Identifying Websites with Flow Simulation

Pierre Senellart

École normale supérieure
(Paris, France)

INRIA Futurs
(Orsay, France)

International Conference on Web Engineering
28 July 2005

What is a website?

- Simple idea: website = webserver
- But:
  - Some websites span over several webservers, or even over several DNS domains.
  - Some webservers host different websites.
- Limits of a website: a subjective notion.

What is a website?

- Simple idea: website = webserver
- But:
  - Some websites span over several webservers, or even over several DNS domains.
  - Some webservers host different websites.
- Limits of a website: a subjective notion.

What is a website?

- Simple idea: website = webserver
- But:
  - Some websites span over several webservers, or even over several DNS domains.
  - Some webservers host different websites.
- Limits of a website: a subjective notion.

What is a website?

- Simple idea: website = webserver
- But:
  - Some websites span over several webservers, or even over several DNS domains.
  - Some webservers host different websites.
- Limits of a website: a subjective notion.

What is a website?

- Simple idea: website = webserver
- But:
  - Some websites span over several webservers, or even over several DNS domains.
  - Some webservers host different websites.
- Limits of a website: a subjective notion.

Why is it important?

Several applications:

- Automatic archiving of websites (without asking the content providers the list of the webpages belonging to their site).
- SiteRank: a ranking measure for websites, as PageRank for webpages.

Why is it important?

Several applications:

- Automatic archiving of websites (without asking the content providers the list of the webpages belonging to their site).
- SiteRank: a ranking measure for websites, as PageRank for webpages.

Why is it important?

Several applications:

- Automatic archiving of websites (without asking the content providers the list of the webpages belonging to their site).
- SiteRank: a ranking measure for websites, as PageRank for webpages.

Outline

1 Introduction

2 Flow Simulation

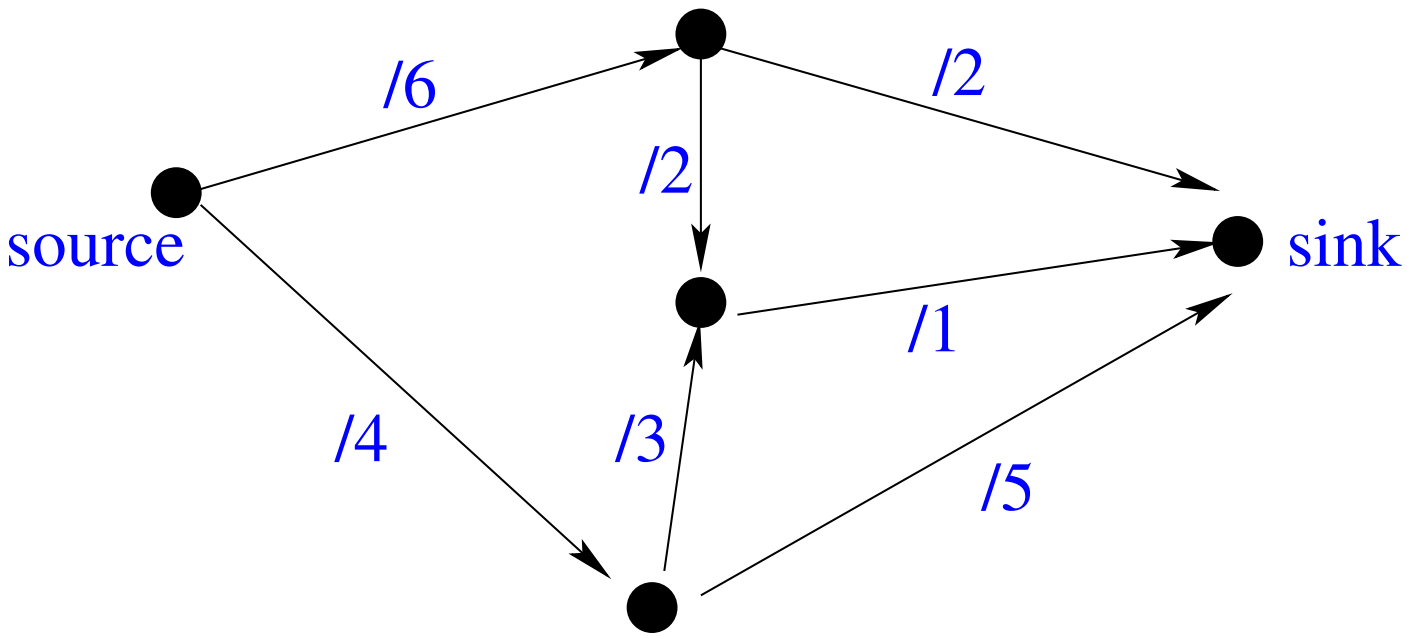
3 Seed Extension

4 Experiment

5 Conclusion

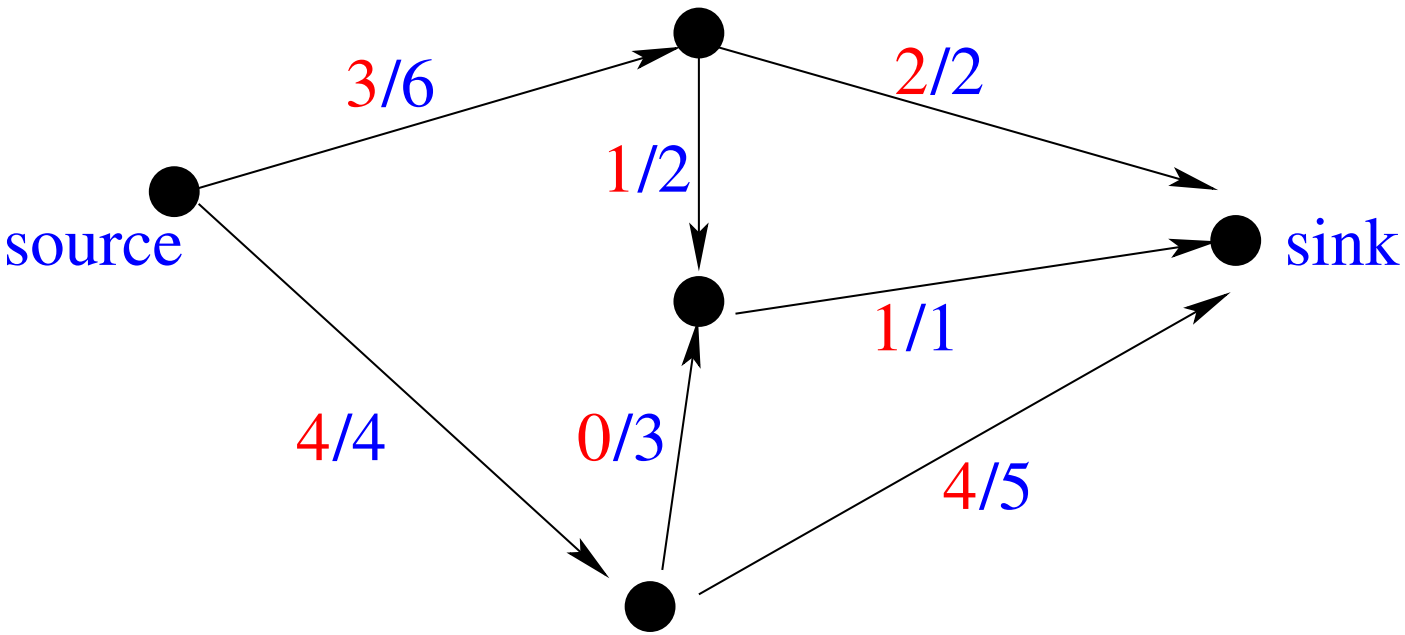
Maximum flow/Minimum cut

- Traffic network.
- Maximum flow $\equiv$ Minimum cut.



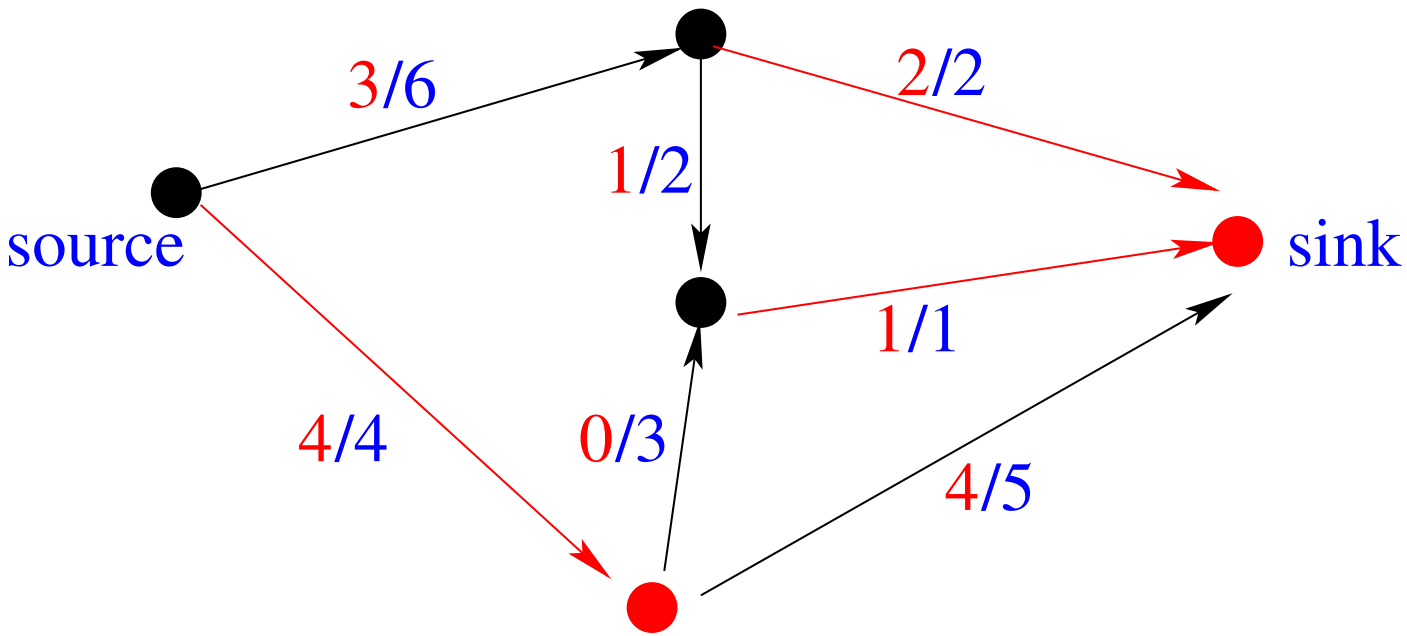
Maximum flow/Minimum cut

- Traffic network.
- Maximum flow $\equiv$ Minimum cut.



Maximum flow/Minimum cut

- Traffic network.
- Maximum flow $\equiv$ Minimum cut.



Preflow-Push algorithm

1. All nodes are assigned a height h: $h(source) = N$, $\forall k \neq source, h(k) = 0$ (N is the number of nodes)
2. Nodes with an overflow are visited, in any order.
   - If possible, the flow is pushed toward a lower node. Capacities of edges are respected.
   - Otherwise, the node is heigtened.

Theorem

The process converges, whatever the sequence of visited nodes may be. The maximum flow is obtained at the limit.

Preflow-Push algorithm

1. All nodes are assigned a height h: $h(source) = N$, $\forall k \neq source, h(k) = 0$ (N is the number of nodes)
2. Nodes with an overflow are visited, in any order.
   - If possible, the flow is pushed toward a lower node. Capacities of edges are respected.
   - Otherwise, the node is heigtened.

Theorem

The process converges, whatever the sequence of visited nodes may be. The maximum flow is obtained at the limit.

Preflow-Push algorithm

1. All nodes are assigned a height h: $h(source) = N$, $\forall k \neq source, h(k) = 0$ (N is the number of nodes)
2. Nodes with an overflow are visited, in any order.
   - If possible, the flow is pushed toward a lower node. Capacities of edges are respected.
   - Otherwise, the node is heigtened.

Theorem

The process converges, whatever the sequence of visited nodes may be. The maximum flow is obtained at the limit.

Preflow-Push algorithm

1. All nodes are assigned a height h: $h(source) = N$, $\forall k \neq source, h(k) = 0$ (N is the number of nodes)
2. Nodes with an overflow are visited, in any order.
   - If possible, the flow is pushed toward a lower node. Capacities of edges are respected.
   - Otherwise, the node is heigtened.

Theorem

The process converges, whatever the sequence of visited nodes may be. The maximum flow is obtained at the limit.

Preflow-Push algorithm

1. All nodes are assigned a height h: $h(source) = N$, $\forall k \neq source, h(k) = 0$ (N is the number of nodes)
2. Nodes with an overflow are visited, in any order.
   - If possible, the flow is pushed toward a lower node. Capacities of edges are respected.
   - Otherwise, the node is heigtened.

Theorem

The process converges, whatever the sequence of visited nodes may be. The maximum flow is obtained at the limit.

Adaptation to the World Wide Web

Website

Nodes of a traffic network delimited by a MaxFlow / MinCut.

- Nodes: webpages, progressively crawled.
- Edges: hyperlinks.
- Capacities: edit distance between URLs.
- A virtual source, pointing to a seed of pages with infinite capacity edges.
- A virtual sink, pointed by all nodes with very low capacity edges.

Adaptation to the World Wide Web

Website

Nodes of a traffic network delimited by a MaxFlow / MinCut.

- Nodes: webpages, progressively crawled.
- Edges: hyperlinks.
- Capacities: edit distance between URLs.
- A virtual source, pointing to a seed of pages with infinite capacity edges.
- A virtual sink, pointed by all nodes with very low capacity edges.

Adaptation to the World Wide Web

Website

Nodes of a traffic network delimited by a MaxFlow / MinCut.

- Nodes: webpages, progressively crawled.
- Edges: hyperlinks.
- Capacities: edit distance between URLs.
- A virtual source, pointing to a seed of pages with infinite capacity edges.
- A virtual sink, pointed by all nodes with very low capacity edges.

Adaptation to the World Wide Web

Website

Nodes of a traffic network delimited by a MaxFlow / MinCut.

- Nodes: webpages, progressively crawled.
- Edges: hyperlinks.
- Capacities: edit distance between URLs.
- A virtual source, pointing to a seed of pages with infinite capacity edges.
- A virtual sink, pointed by all nodes with very low capacity edges.

Adaptation to the World Wide Web

Website

Nodes of a traffic network delimited by a MaxFlow / MinCut.

- Nodes: webpages, progressively crawled.
- Edges: hyperlinks.
- Capacities: edit distance between URLs.
- A virtual source, pointing to a seed of pages with infinite capacity edges.
- A virtual sink, pointed by all nodes with very low capacity edges.

Adaptation to the World Wide Web

Website

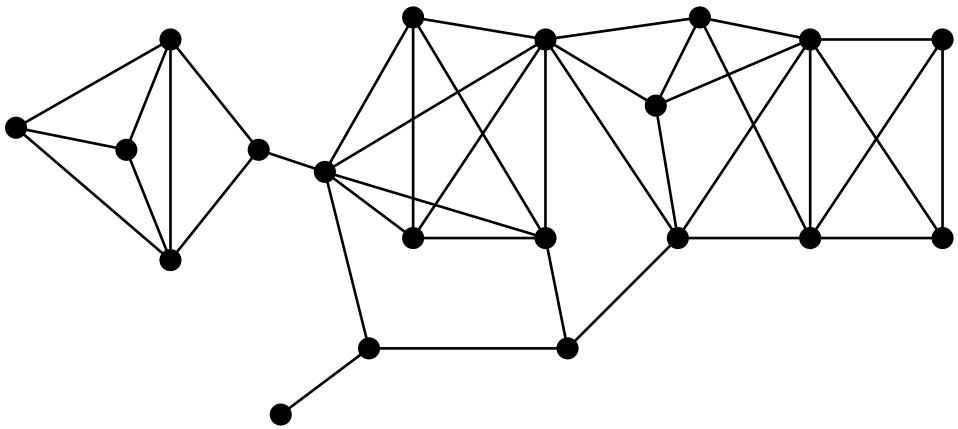
Nodes of a traffic network delimited by a MaxFlow / MinCut.

- Nodes: webpages, progressively crawled.
- Edges: hyperlinks.
- Capacities: edit distance between URLs.
- A virtual source, pointing to a seed of pages with infinite capacity edges.
- A virtual sink, pointed by all nodes with very low capacity edges.

Markov CLustering algorithm (MCL)

MCL

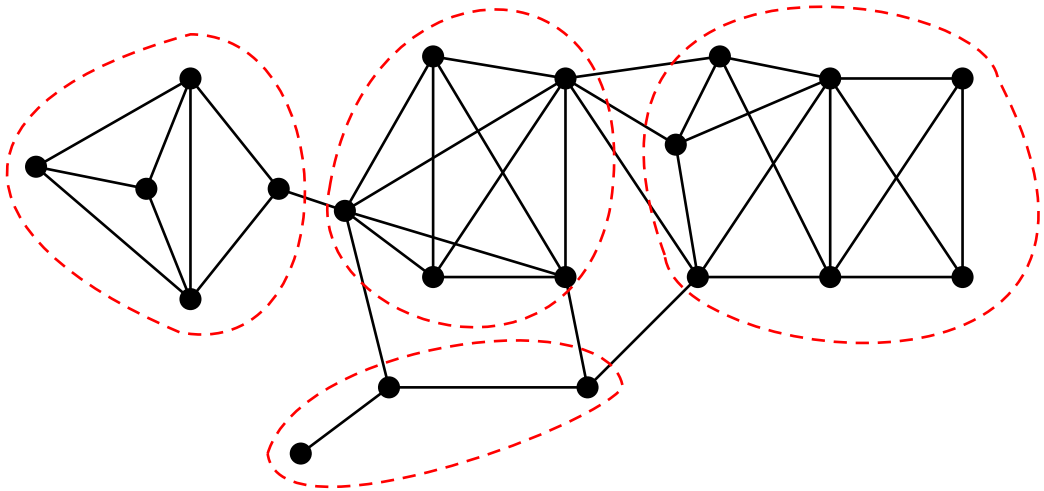
An off-line graph clustering algorithm



Markov CLustering algorithm (MCL)

MCL

An off-line graph clustering algorithm



Flow simulation from MCL clusters

Process

1. MCL Clustering of a large, a priori relevant, portion of the Web graph.
2. Identification of the most relevant cluster(s).
3. Flow simulation starting from this cluster.

Advantages over MCL alone

- Dynamic discovery of clusters.
- Use the fact that the graph is directed.

Flow simulation from MCL clusters

Process

1. MCL Clustering of a large, a priori relevant, portion of the Web graph.
2. Identification of the most relevant cluster(s).
3. Flow simulation starting from this cluster.

Advantages over MCL alone

- Dynamic discovery of clusters.
- Use the fact that the graph is directed.

Flow simulation from MCL clusters

Process

1. MCL Clustering of a large, a priori relevant, portion of the Web graph.
2. Identification of the most relevant cluster(s).
3. Flow simulation starting from this cluster.

Advantages over MCL alone

- Dynamic discovery of clusters.
- Use the fact that the graph is directed.

Flow simulation from MCL clusters

Process

1. MCL Clustering of a large, a priori relevant, portion of the Web graph.
2. Identification of the most relevant cluster(s).
3. Flow simulation starting from this cluster.

Advantages over MCL alone

- Dynamic discovery of clusters.
- Use the fact that the graph is directed.

Flow simulation from MCL clusters

Process

1. MCL Clustering of a large, a priori relevant, portion of the Web graph.
2. Identification of the most relevant cluster(s).
3. Flow simulation starting from this cluster.

Advantages over MCL alone

- Dynamic discovery of clusters.
- Use the fact that the graph is directed.

Flow simulation from MCL clusters

Process

1. MCL Clustering of a large, a priori relevant, portion of the Web graph.
2. Identification of the most relevant cluster(s).
3. Flow simulation starting from this cluster.

Advantages over MCL alone

- Dynamic discovery of clusters.
- Use the fact that the graph is directed.

Experiment description

GEMO website identification

1. Crawl of a large part of `*.inria.fr/*`
2. MCL clustering of the obtained graph
3. Identification of the GEMO cluster
4. Flow Simulation from this cluster

Experiment description

GEMO website identification

1. Crawl of a large part of `*.inria.fr/*`
2. MCL clustering of the obtained graph
3. Identification of the GEMO cluster
4. Flow Simulation from this cluster

Experiment description

GEMO website identification

1. Crawl of a large part of `*.inria.fr/*`
2. MCL clustering of the obtained graph
3. Identification of the GEMO cluster
4. Flow Simulation from this cluster

Experiment description

GEMO website identification

1. Crawl of a large part of `*.inria.fr/*`
2. MCL clustering of the obtained graph
3. Identification of the GEMO cluster
4. Flow Simulation from this cluster

Results

	Pages	Precision	Recall
Flow Simulation	8	87.5%	1.3%
MCL	320	99.7%	33.0%
MCL + Flow Sim.	788	90.4%	86.4%
`http://www-rocq.inria.fr/verso/*`	221	100%	44.4%
`http://*.inria.fr/verso/*`	683	100%	68.6%

Summary

- Website: subjective, non-obvious but important notion.
- Flow simulation used to discover the boundaries of a website.
- Best results obtained by combining off-line graph clustering and on-line flow simulation.

Summary

- Website: subjective, non-obvious but important notion.
- Flow simulation used to discover the boundaries of a website.
- Best results obtained by combining off-line graph clustering and on-line flow simulation.

Summary

- Website: subjective, non-obvious but important notion.
- Flow simulation used to discover the boundaries of a website.
- Best results obtained by combining off-line graph clustering and on-line flow simulation.

Perspectives

- On-line MCL computation.
- Efficient crawling strategy.
- Combination with semantic methods.

Perspectives

- On-line MCL computation.
- Efficient crawling strategy.
- Combination with semantic methods.

Perspectives

- On-line MCL computation.
- Efficient crawling strategy.
- Combination with semantic methods.