Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Monte-Carlo Estimation of s-t Reliability in Acyclic Networks

Rico Zenklusen

Institute for Operations Research, D–MATH, ETH Zurich
`rico.zenklusen@ifor.math.ethz.ch`

Joint work with Marco Laumanns

European Conference on Complex Systems, Dresden, 1 Oct 2007

Outline

Outline

1. **Introduction**
   - The *s*-*t* reliability problem
   - Motivation: spreading processes on networks
   - Complexity results
2. **New algorithm for s-t reliability estimation in DAGs**
   - The method
   - Worst-case bounds on the running time
   - Computational results
3. **Conclusions**

The s-t reliability problem

In classical network reliability problems we consider a graph whose edges fail independently of each other with some given probability.

Given is a network $G = (V, E, p)$ with

- V: set of vertices
- E: set of edges (directed, undirected or mixed)
- $p(e)$: failure probability of edge e $(\forall e \in E)$

and $s, t \in V$ two special terminals.

Definition (s-t reliability)

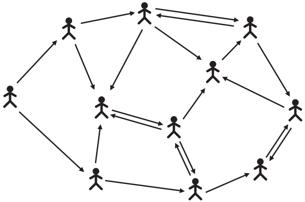
The s-t reliability $\mathrm{REL}_{s,t}(G)$ is the probability that the graph obtained from G after edge failures contains a path from s to t.

Outline

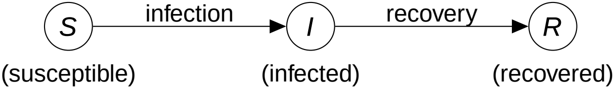
1. **Introduction**
   - The *s*-*t* reliability problem
   - Motivation: spreading processes on networks
   - Complexity results
2. **New algorithm for s-t reliability estimation in DAGs**
   - The method
   - Worst-case bounds on the running time
   - Computational results
3. **Conclusions**

SIR disease spreading model

- Continuous time model for the spread of a disease over the social network of a fixed population.

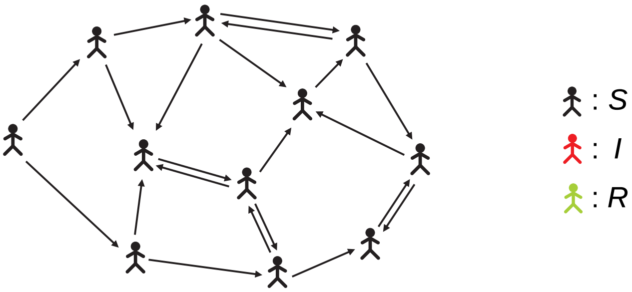


- At every time each individuum is in one of the states S,I or R.



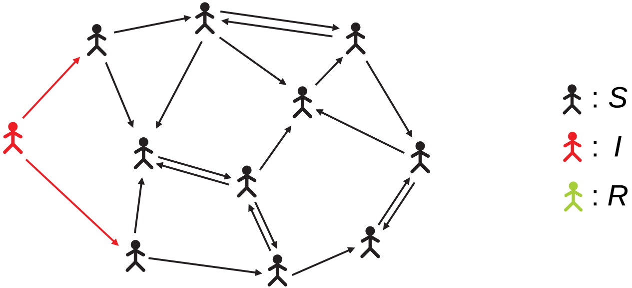
SIR model (example simulation)

Every node has an associated recovery time and every edge an associated infection rate.



SIR model (example simulation)

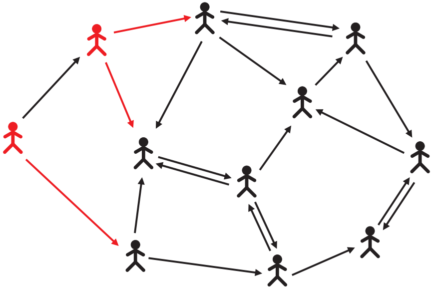
Every node has an associated recovery time and every edge an associated infection rate.



Initial outbreak of a disease.

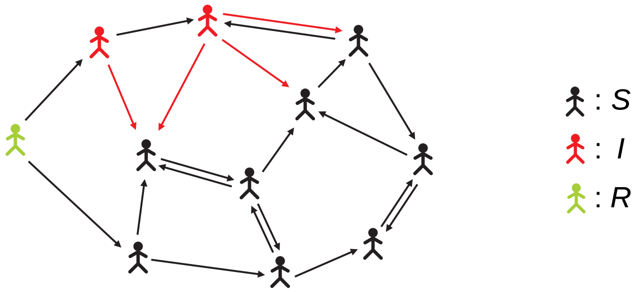
SIR model (example simulation)

Every node has an associated recovery time and every edge an associated infection rate.



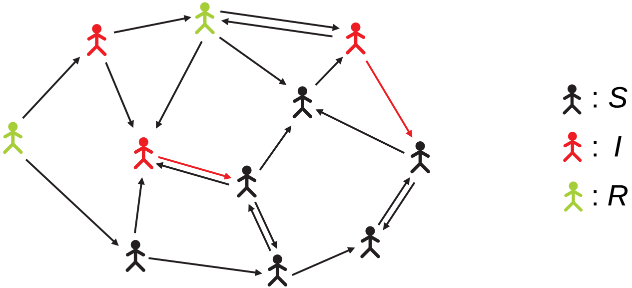
SIR model (example simulation)

Every node has an associated recovery time and every edge an associated infection rate.



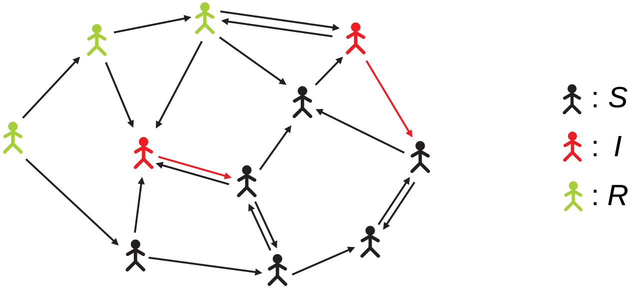
SIR model (example simulation)

Every node has an associated recovery time and every edge an associated infection rate.



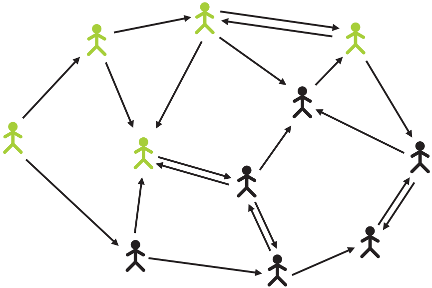
SIR model (example simulation)

Every node has an associated recovery time and every edge an associated infection rate.



SIR model (example simulation)

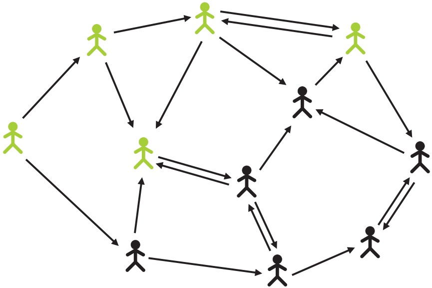
Every node has an associated recovery time and every edge an associated infection rate.



Final state of the disease spreading.

SIR model (example simulation)

Every node has an associated recovery time and every edge an associated infection rate.

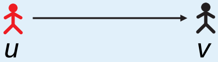


What is the probability that some fixed individuum gets infected?

Mapping SIR models to reliability problems

The final state of a disease spreading can be expressed by a classical reliability model on the social network with edge intactness probabilities defined as follows. [Grassberger, 1983]

$q(u,v) = 1 - p(u,v)$: Probability that u infects v in the situation where u just switched to state I and v is in state S.



Correspondence between the models

The set of all vertices reachable from the source s of the disease in the reliability model (after edge failures) corresponds to the part of the population who got infected.

s-t reliability models in SIR spreadings

Considering the spread of a disease starting at node s, the s-t reliability corresponds to the probability that node t gets infected.

We are interested in estimating the probability of the rare event that the disease spreads over a long distance from s to t.

Very small s-t reliabilities have to be estimated.
Standard Monte-Carlo approach is not efficient for this problem as for getting an "accurate" estimate, the number of samples N must satisfy:

$$N \propto \frac{1}{\mathrm{REL}_{s,t}(G)}$$

s-t reliability models in SIR spreadings

Considering the spread of a disease starting at node s, the s-t reliability corresponds to the probability that node t gets infected.

We are interested in estimating the probability of the rare event that the disease spreads over a long distance from s to t.

Very small s-t reliabilities have to be estimated.

Standard Monte-Carlo approach is not efficient for this problem as for getting an “accurate” estimate, the number of samples N must satisfy:

$$N \propto \frac{1}{\mathrm{REL}_{s,t}(G)}$$

s-t reliability models in SIR spreadings

Considering the spread of a disease starting at node *s*, the *s*-*t* reliability corresponds to the probability that node *t* gets infected.

> We are interested in estimating the probability of the rare event that the disease spreads over a long distance from *s* to *t*.

Very small *s*-*t* reliabilities have to be estimated.
Standard Monte-Carlo approach is not efficient for this problem as for getting an “accurate” estimate, the number of samples N must satisfy:

$$N \propto \frac{1}{\mathrm{REL}_{s,t}(G)}$$

Outline

1. **Introduction**
   - The *s*-*t* reliability problem
   - Motivation: spreading processes on networks
   - Complexity results
2. **New algorithm for s-t reliability estimation in DAGs**
   - The method
   - Worst-case bounds on the running time
   - Computational results
3. **Conclusions**

Exact computation of $\mathrm{REL}_{s,t}(G)$

Even on planar acyclic graphs with uniform edge failure probability (i.e. $p(e) = \overline{p}\ \forall e \in E$) the problem is $\#$P-complete.

Network classes where computation of $\mathrm{REL}_{s,t}(G)$ is easy

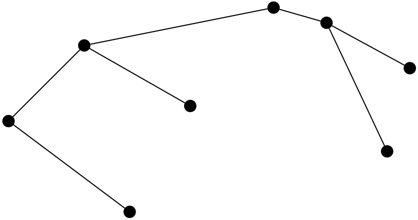
- Trees
- Series-parallel graphs
- Other graphs with bounded tree-width

Exact computation of $\mathrm{REL}_{s,t}(G)$

Even on planar acyclic graphs with uniform edge failure probability (i.e. $p(e) = \overline{p}\ \forall e \in E$) the problem is #P-complete.

Network classes where computation of $\mathrm{REL}_{s,t}(G)$ is easy

- Trees
- Series-parallel graphs
- Other graphs with bounded tree-width

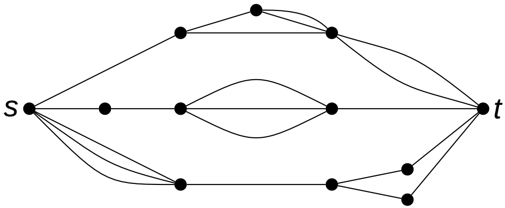


Exact computation of $\mathrm{REL}_{s,t}(G)$

Even on planar acyclic graphs with uniform edge failure probability (i.e. $p(e) = \bar{p}\ \forall e \in E$) the problem is $\#$P-complete.

Network classes where computation of $\mathrm{REL}_{s,t}(G)$ is easy

- Trees
- Series-parallel graphs
- Other graphs with bounded tree-width

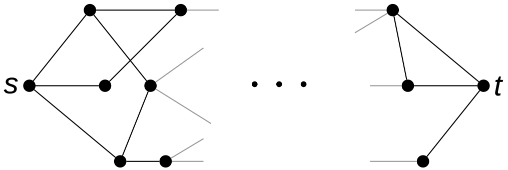


Exact computation of $\mathrm{REL}_{s,t}(G)$

Even on planar acyclic graphs with uniform edge failure probability (i.e. $p(e) = \overline{p}\ \forall e \in E$) the problem is #P-complete.

Network classes where computation of $\mathrm{REL}_{s,t}(G)$ is easy

- Trees
- Series-parallel graphs
- Other graphs with bounded tree-width



Approximating $\mathrm{REL}_{s,t}(G)$

Definition

An $\epsilon - \delta$ approximation is an algorithm returning with probability at least $1 - \delta$ a (relative) ϵ-approximation.

Open question

Are there $\epsilon - \delta$ approximations with running time polynomial in the input size and $1/\epsilon$ (i.e. FPRAS)?

Even practically useful algorithms for non-trivial large instances are missing.

Approximating $\mathrm{REL}_{s,t}(G)$

Definition

An $\epsilon - \delta$ approximation is an algorithm returning with probability at least $1 - \delta$ a (relative) ϵ-approximation.

Open question

Are there $\epsilon - \delta$ approximations with running time polynomial in the input size and $1/\epsilon$ (i.e. FPRAS)?

Even practically useful algorithms for non-trivial large instances are missing.

Outline

1. Introduction
   - The *s*-*t* reliability problem
   - Motivation: spreading processes on networks
   - Complexity results
2. **New algorithm for s-t reliability estimation in DAGs**
   - **The method**
   - Worst-case bounds on the running time
   - Computational results
3. Conclusions

General remarks

Goal

Construction of an algorithm allowing to estimate small *s*-*t* reliabilities in large-scale acyclic networks.

Main idea

Creation of a Monte-Carlo algorithm taking samples over a different space than 2^E, giving more influence on intact states (importance sampling).

Terminology and Notations

We represent the state of a network after edge failures by the set of intact (not failed) edges.

Notations

- $w(z)$ (for $z \subseteq E$): probability that state z occurs, i.e.

$$w(z) = \prod_{e \in z} q(e) \prod_{e \in E \setminus z} p(e)$$

- $\mathcal{P} \subseteq 2^E$: set of all paths from s to t.
- $\mathcal{P}(z) \subseteq \mathcal{P}$ (for $z \subseteq E$): set of $s - t$ paths contained in state z.
- A: set of all intact states, i.e. $A = \{a \subseteq E \mid \mathcal{P}(a) \neq \emptyset\}$

We thus have $w(A) := \sum_{a \in A} w(a) = \mathrm{REL}_{s,t}(G)$.

Terminology and Notations

We represent the state of a network after edge failures by the set of intact (not failed) edges.

Notations

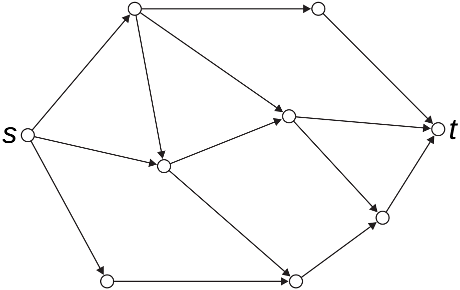
- $w(z)$ (for $z \subseteq E$): probability that state z occurs, i.e.

$$w(z) = \prod_{e \in z} q(e) \prod_{e \in E \setminus z} p(e)$$

- $\mathcal{P} \subseteq 2^E$: set of all paths from s to t.
- $\mathcal{P}(z) \subseteq \mathcal{P}$ (for $z \subseteq E$): set of $s - t$ paths contained in state z.
- A: set of all intact states, i.e. $A = \{a \subseteq E \mid \mathcal{P}(a) \neq \emptyset\}$

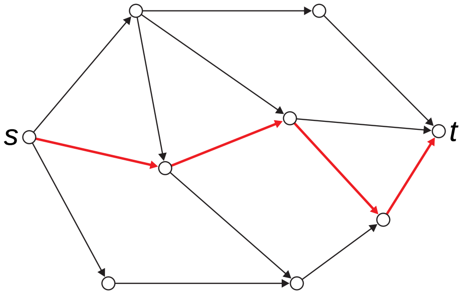
We thus have $w(A) := \sum_{a \in A} w(a) = \mathrm{REL}_{s,t}(G)$.

Sampling intact states



1. sample *s*-*t* path
2. sample remaining edges

Sampling intact states

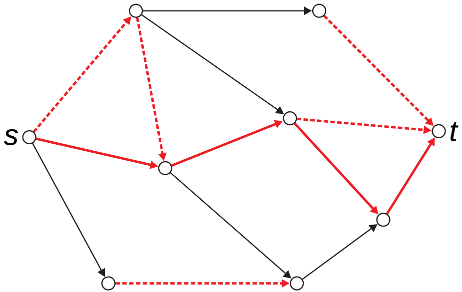


1. sample s-t path
2. sample remaining edges

A path $\gamma \in \mathcal{P}$ is sampled with probability proportional to

$$\prod_{e \in \gamma} q(\gamma).$$

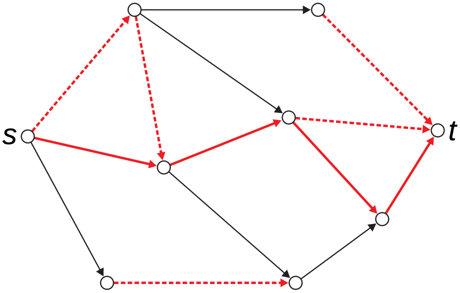
Sampling intact states



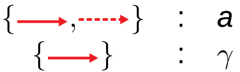
1. sample s-t path
2. sample remaining edges

Every edge $e \in E \setminus \gamma$ is intact with probability equal to $q(e)$.

Sampling intact states



1. sample s-t path
2. sample remaining edges



Sample space: $\Omega = \{(a, \gamma) \in A \times \mathcal{P} \mid \gamma \in \mathcal{P}(a)\}$

Each element $\omega = (\gamma, a) \in \Omega$ is sampled with probability $w(\omega) \propto w(a)$ (i.e., $w(\omega) = w(a) / \sum_{(a', \gamma') \in \Omega} w(a')$).

Remarks on Ω

Every element $a \in A$ has $|\mathcal{P}(a)|$ corresponding elements in Ω (overcounting).

$w(\Omega) = \sum_{\omega \in \Omega} w(\omega)$: mean number of intact paths in a state after edge failures (easy to calculate in acyclic graphs).

$w(\Omega)/w(A)$: mean number of intact paths in an intact state.

$w(\Omega)$ is often a reasonable estimate for low s-t reliabilities.

Approach

We will estimate $\mathrm{REL}_{s,t}(G) = w(A)$ by estimating $w(A)/w(\Omega)$ with a Monte-Carlo method and multiplying this estimate by $w(\Omega)$.

Remarks on Ω

Every element $a \in A$ has $|\mathcal{P}(a)|$ corresponding elements in Ω (overcounting).

$w(\Omega) = \sum_{\omega \in \Omega} w(\omega)$: mean number of intact paths in a state after edge failures (easy to calculate in acyclic graphs).

$w(\Omega)/w(A)$: mean number of intact paths in an intact state.

$w(\Omega)$ is often a reasonable estimate for low s-t reliabilities.

Approach

We will estimate $\mathrm{REL}_{s,t}(G) = w(A)$ by estimating $w(A)/w(\Omega)$ with a Monte-Carlo method and multiplying this estimate by $w(\Omega)$.

Introducing influence values on Ω for estimating $w(A)/w(\Omega)$

Influence values will compensate the overcounting.

We associate to each $\omega = (\gamma, a) \in \Omega$ an influence value $m(\omega) = 1/|\mathcal{P}(a)|$.

The mean influence value of Ω is $w(A)/w(\Omega)$.

Outline

Number of samples to draw

By the theory of zero-one estimators we have:

Theorem

The proposed algorithm is an $\epsilon - \delta$ approximation when the number of samples N satisfies

$$N \geq 4(e-2)\ln\left(\frac{2}{\delta}\right)\frac{1}{\epsilon^2}\frac{w(\Omega)}{\mathrm{REL}_{s,t}(G)}.$$

Finding upper bounds for $\frac{w(\Omega)}{\mathrm{REL}_{s,t}(G)}$ allows to eliminate the unknown value $\mathrm{REL}_{s,t}(G)$ in the above theorem.

Upper bounds for $\frac{w(\Omega)}{\mathrm{REL}_{s,t}(G)}$

An algebraic argument due to Karp and Luby gives the following bound.

$$\frac{w(\Omega)}{\mathrm{REL}_{s,t}(G)} \leq \prod_{e \in E}(1+q(e))$$

In the case of uniform edge failure probability $\overline{p} = 1 - \overline{q}$, the above bound can be sharpened to

$$\frac{w(\Omega)}{\mathrm{REL}_{s,t}(G)} \leq \min\left\{\left(\frac{2}{2-\overline{q}}\right)^{m-l_{min}}, 2^{1+\frac{\mu}{m}[(\overline{q}m+\ln(2))(\overline{q}m+\ln(2)+l_{max})]}\right\}$$

where l_{min} resp. l_{max} are the lengths of an $s-t$ path in G with minimal resp. maximal cardinality and μ is the maximal edge-to-vertex ratio over all subgraps of G.

Outline

1. Introduction
   - The s-t reliability problem
   - Motivation: spreading processes on networks
   - Complexity results

2. **New algorithm for s-t reliability estimation in DAGs**
   - The method
   - Worst-case bounds on the running time
   - Computational results

3. Conclusions

Instances

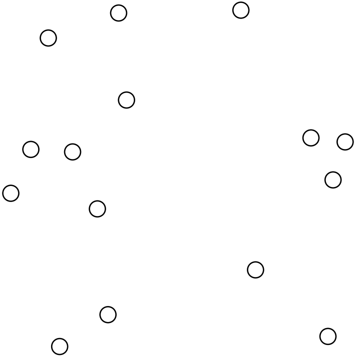
For numerical tests two types of random generators were used for the creation of directed acyclic networks:

- DEL (Delaunay) generator
- TC (Topological construction) generator

DEL (Delaunay) instances

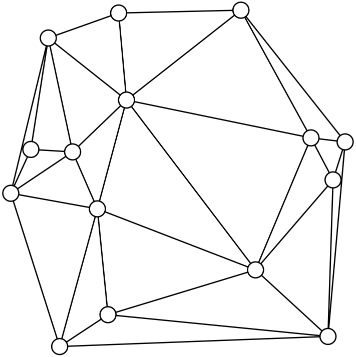
1. N vertices are chosen in $[0, 1]^2$.
2. Construction of Delaunay triangulation.
3. Two vertices with maximal Euclidian distance are declared as s and t.
4. All edges are directed "from s to t".
5. All arcs get uniform intactness probability $\overline{q}$.

DEL (Delaunay) instances



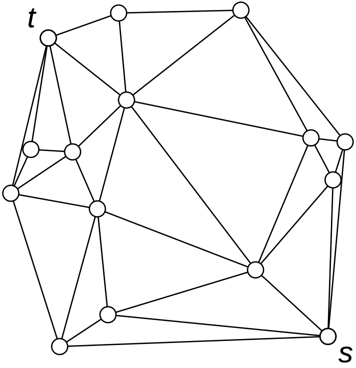
1. N vertices are chosen in $[0,1]^2$.
2. Construction of Delaunay triangulation.
3. Two vertices with maximal Euclidian distance are declared as s and t.
4. All edges are directed "from s to t".
5. All arcs get uniform intactness probability $\overline{q}$.

DEL (Delaunay) instances



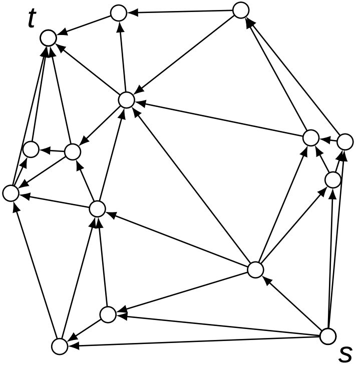
1. N vertices are chosen in $[0, 1]^2$.
2. Construction of Delaunay triangulation.
3. Two vertices with maximal Euclidian distance are declared as s and t.
4. All edges are directed “from s to t”.
5. All arcs get uniform intactness probability $\overline{q}$.

DEL (Delaunay) instances



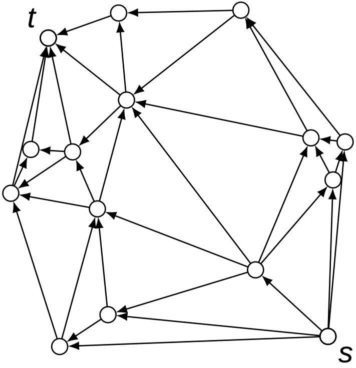
1. N vertices are chosen in $[0, 1]^2$.
2. Construction of Delaunay triangulation.
3. Two vertices with maximal Euclidian distance are declared as s and t.
4. All edges are directed “from s to t”.
5. All arcs get uniform intactness probability $\overline{q}$.

DEL (Delaunay) instances



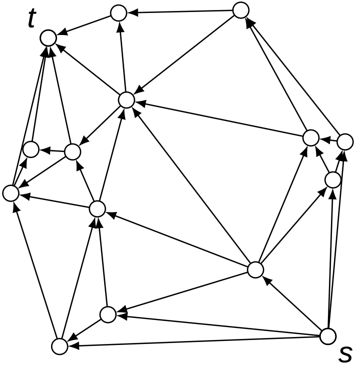
1. N vertices are chosen in $[0, 1]^2$.
2. Construction of Delaunay triangulation.
3. Two vertices with maximal Euclidian distance are declared as s and t.
4. All edges are directed "from s to t".
5. All arcs get uniform intactness probability $\overline{q}$.

DEL (Delaunay) instances



1. N vertices are chosen in $[0,1]^2$.
2. Construction of Delaunay triangulation.
3. Two vertices with maximal Euclidian distance are declared as s and t.
4. All edges are directed “from s to t”.
5. All arcs get uniform intactness probability $\overline{q}$.

DEL (Delaunay) instances



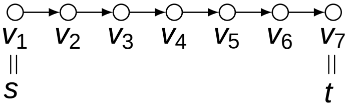
1. N vertices are chosen in $[0, 1]^2$.
2. Construction of Delaunay triangulation.
3. Two vertices with maximal Euclidian distance are declared as s and t.
4. All edges are directed “from s to t”.
5. All arcs get uniform intactness probability $\overline{q}$.

Parameters: $N, \overline{q}$

TC (Topological construction) instances

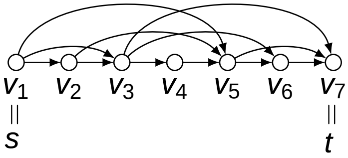
1. Start with a path over N vertices: $s = v_1 \to v_2 \to \cdots \to v_n = t$.
2. Every other possible arc is inserted with probability λ.
3. Edge (v_i, v_j) gets intactness probability uniformly distributed in $[0, 1/(j-i)^{1-\alpha}]$.

TC (Topological construction) instances



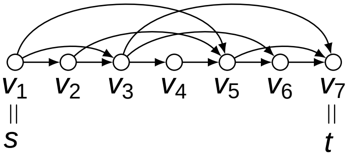
1. Start with a path over N vertices: $s = v_1 \to v_2 \to \cdots \to v_n = t$.
2. Every other possible arc is inserted with probability λ.
3. Edge (v_i, v_j) gets intactness probability uniformly distributed in $[0, 1/(j-i)^{1-\alpha}]$.

TC (Topological construction) instances



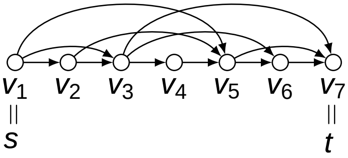
1. Start with a path over N vertices: $s = v_1 \to v_2 \to \cdots \to v_n = t$.
2. Every other possible arc is inserted with probability λ.
3. Edge (v_i, v_j) gets intactness probability uniformly distributed in $[0, 1/(j-i)^{1-\alpha}]$.

TC (Topological construction) instances



1. Start with a path over N vertices: $s = v_1 \to v_2 \to \cdots \to v_n = t$.
2. Every other possible arc is inserted with probability λ.
3. Edge (v_i, v_j) gets intactness probability uniformly distributed in $[0, 1/(j-i)^{1-\alpha}]$.

TC (Topological construction) instances

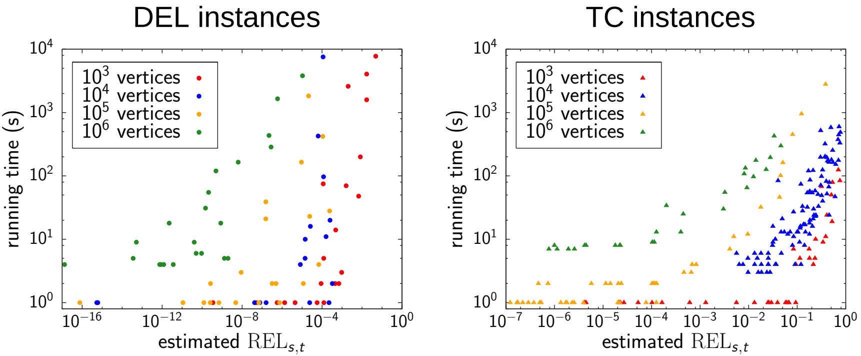


1. Start with a path over N vertices: $s = v_1 \to v_2 \to \cdots \to v_n = t$.
2. Every other possible arc is inserted with probability λ.
3. Edge (v_i, v_j) gets intactness probability uniformly distributed in $[0, 1/(j-i)^{1-\alpha}]$.

Parameters: N, λ, α

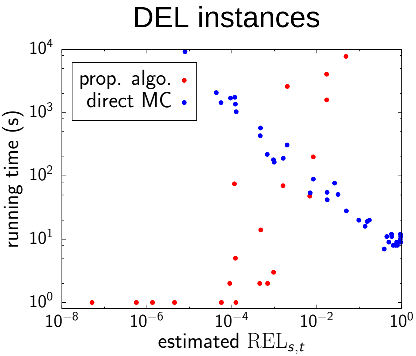
λ was chosen such that the expected degree of the vertices is equal to ten.

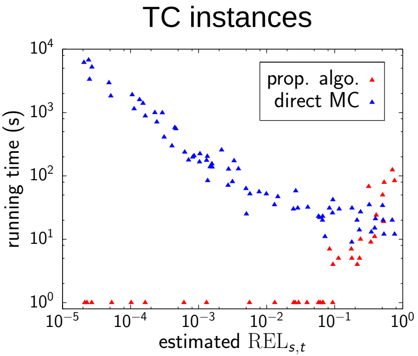
Running time of proposed algorithm



The dependence of the running time on ϵ and δ was essentially as predicted, i.e. $\propto \ln(1/\delta)/\epsilon^2 \rightarrow$ we fixed $\epsilon = 0.1$, $\delta = 0.001$.

Comparison with direct Monte-Carlo





Instances with 1000 vertices were used for these comparisons.

Conclusions

- A new algorithm for estimating small $s-t$ reliabilities in large acyclic networks was presented.
- Computational results show the successful application of the proposed algorithm on two types of large-scale instances and its advantage compared to the direct Monte-Carlo approach.
- For uniform edge failure probabilities, a worst-case bound on the number of samples to be drawn was given which sharpens a previous result by Karp and Luby.