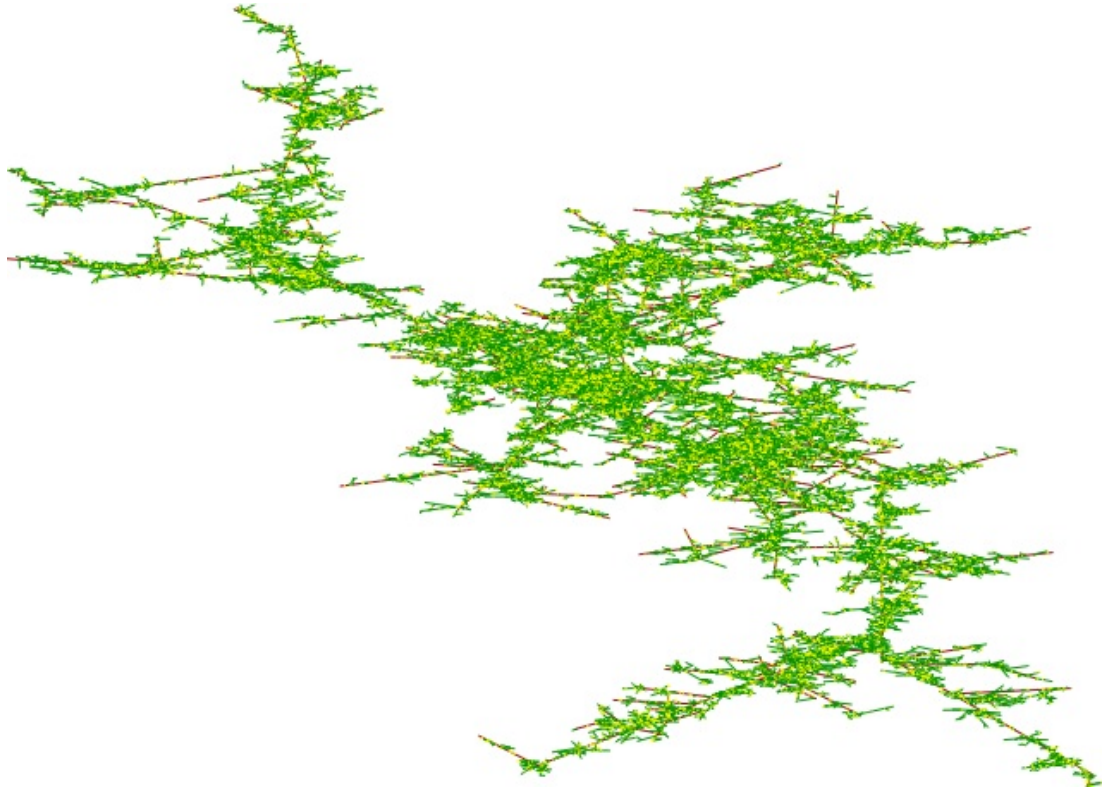


Real trees from excursions

Now put glue on the underside of the excursion and push the two sides together to get a real tree.



The Brownian CRT



[Picture by Grégory Miermont]

The Brownian continuum random tree $\mathcal{T}$ is the random real tree we obtain by doing this gluing procedure to the function $2e$, where $(e(x), 0 \leq x \leq 1)$ a standard Brownian excursion.

Properties of the Brownian CRT

Since local minima of e are a.s. unique, $\mathcal{T}$ is binary.

Properties of the Brownian CRT

Since local minima of e are a.s. unique, $\mathcal{T}$ is binary.

$\mathcal{T}$ comes naturally equipped with a uniform measure μ, which is the probability measure induced on $\mathcal{T}$ from the Lebesgue measure on $[0, 1]$. μ is concentrated on the leaves of the tree.

The scaling limit of the UST

Consider T_n, the UST of the complete graph on n vertices. Think of T_n as a measured metric space by using the graph distance and assigning each vertex weight $1/n$.

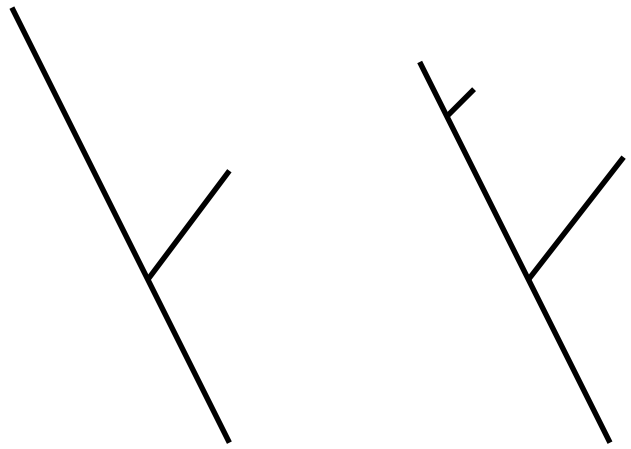
Theorem. (Aldous (1993); Le Gall (2006)) As $n \to \infty$,

$$\frac{1}{\sqrt{n}} T_n \xrightarrow{d} \mathcal{T},$$

in the sense of the Gromov-Hausdorff-Prokohorov distance $\mathrm{d_{GHP}}$ between compact measured metric spaces, where $\mathcal{T}$ is the Brownian continuum random tree.

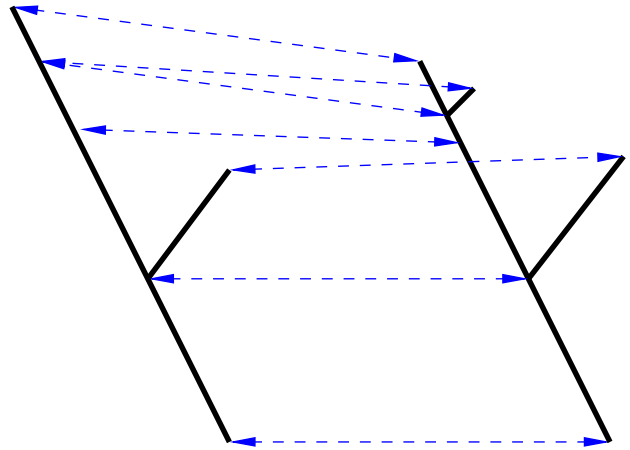
The distance between measured metric spaces

Suppose that (X, d, μ) and (X', d', μ') are measured metric spaces.



The distance between measured metric spaces

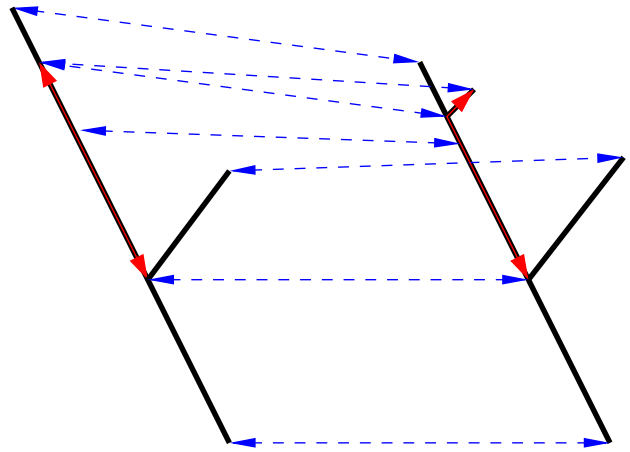
Suppose that (X, d, μ) and (X', d', μ') are measured metric spaces.



A correspondence R is a subset of $X \times X'$ such that for every $x \in X$, there exists $x' \in X'$ with $(x, x') \in R$ and vice versa.

The distance between measured metric spaces

Suppose that (X, d, μ) and (X', d', μ') are measured metric spaces.



A correspondence R is a subset of $X \times X'$ such that for every $x \in X$, there exists $x' \in X'$ with $(x, x') \in R$ and vice versa.

The distortion of R is

$$\mathrm{dis}(R) = \sup\{|d(x, y) - d'(x', y')| : (x, x'), (y, y') \in R\}.$$

Measuring the distance between measured metric spaces

(X, d, μ) and (X', d', μ') are close in the Gromov-Hausdorff-Prokhorov distance if:

We can find a correspondence R between X and X' and a measure π on $X \times X'$ in such a way that

Measuring the distance between measured metric spaces

(X, d, μ) and (X', d', μ') are close in the Gromov-Hausdorff-Prokhorov distance if:

We can find a correspondence R between X and X' and a measure π on $X \times X'$ in such a way that

- R has small distortion (i.e. $d(x, y)$ and $d'(x', y')$ are close whenever $x \in X$ corresponds to $x' \in X'$ and $y \in X$ corresponds to $y' \in X'$);

Measuring the distance between measured metric spaces

(X, d, μ) and (X', d', μ') are close in the Gromov-Hausdorff-Prokhorov distance if:

We can find a correspondence R between X and X' and a measure π on $X \times X'$ in such a way that

- R has small distortion (i.e. $d(x, y)$ and $d'(x', y')$ are close whenever $x \in X$ corresponds to $x' \in X'$ and $y \in X$ corresponds to $y' \in X'$);
- π is almost a coupling of μ and μ';

Measuring the distance between measured metric spaces

(X, d, μ) and (X', d', μ') are close in the Gromov-Hausdorff-Prokhorov distance if:

We can find a correspondence R between X and X' and a measure π on $X \times X'$ in such a way that

- R has small distortion (i.e. $d(x, y)$ and $d'(x', y')$ are close whenever $x \in X$ corresponds to $x' \in X'$ and $y \in X$ corresponds to $y' \in X'$);
- π is almost a coupling of μ and μ';
- $\pi(R^c)$ is small.

The scaling limit of the UST

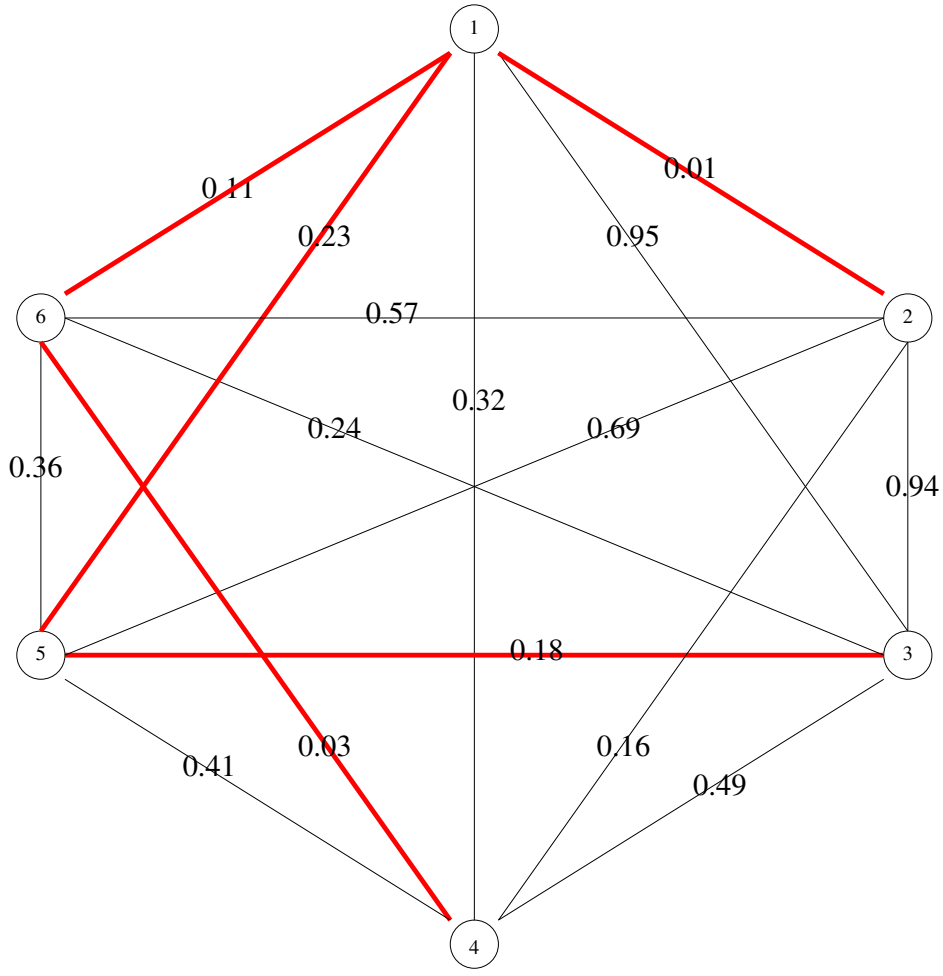
Consider T_n, the UST of the complete graph on n vertices. Think of T_n as a measured metric space by using the graph distance and assigning each vertex weight $1/n$.

Theorem. (Aldous (1993); Le Gall (2006)) As $n \to \infty$,

$$\frac{1}{\sqrt{n}} T_n \xrightarrow{d} \mathcal{T},$$

in the sense of the Gromov-Hausdorff-Prokohorov distance $\mathrm{d_{GHP}}$ between compact measured metric spaces, where $\mathcal{T}$ is the Brownian continuum random tree.

Back to the original question: what about the MST?



Does the MST of the complete graph on n vertices possess a scaling limit?

Hold on, isn't it the same?!

a UST on 3000 vertices

an MST on 3000 vertices

[Pictures by Karl-Friedrich Israel]

The scaling limit of the MST

Let M_n be the MST of the complete graph on n vertices, considered as a metric space using the graph distance, and endowed with the uniform measure on its vertices.

Theorem. (Addario-Berry, Broutin, G. & Miermont) There exists a random compact measured metric space $\mathcal{M}$ such that

$$\frac{1}{n^{1/3}} M_n \xrightarrow{d} \mathcal{M}$$

as $n \to \infty$, in the sense of $\mathrm{d_{GHP}}$.

The scaling limit of the MST

Let M_n be the MST of the complete graph on n vertices, considered as a metric space using the graph distance, and endowed with the uniform measure on its vertices.

Theorem. (Addario-Berry, Broutin, G. & Miermont) There exists a random compact measured metric space $\mathcal{M}$ such that

$$\frac{1}{n^{1/3}} M_n \xrightarrow{d} \mathcal{M}$$

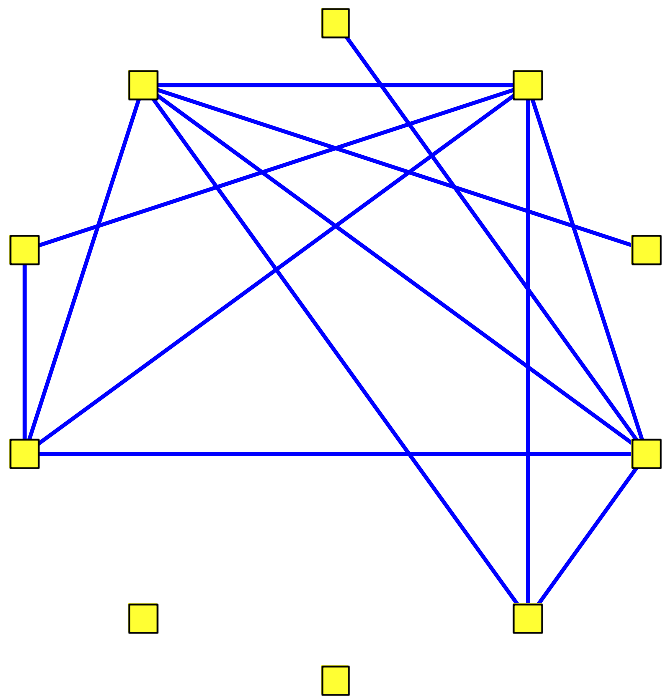
as $n \to \infty$, in the sense of $\mathrm{d_{GHP}}$.

The key to understanding this result is a connection to the Erdős-Rényi random graph.

Reminder: the Erdős-Rényi random graph

Take n vertices labelled by $\{1, 2, \ldots, n\}$ and put an edge between any pair independently with probability p. Call the resulting model $\mathbb{G}(n, p)$. (Bond percolation on the complete graph.)

Example: $n = 10$, $p = 0.4$ (vertex labels omitted).



The Erdős-Rényi random graph process

Take the obvious coupling of $(\mathbb{G}(n,p), p \in [0,1])$: simply assign each edge e an independent random edge-weight $W(e)$ with uniform distribution on $[0,1]$ and keep all edges with weight $W(e) \leq p$.

The Erdős-Rényi random graph process

Take the obvious coupling of $(\mathbb{G}(n,p), p \in [0,1])$: simply assign each edge e an independent random edge-weight $W(e)$ with uniform distribution on $[0,1]$ and keep all edges with weight $W(e) \leq p$.

It's then natural to think of p as a time and the evolving graph as a process.

Kruskal and Erdős-Rényi

By using the same weights, we can think of (a continuous-time version of) Kruskal's algorithm as "sitting inside" the Erdős-Rényi process: we call this the Kruskal process. When an edge appears in the Erdős-Rényi process, it gets included in the Kruskal process as long as it doesn't create a cycle.

Kruskal and Erdős-Rényi

By using the same weights, we can think of (a continuous-time version of) Kruskal's algorithm as "sitting inside" the Erdős-Rényi process: we call this the Kruskal process. When an edge appears in the Erdős-Rényi process, it gets included in the Kruskal process as long as it doesn't create a cycle.

In particular, the components in both processes (which are all trees in the case of Kruskal) have the same vertex-sets.

Kruskal and Erdős-Rényi

By using the same weights, we can think of (a continuous-time version of) Kruskal's algorithm as "sitting inside" the Erdős-Rényi process: we call this the Kruskal process. When an edge appears in the Erdős-Rényi process, it gets included in the Kruskal process as long as it doesn't create a cycle.

In particular, the components in both processes (which are all trees in the case of Kruskal) have the same vertex-sets.

Indeed, the Kruskal components are the MST's of the Erdős-Rényi components.

The phase transition

Sitting inside each component of the Erdős-Rényi process is a component of the Kruskal process.

The phase transition

Sitting inside each component of the Erdős-Rényi process is a component of the Kruskal process. So above the critical point $p = 1/n$, there is a giant component for Kruskal too and, outside the giant component, there are only tiny tree components which gradually get glued onto the giant.

The phase transition

Sitting inside each component of the Erdős-Rényi process is a component of the Kruskal process. So above the critical point $p = 1/n$, there is a giant component for Kruskal too and, outside the giant component, there are only tiny tree components which gradually get glued onto the giant.

This entails that by time $p = (1 + \epsilon)/n$, the global metric structure of the minimum spanning tree has essentially already been built.

The phase transition

Sitting inside each component of the Erdős-Rényi process is a component of the Kruskal process. So above the critical point $p = 1/n$, there is a giant component for Kruskal too and, outside the giant component, there are only tiny tree components which gradually get glued onto the giant.

This entails that by time $p = (1 + \epsilon)/n$, the global metric structure of the minimum spanning tree has essentially already been built. (However, the vast majority of the mass is still outside the largest component!)

The phase transition

Sitting inside each component of the Erdős-Rényi process is a component of the Kruskal process. So above the critical point $p = 1/n$, there is a giant component for Kruskal too and, outside the giant component, there are only tiny tree components which gradually get glued onto the giant.

This entails that by time $p = (1 + \epsilon)/n$, the global metric structure of the minimum spanning tree has essentially already been built. (However, the vast majority of the mass is still outside the largest component!)

So the action is happening a bit earlier, in the critical window for the Erdős-Rényi phase transition.

The critical random graph

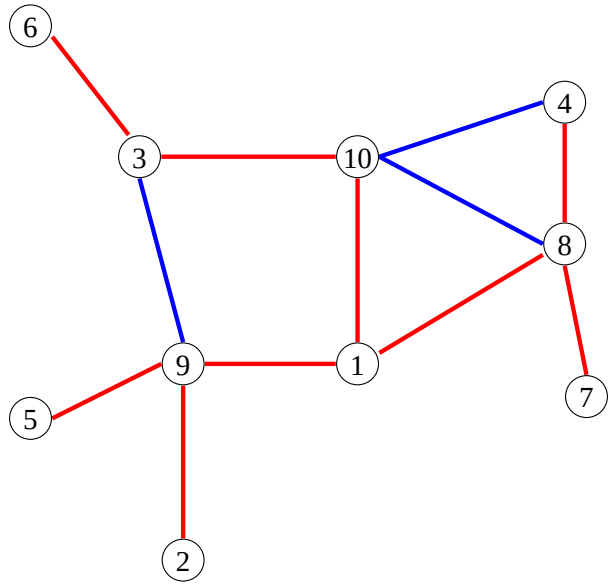
The critical window: $p = \frac{1}{n} + \frac{\lambda}{n^{4/3}}$, where $\lambda \in \mathbb{R}$. For such p, the largest components have size $\Theta(n^{2/3})$.

The critical random graph

The critical window: $p = \frac{1}{n} + \frac{\lambda}{n^{4/3}}$, where $\lambda \in \mathbb{R}$. For such p, the largest components have size $\Theta(n^{2/3})$.

Define the surplus of a component to be the number of edges more than a tree that it has.

A component with surplus 3:



Convergence of the sizes and surpluses

Fix $\lambda \in \mathbb{R}$. Let $C_\lambda^{n,1}, C_\lambda^{n,2}, \ldots$ be the sequence of component sizes of $\mathbb{G}(n, n^{-1} + \lambda n^{-4/3})$ in decreasing order, and let $S_\lambda^{n,1}, S_\lambda^{n,2}, \ldots$ be their surpluses.

Write $\mathbf{C}_\lambda^n = (C_\lambda^{n,1}, C_\lambda^{n,1}, \ldots)$ and $\mathbf{S}_\lambda^n = (S_\lambda^{n,1}, S_\lambda^{n,2}, \ldots)$.

Convergence of the sizes and surpluses

Fix $\lambda \in \mathbb{R}$. Let $C_\lambda^{n,1}, C_\lambda^{n,2}, \ldots$ be the sequence of component sizes of $\mathbb{G}(n, n^{-1} + \lambda n^{-4/3})$ in decreasing order, and let $S_\lambda^{n,1}, S_\lambda^{n,2}, \ldots$ be their surpluses.

Write $\mathbf{C}_\lambda^n = (C_\lambda^{n,1}, C_\lambda^{n,1}, \ldots)$ and $\mathbf{S}_\lambda^n = (S_\lambda^{n,1}, S_\lambda^{n,2}, \ldots)$.

Theorem (Aldous (1997)). For fixed $\lambda \in \mathbb{R}$, as $n \to \infty$,

$$(n^{-2/3}\mathbf{C}_\lambda^n, \mathbf{S}_\lambda^n) \xrightarrow{d} (\mathbf{C}_\lambda, \mathbf{S}_\lambda).$$

Convergence of the sizes and surpluses

Fix $\lambda \in \mathbb{R}$. Let $C_\lambda^{n,1}, C_\lambda^{n,2}, \ldots$ be the sequence of component sizes of $\mathbb{G}(n, n^{-1} + \lambda n^{-4/3})$ in decreasing order, and let $S_\lambda^{n,1}, S_\lambda^{n,2}, \ldots$ be their surpluses.

Write $\mathbf{C}_\lambda^n = (C_\lambda^{n,1}, C_\lambda^{n,1}, \ldots)$ and $\mathbf{S}_\lambda^n = (S_\lambda^{n,1}, S_\lambda^{n,2}, \ldots)$.

Theorem (Aldous (1997)). For fixed $\lambda \in \mathbb{R}$, as $n \to \infty$,

$$(n^{-2/3}\mathbf{C}_\lambda^n, \mathbf{S}_\lambda^n) \xrightarrow{d} (\mathbf{C}_\lambda, \mathbf{S}_\lambda).$$

(Convergence for the first co-ordinate takes place in

$$\ell^2_\searrow := \left\{ \mathbf{x} = (x_1, x_2, \ldots) : x_1 \geq x_2 \geq \ldots \geq 0, \sum_{i=1}^{\infty} x_i^2 < \infty \right\}$$

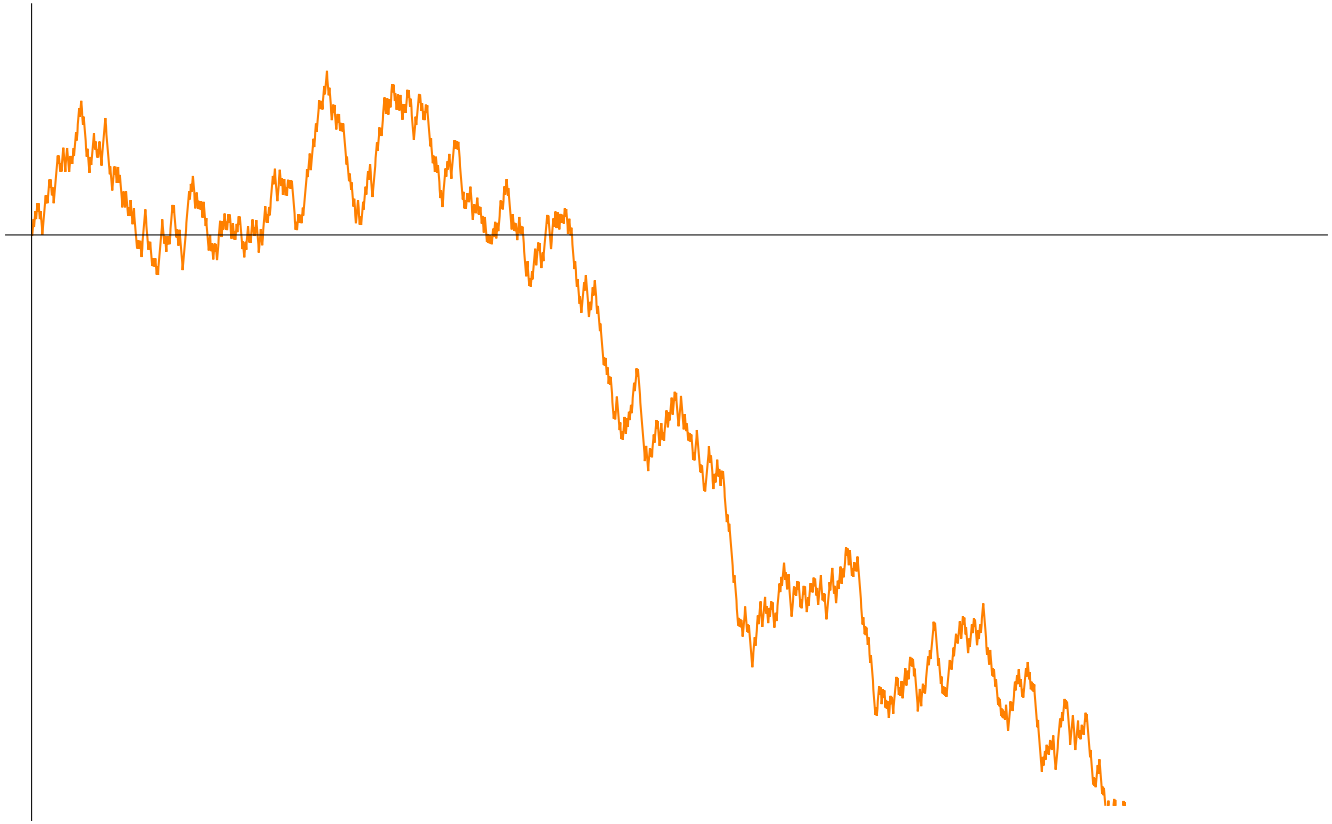
and for the second in the sense of finite-dimensional distributions.)

Limiting sizes and surpluses

Let $W_\lambda(t) = W(t) + \lambda t - \frac{t^2}{2}$, $t \geq 0$, where $(W(t), t \geq 0)$ is a standard Brownian motion.

Limiting sizes and surpluses

Let $W_\lambda(t) = W(t) + \lambda t - \frac{t^2}{2}$, $t \geq 0$, where $(W(t), t \geq 0)$ is a standard Brownian motion.

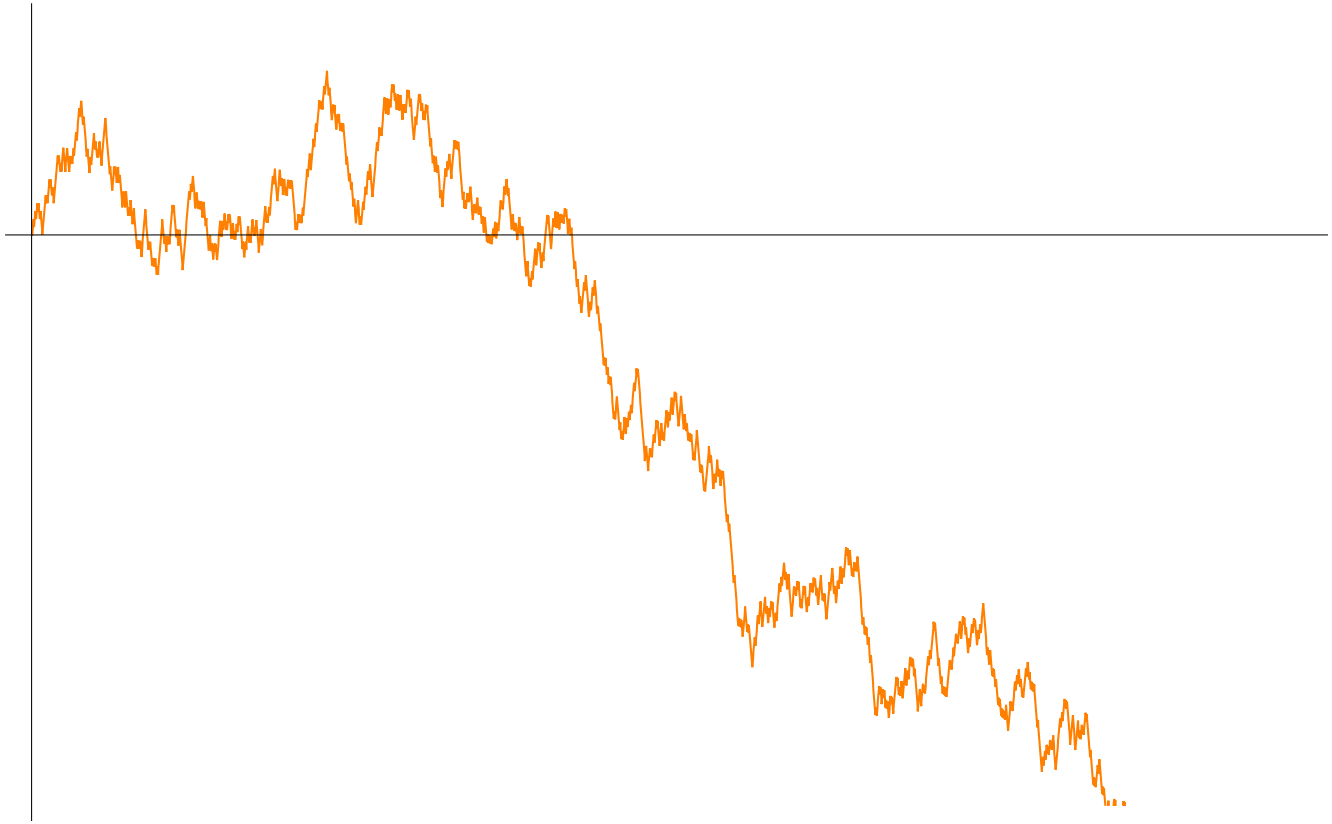


[Pictures by Louigi Addario-Berry]

Limiting sizes and surpluses

Let $W_\lambda(t) = W(t) + \lambda t - \frac{t^2}{2}$, $t \geq 0$, where $(W(t), t \geq 0)$ is a standard Brownian motion.

Let $B_\lambda(t) = W_\lambda(t) - \min_{0 \leq s \leq t} W_\lambda(s)$ be the process reflected at its minimum.

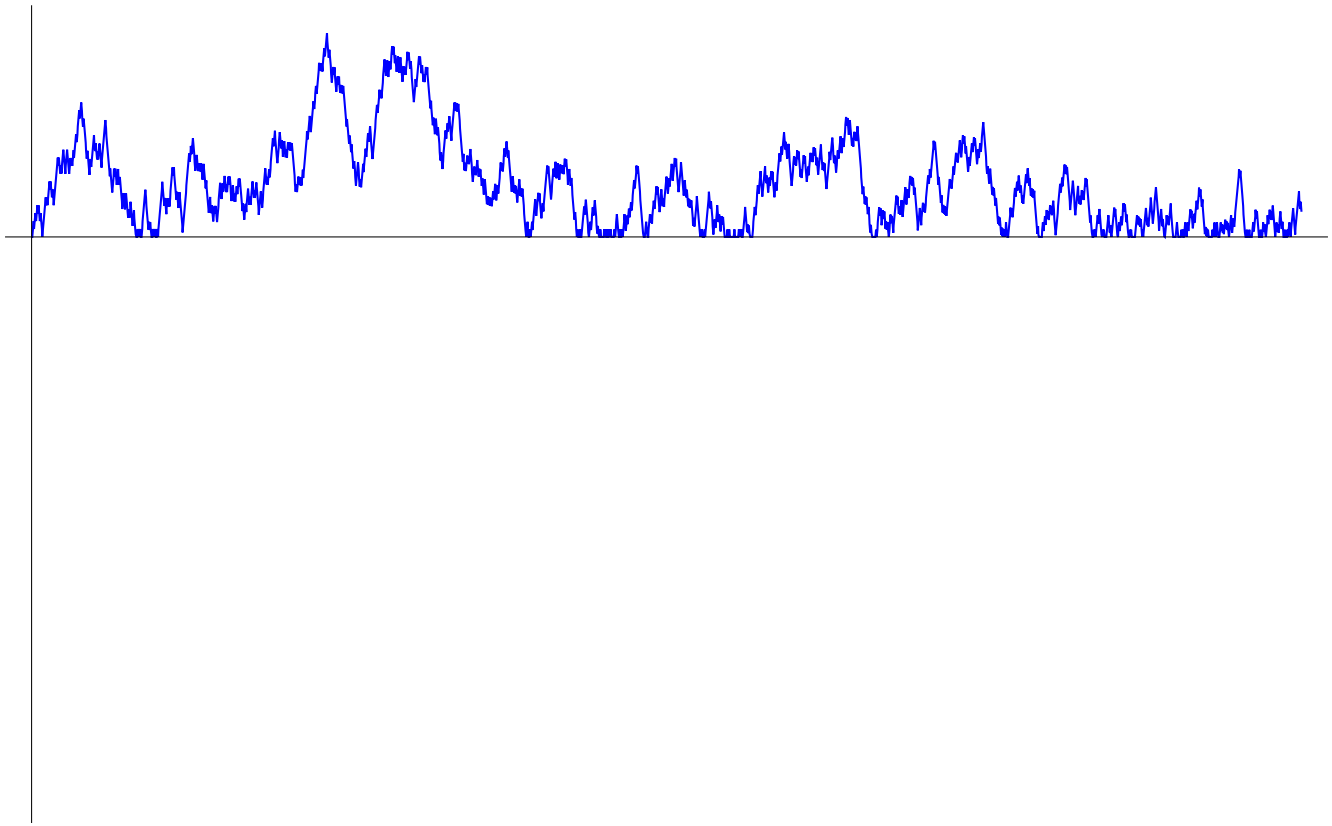


[Pictures by Louigi Addario-Berry]

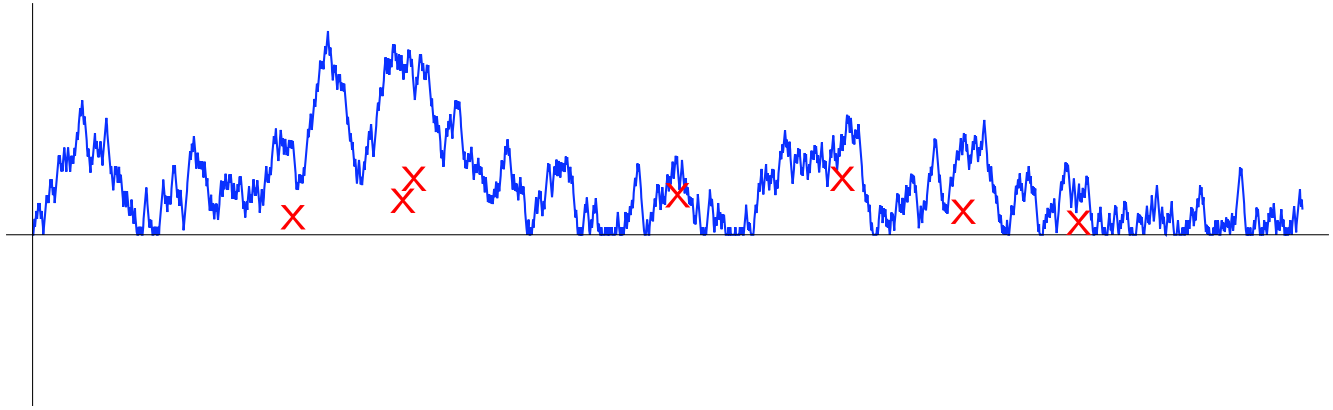
Limiting sizes and surpluses

Let $W_\lambda(t) = W(t) + \lambda t - \frac{t^2}{2}$, $t \geq 0$, where $(W(t), t \geq 0)$ is a standard Brownian motion.

Let $B_\lambda(t) = W_\lambda(t) - \min_{0 \leq s \leq t} W_\lambda(s)$ be the process reflected at its minimum.



[Pictures by Louigi Addario-Berry]



Decorate the picture with the points of a rate one Poisson process which fall above the x-axis and below the graph.

$\mathbf{C}_\lambda$ is the sequence of excursion-lengths of this process, in decreasing order.

$\mathbf{S}_\lambda$ is the sequence of numbers of points falling in the corresponding excursions.

The critical Erdős-Rényi random graph

Let $\mathbf{G}_\lambda^n = (G_\lambda^{n,1}, G_\lambda^{n,2}, \ldots)$ be the sequence of components of $\mathbb{G}(n, n^{-1} + \lambda n^{-4/3})$ in decreasing order of size, each considered as a metric space with the graph distance. Put mass $n^{-2/3}$ on each vertex.

The critical Erdős-Rényi random graph

Let $\mathbf{G}_\lambda^n = (G_\lambda^{n,1}, G_\lambda^{n,2}, \ldots)$ be the sequence of components of $\mathbb{G}(n, n^{-1} + \lambda n^{-4/3})$ in decreasing order of size, each considered as a metric space with the graph distance. Put mass $n^{-2/3}$ on each vertex.

Theorem (A-B, B, G). As $n \to \infty$,

$$n^{-1/3}\mathbf{G}_\lambda^n \xrightarrow{d} \mathscr{G}_\lambda,$$

where $\mathscr{G}_\lambda = (\mathscr{G}_\lambda^1, \mathscr{G}_\lambda^2, \ldots)$ is a sequence of measured real trees with vertex-identifications encoded by Aldous' marked limit process in decreasing order of length.

The critical Erdős-Rényi random graph

Let $\mathbf{G}_\lambda^n = (G_\lambda^{n,1}, G_\lambda^{n,2}, \ldots)$ be the sequence of components of $\mathbb{G}(n, n^{-1} + \lambda n^{-4/3})$ in decreasing order of size, each considered as a metric space with the graph distance. Put mass $n^{-2/3}$ on each vertex.

Theorem (A-B, B, G). As $n \to \infty$,

$$n^{-1/3}\mathbf{G}_\lambda^n \xrightarrow{d} \mathscr{G}_\lambda,$$

where $\mathscr{G}_\lambda = (\mathscr{G}_\lambda^1, \mathscr{G}_\lambda^2, \ldots)$ is a sequence of measured real trees with vertex-identifications encoded by Aldous' marked limit process in decreasing order of length.

Convergence is with respect to the distance

$$\mathrm{dist}^4_{\mathrm{GHP}}(\mathcal{A}, \mathcal{B}) := \left(\sum_{i=1}^{\infty} \mathrm{d}_{\mathrm{GHP}}(\mathcal{A}_i, \mathcal{B}_i)^4 \right)^{1/4}.$$

The limit of the random graph

- Each excursion of the process $(B_\lambda(t), t \geq 0)$ encodes a random real tree, which is a "spanning tree" for a limit component.

The limit of the random graph

- Each excursion of the process $(B_\lambda(t), t \geq 0)$ encodes a random real tree, which is a "spanning tree" for a limit component. (Unfortunately, it's not a *minimum* spanning tree!)

The limit of the random graph

- Each excursion of the process $(B_\lambda(t), t \geq 0)$ encodes a random real tree, which is a "spanning tree" for a limit component. (Unfortunately, it's not a *minimum* spanning tree!)
- These are not scaled Brownian CRT's, but random real trees whose distribution has been biased in favour of excursions with larger area.

The limit of the random graph

- Each excursion of the process $(B_\lambda(t), t \geq 0)$ encodes a random real tree, which is a "spanning tree" for a limit component. (Unfortunately, it's not a *minimum* spanning tree!)
- These are not scaled Brownian CRT's, but random real trees whose distribution has been biased in favour of excursions with larger area.
- The kth-longest excursion has length C_λ^k and so the corresponding tree is endowed with a uniform measure of that total mass.

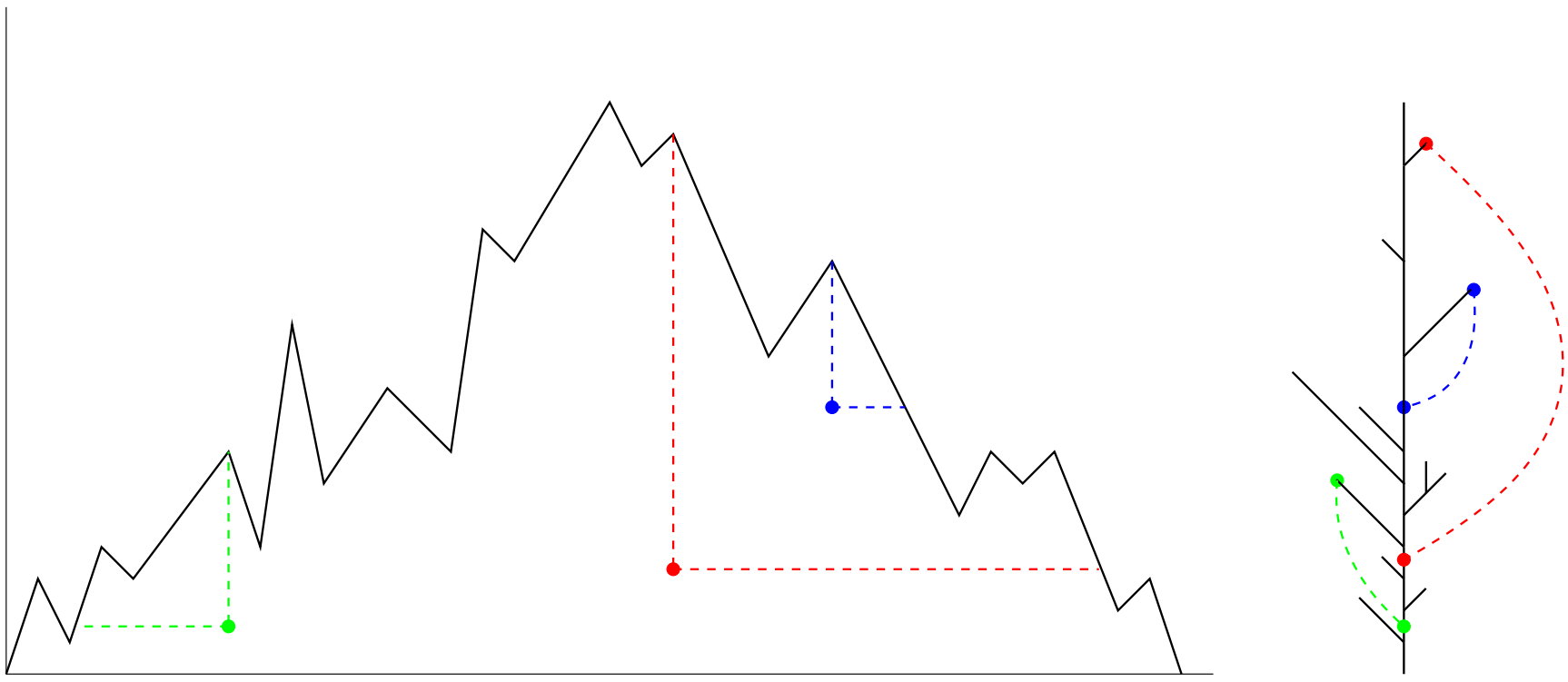
The limit of the random graph

- Each excursion of the process $(B_\lambda(t), t \geq 0)$ encodes a random real tree, which is a "spanning tree" for a limit component. (Unfortunately, it's not a *minimum* spanning tree!)
- These are not scaled Brownian CRT's, but random real trees whose distribution has been biased in favour of excursions with larger area.
- The kth-longest excursion has length C_λ^k and so the corresponding tree is endowed with a uniform measure of that total mass.
- In the limit, surplus edges correspond to vertex-identifications.

The limit of the random graph

- Each excursion of the process $(B_\lambda(t), t \geq 0)$ encodes a random real tree, which is a "spanning tree" for a limit component. (Unfortunately, it's not a *minimum* spanning tree!)
- These are not scaled Brownian CRT's, but random real trees whose distribution has been biased in favour of excursions with larger area.
- The kth-longest excursion has length C_λ^k and so the corresponding tree is endowed with a uniform measure of that total mass.
- In the limit, surplus edges correspond to vertex-identifications.
- In each excursion, the points of the Poisson process tell us where these vertex-identifications should occur.

Spanning tree and vertex identifications



Kruskal from Erdős-Rényi

To get from the state of the Erdős-Rényi process at a fixed time $p \in [0, 1]$ to the state of the Kruskal process, we need to break the cycles at high-weight edges.

Kruskal from Erdős-Rényi

Consider the Erdős-Rényi random graph at the point λ in the critical window: $\mathbf{G}_\lambda^n = (G_\lambda^{n,1}, G_\lambda^{n,2}, \ldots)$.

Kruskal from Erdős-Rényi

Consider the Erdős-Rényi random graph at the point λ in the critical window: $\mathbf{G}_\lambda^n = (G_\lambda^{n,1}, G_\lambda^{n,2}, \ldots)$.

Let $\mathbf{T}_\lambda^n = (T_\lambda^{n,1}, T_\lambda^{n,2}, \ldots)$ be the components in the Kruskal process (which have the same vertex-sets but no cycles).

Kruskal from Erdős-Rényi

Consider the Erdős-Rényi random graph at the point λ in the critical window: $\mathbf{G}_\lambda^n = (G_\lambda^{n,1}, G_\lambda^{n,2}, \ldots)$.

Let $\mathbf{T}_\lambda^n = (T_\lambda^{n,1}, T_\lambda^{n,2}, \ldots)$ be the components in the Kruskal process (which have the same vertex-sets but no cycles).

For fixed λ and k, we can construct $T_\lambda^{n,k}$ from $G_\lambda^{n,k}$ by running the cycle-breaking algorithm on the latter.

Kruskal from Erdős-Rényi

Reminder: cycle-breaking

- Start with all edges present. List them as $e_1, e_2, \ldots$ in decreasing order of weight.
- At step i, remove e_i as long as it does not disconnect the graph (i.e. as long as it lies in a cycle).
- Stop when no cycles remain.

Kruskal from Erdős-Rényi

Notice that only edges in cycles are affected by this procedure. So we could just ignore any edges which don't lie in cycles.

Kruskal from Erdős-Rényi

Notice that only edges in cycles are affected by this procedure. So we could just ignore any edges which don't lie in cycles. If we don't know the weights in advance then the highest-weight edge in a given cycle is equally likely to be any edge in that cycle.

Kruskal from Erdős-Rényi

Notice that only edges in cycles are affected by this procedure. So we could just ignore any edges which don't lie in cycles. If we don't know the weights in advance then the highest-weight edge in a given cycle is equally likely to be any edge in that cycle.

This tells us what the limiting analogue of the cycle-breaking procedure should be: repeatedly remove points chosen according to the uniform (Lebesgue) measure on the cycles in the metric space, until no cycles remain.

Kruskal from Erdős-Rényi

Notice that only edges in cycles are affected by this procedure. So we could just ignore any edges which don't lie in cycles. If we don't know the weights in advance then the highest-weight edge in a given cycle is equally likely to be any edge in that cycle.

This tells us what the limiting analogue of the cycle-breaking procedure should be: repeatedly remove points chosen according to the uniform (Lebesgue) measure on the cycles in the metric space, until no cycles remain.

(It's actually a little delicate to check that the cycle-breaking procedure passes nicely to the limit . . .)

A limit for the Kruskal process

Let $\mathbf{T}_\lambda^n = (\mathcal{T}_\lambda^{n,1}, \mathcal{T}_\lambda^{n,2}, \ldots)$ be the components of the Kruskal process, each endowed with the measure which assigns mass $n^{-2/3}$ to each vertex, and let $\mathscr{T}_\lambda = (\mathscr{T}_\lambda^1, \mathscr{T}_\lambda^2, \ldots)$ be the sequence of spaces obtained from $\mathscr{G}_\lambda$ by cycle-breaking.

A limit for the Kruskal process

Let $\mathbf{T}_\lambda^n = (T_\lambda^{n,1}, T_\lambda^{n,2}, \ldots)$ be the components of the Kruskal process, each endowed with the measure which assigns mass $n^{-2/3}$ to each vertex, and let $\mathscr{T}_\lambda = (\mathscr{T}_\lambda^1, \mathscr{T}_\lambda^2, \ldots)$ be the sequence of spaces obtained from $\mathscr{G}_\lambda$ by cycle-breaking.

Theorem. As $n \to \infty$,

$$\frac{1}{n^{1/3}} \mathbf{T}_\lambda^n \xrightarrow{d} \mathscr{T}_\lambda,$$

where convergence is with respect to the distance

$$\mathrm{dist}_{\mathrm{GHP}}^4(\mathcal{A}, \mathcal{B}) = \left(\sum_{i \geq 1} \mathrm{d}_{\mathrm{GHP}}(\mathcal{A}_i, \mathcal{B}_i)^4 \right)^{1/4}.$$

From Kruskal to the MST

- For fixed n, the process $(T_\lambda^{n,1}, \lambda \in \mathbb{R})$, which tracks the largest tree in the Kruskal forest, is eventually constant and equal to M^n, the MST of the complete graph.

From Kruskal to the MST

- For fixed n, the process $(T_\lambda^{n,1}, \lambda \in \mathbb{R})$, which tracks the largest tree in the Kruskal forest, is eventually constant and equal to M^n, the MST of the complete graph.
- For fixed λ, we have $n^{-1/3} T_\lambda^{n,1} \xrightarrow{d} \mathscr{T}_\lambda^1$ as $n \to \infty$ in $\mathrm{d_{GHP}}$.

From Kruskal to the MST

- For fixed n, the process $(T_\lambda^{n,1}, \lambda \in \mathbb{R})$, which tracks the largest tree in the Kruskal forest, is eventually constant and equal to M^n, the MST of the complete graph.
- For fixed λ, we have $n^{-1/3} T_\lambda^{n,1} \xrightarrow{d} \mathscr{T}_\lambda^1$ as $n \to \infty$ in $\mathrm{d_{GHP}}$.
- The process $(\mathscr{T}_\lambda^1, \lambda \in \mathbb{R})$ is eventually "increasing" in an appropriate sense.

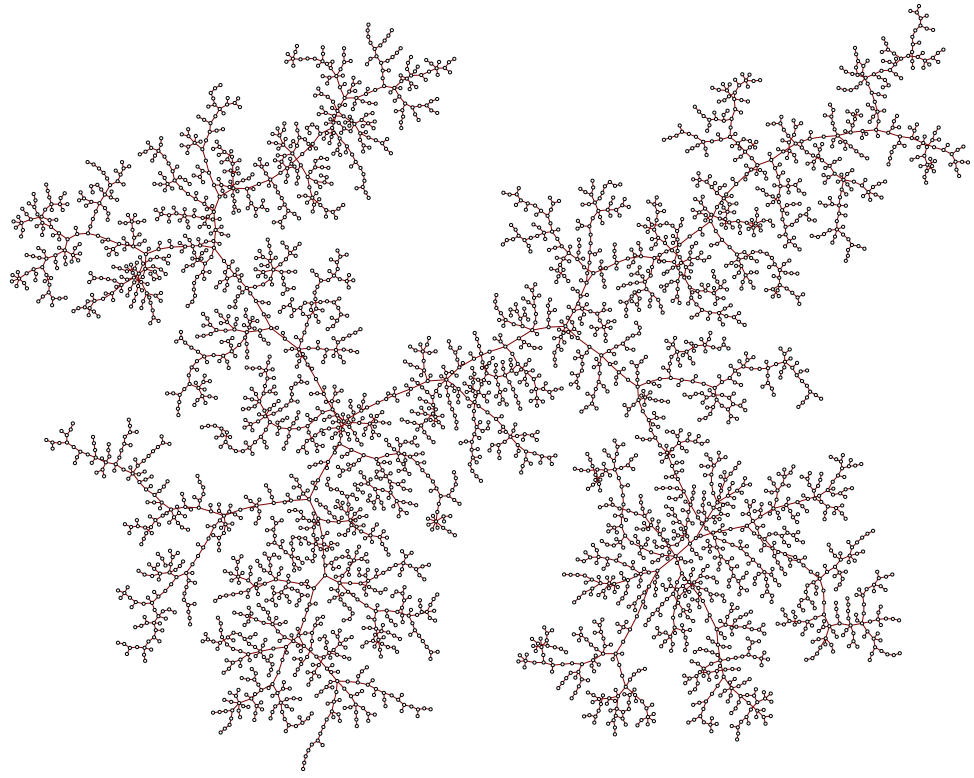
From Kruskal to the MST

- For fixed n, the process $(T_\lambda^{n,1}, \lambda \in \mathbb{R})$, which tracks the largest tree in the Kruskal forest, is eventually constant and equal to M^n, the MST of the complete graph.
- For fixed λ, we have $n^{-1/3} T_\lambda^{n,1} \xrightarrow{d} \mathscr{T}_\lambda^1$ as $n \to \infty$ in $\mathrm{d_{GHP}}$.
- The process $(\mathscr{T}_\lambda^1, \lambda \in \mathbb{R})$ is eventually "increasing" in an appropriate sense.
- As λ increases, we glue more and more little trees onto $\mathscr{T}_\lambda^1$. However, its diameter remains bounded. Moreover, the additional mass coming from the little trees gets spread out essentially uniformly over the tree.

From Kruskal to the MST

- For fixed n, the process $(T_\lambda^{n,1}, \lambda \in \mathbb{R})$, which tracks the largest tree in the Kruskal forest, is eventually constant and equal to M^n, the MST of the complete graph.
- For fixed λ, we have $n^{-1/3} T_\lambda^{n,1} \xrightarrow{d} \mathscr{T}_\lambda^1$ as $n \to \infty$ in $\mathrm{d_{GHP}}$.
- The process $(\mathscr{T}_\lambda^1, \lambda \in \mathbb{R})$ is eventually "increasing" in an appropriate sense.
- As λ increases, we glue more and more little trees onto $\mathscr{T}_\lambda^1$. However, its diameter remains bounded. Moreover, the additional mass coming from the little trees gets spread out essentially uniformly over the tree.
- The mass of $\mathscr{T}_\lambda^1$ diverges as $\lambda \to \infty$, so we renormalise to get total mass measure 1 for each λ. If we do this, we obtain that there exists a limiting measured metric space $\mathscr{M}$ as $\lambda \to \infty$.

The scaling limit of the MST



We obtain

$$\frac{1}{n^{1/3}} M^n \xrightarrow{d} \mathscr{M}$$

as $n \to \infty$, in the sense of $\mathrm{d}_{\mathrm{GHP}}$.

References

The continuum limit of critical random graphs
L. Addario-Berry, N. Broutin and C. Goldschmidt
Probability Theory and Related Fields 152, 3-4 (2012), pp.367-406.

Critical random graphs: limiting constructions and distributional properties
L. Addario-Berry, N. Broutin and C. Goldschmidt
Electronic Journal of Probability 15 (2010), pp.741-775.

The scaling limit of the minimum spanning tree of the complete graph
L. Addario-Berry, N. Broutin, C. Goldschmidt and G. Miermont
`arXiv:1301.1664` (2013+).