UNIVERSITY OF CAMBRIDGE

Representative Subgraph Sampling using MCMC Methods

Karsten Borgwardt

MLG 2008, Helsinki, July 4, 2008

Machine Learning Group
Department of Engineering
University of Cambridge

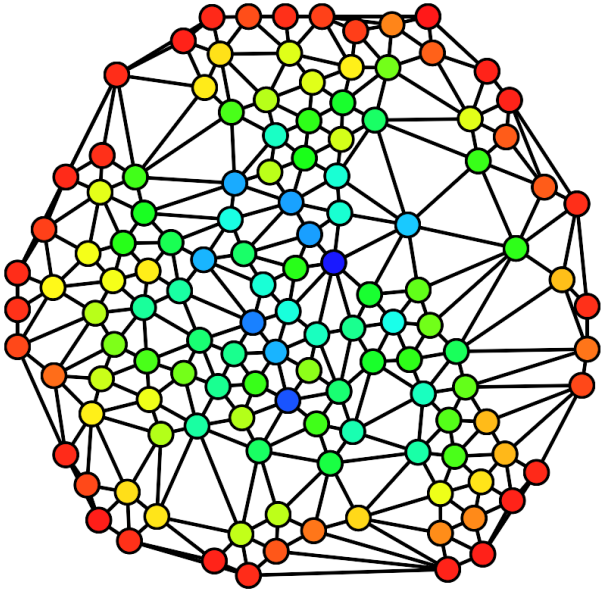
joint work with Christian Hübler, Hans-Peter Kriegel, and Zoubin Ghahramani

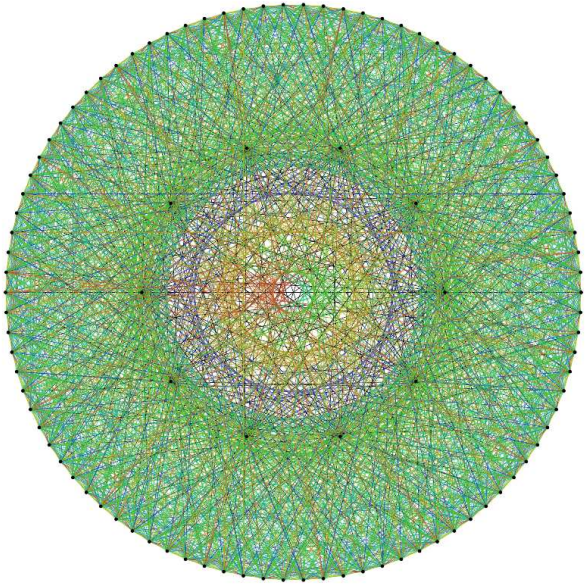
Graphs are growing...

UNIVERSITY OF CAMBRIDGE

Increase in graph sizes

- Classic chemoinformatics: dozens of nodes
- Systems biology: thousands of nodes
- Internet and online communities: hundreds of thousands and millions of nodes





Graph size is a challenge

UNIVERSITY OF CAMBRIDGE

Algorithmic challenge

- Basic operations on graphs are expensive:
  - graph isomorphism: Checking identity of two graphs
  - subgraph isomorphism: Checking if graph G includes graph S
- No polynomial runtime algorithm (in the number of nodes n) is known for any of these basic operations
- As a consequence, runtime for all algorithms that use these basic operations scales exponentially with the size of the graph

Potential solutions

UNIVERSITY OF CAMBRIDGE

Make your algorithms more efficient

- look at special subclasses of graphs
- treat graphs as sets
- map graphs to feature vectors

Make your graphs smaller

- find a smaller graph S that preserves key characteristics of the original graph G.
- Strategy 1: generate S synthetically
- Strategy 2: sample S from G
- Strategy 2 is the topic of this talk.

Related work I

UNIVERSITY OF CAMBRIDGE

Random Node Selection (Leskovec & Faloutsos, 2006)

- Randomly pick n' nodes from the graph
- Different Variants:
  - Uniform random sampling
  - Sampling according to PageRank weight
  - Sampling according to node degree

Random Edge Selection

- Randomly pick edges from the graph
  - Uniform random sampling
  - Pick combinations of nodes and edges
  - Combination of both

Related work II

UNIVERSITY OF CAMBRIDGE

Sampling by exploration

- Different variants of picking a node randomly and then exploring its neighborhood
- Forest Fire
  - randomly pick a node v from the graph
  - randomly pick a random number x
  - select x neighbors of v: $w_1, \ldots, w_x$ (edges not visited before)
  - recursively repeat this for all $w_1, \ldots, w_x$
  - terminate when the desired number of nodes (n') nodes have been 'burnt'

State-of-the-art with respect to sample size: Good samples down to 15% of the original graph size

Optimisation problem

UNIVERSITY OF CAMBRIDGE

Goal

- Optimal subgraph sample is

$$\text{argmin}_{|S|=n'} \Delta(\sigma(S), \sigma(G)) \tag{1}$$

- G is the original graph
- S the subgraph sample
- $\sigma(X)$ is a topological property of graph X
- Δ is a distance function on these topological properties
- n' is the desired size of the subgraph sample that has to be pre-specified

Metropolis graph sampling

UNIVERSITY OF CAMBRIDGE

Algorithm

- Use Metropolis to search the space of subgraphs of G
- Initially pick one subgraph sample S with n' nodes randomly
- Iterate the following steps until convergence
  - Remove one node from S
  - Randomly add a new node to $S \rightarrow S'$
  - Compute the likelihood ratio $a = \frac{\varrho^*(S')}{\varrho^*(S)}$

    if $a \geq 1$: accept transition: $S := S'$
  - if $a < 1$: accept transition: $S := S'$ with probability a

    reject transition: $S := S$ with probability $1 - a$

How to define $\varrho^*(S)$ to get a 'good' subgraph sample?

Defining $\varrho^*(S)$

UNIVERSITY OF CAMBRIDGE

Two roles

- ϱ is a probability distribution that Metropolis samples from (ϱ^* proportional to ϱ)
- ϱ^* shall reflect subgraph quality

Definition

- We define $\varrho^*(S)$ to be the inverse of a distance Δ between the sample S and the original graph G wrt to some graph property σ (to the power of p):

$$\varrho^*(S) := \frac{1}{\Delta_{G,\sigma}(S)^p} = \frac{1}{\Delta(\sigma(S), \sigma(G))^p}, \tag{2}$$

where $p \in \mathbb{R}^+$ and $p \gg 0$.

Chaining

UNIVERSITY OF CAMBRIDGE

Strategy

- Observation: Good samples are often connected subgraphs
- Idea: Restrict search space to connected subgraphs
- Allow transition from S to S' only if S' is connected

Runtime

- Size of search space is reduced
- Additional check of connectivity is required for each transition

Simulated annealing

UNIVERSITY OF CAMBRIDGE

Strategy (Kirkpatrick et al., 1983)

- Metropolis-variant for global optimization
- ϱ^* is controlled by temperature parameter T (in denominator):
- T slowly decreases to zero ('initially huge jumps, later on small steps')

We define the temperature-dependent density as

$$\varrho^*_{p,T}(S) = e^{\frac{\log \varrho^*_p(S)}{T}} = e^{log\Delta_{G,\sigma}(S)^{(-\frac{p}{T})}} = \Delta_{G,\sigma}(S)^{(-\frac{p}{T})}$$

so that a new acceptance-probability follows

$$a(S, S') = min(1, \frac{\varrho^*_{p,T}(S')}{\varrho^*_{p,T}(S)}) = min(1, (\frac{\Delta_{G,\sigma}(S)}{\Delta_{G,\sigma}(S')})^{\frac{p}{T}})$$

assuming a symmetric proposal distribution.

Experiments

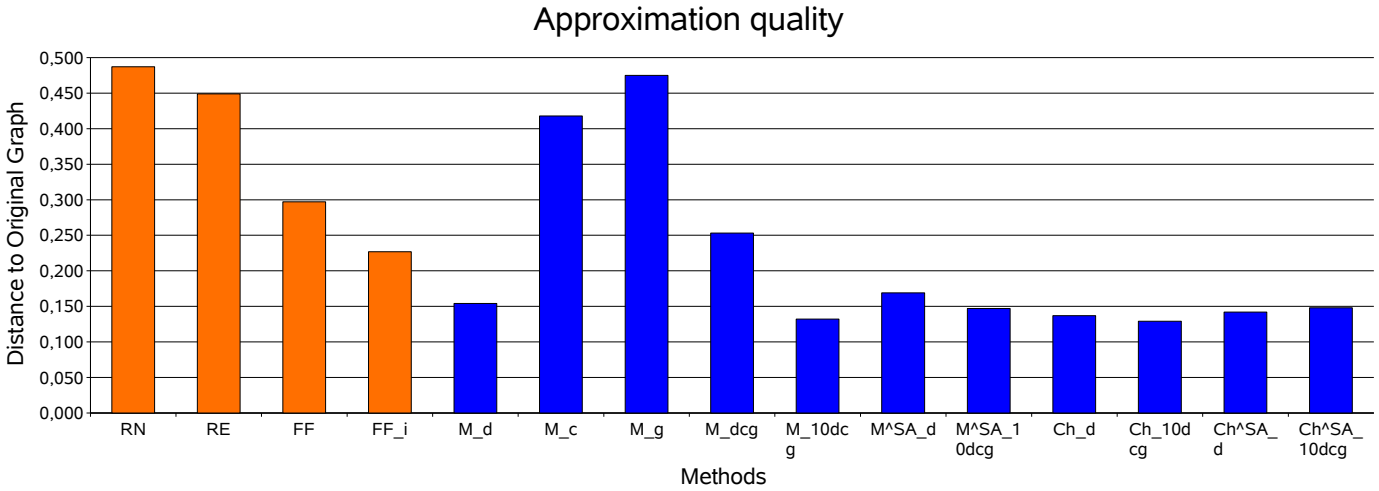
UNIVERSITY OF CAMBRIDGE

Setup

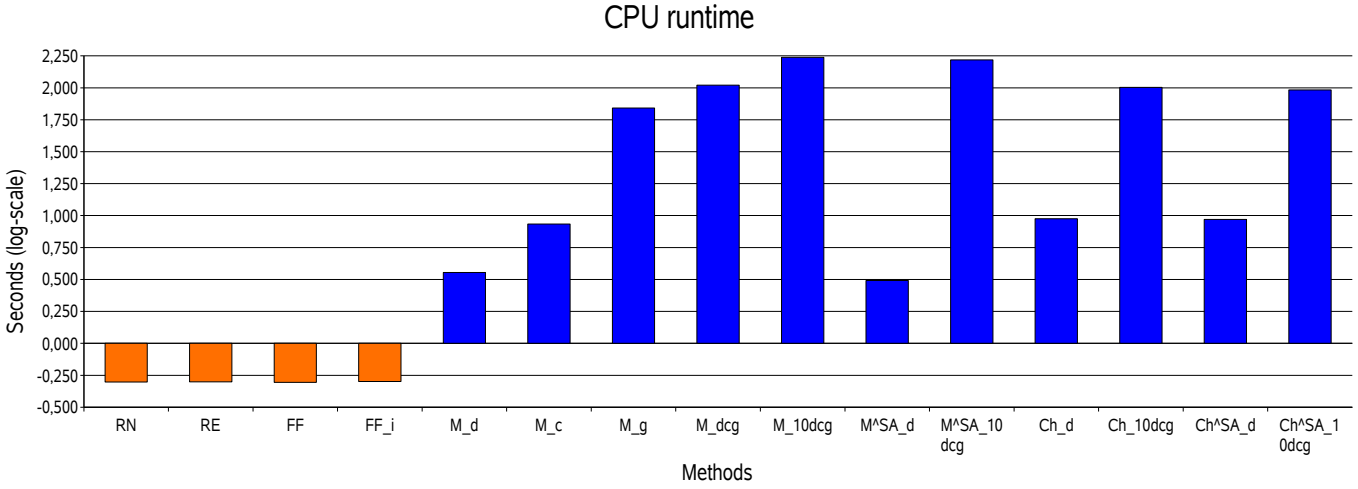
- Data: 5 datasets (329 to 75,879 nodes)
- Goal: Find representative subgraph with $n' = 100$ nodes
- Our Methods:
  - Metropolis M
  - Simulated Annealing SA
  - Chaining Ch
- Graph properties σ (for sampling):
  - degree distribution d
  - clustering coefficient c
  - graphlet distribution g
  - weighted combination of all three dcg
- Compare to Random Edge (RE), Random Node (RN), Forest Fire (FF,FF$_i$)

Sample quality

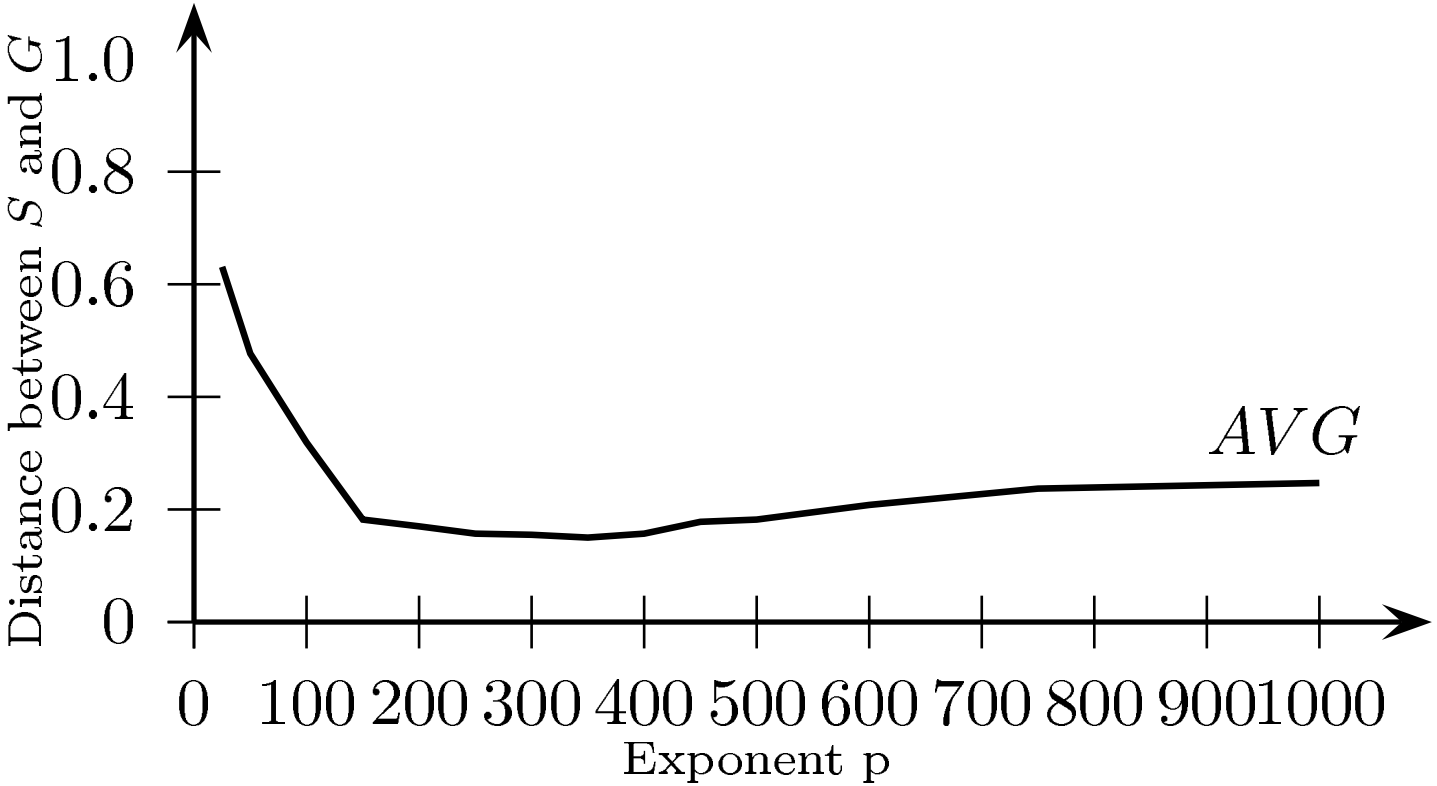
- measured in terms of graph properties (degree distribution, diameter, clustering coefficient, graphlet distribution)
- Results here are averages over all these four criteria



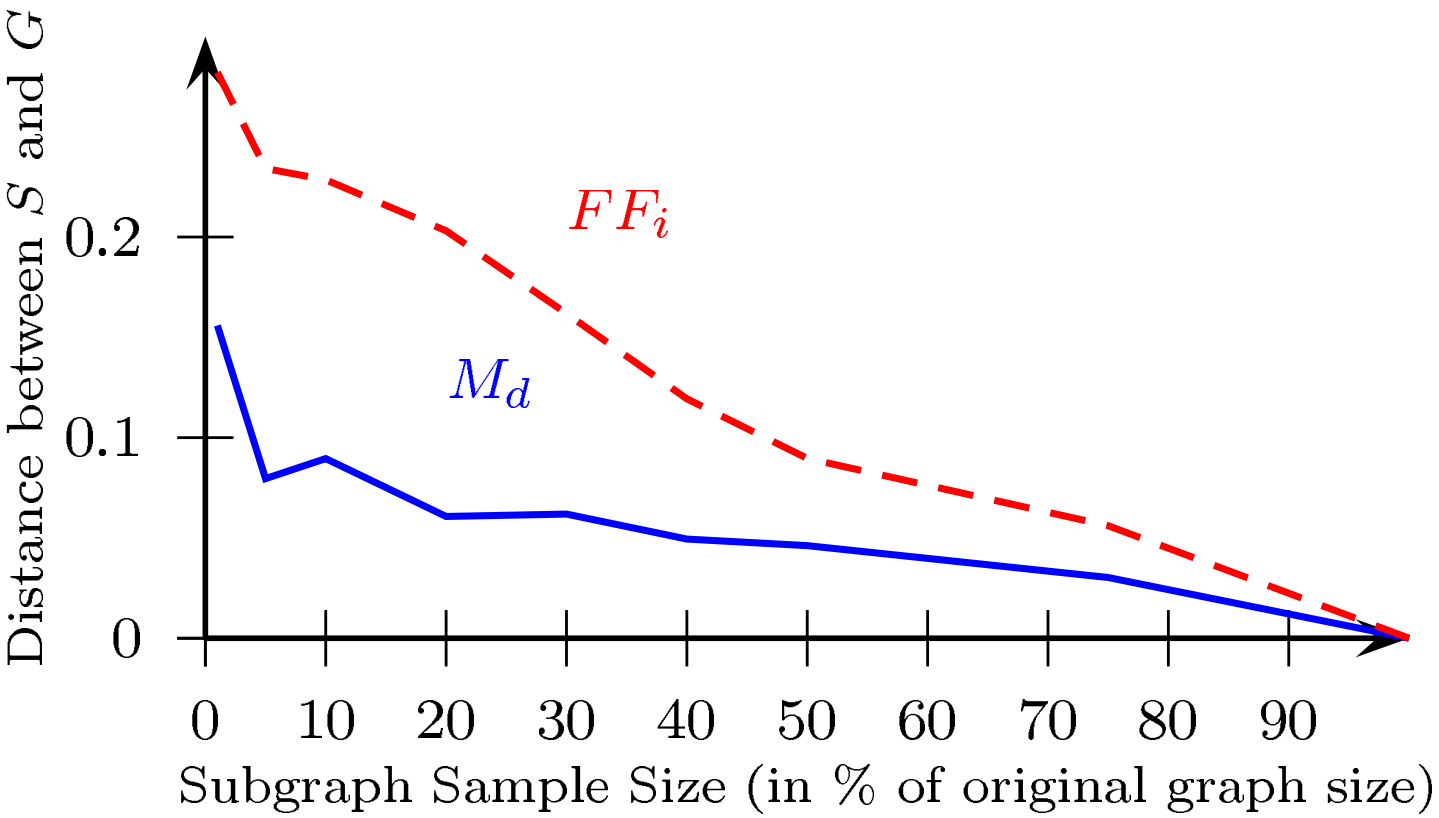
CPU runtime


CPU runtime
Seconds (log-scale)
2,250
2,000
1,750
1,500
1,250
1,000
0,750
0,500
0,250
0,000
-0,250
-0,500
RN
RE
FF
FF_i
M_d
M_c
M_g
M_dcg
M_10dcg
M^SA_d
M^SA_10 dcg
Ch_d
Ch_10dcg
Ch^SA_d
Ch^SA_1 0dcg
Methods

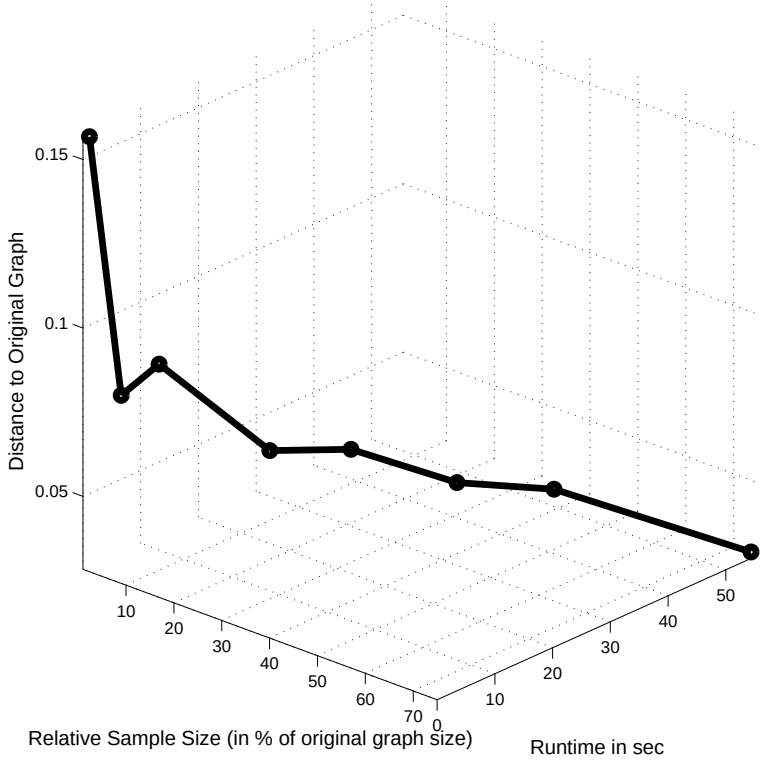
Impact of exponent



Impact of sample size



Distance vs. Runtime vs. Size



Summary

UNIVERSITY OF CAMBRIDGE

Results and Observations

- Metropolis methods for Representative Subgraph Sampling
- High-quality samples down to 0.15% of original graph size
- Metropolis algorithm approximating the degree distribution best combines efficiency and sample quality

Applications and Future Work

- Subgraph samples can be used directly for visualization, graph kernel computation, and frequent subgraph mining
- Future work will look into representative subgraph mining for tasks that only indirectly rely on topol. properties (community detection, graph clustering)

The End

UNIVERSITY OF CAMBRIDGE

THANK YOU! QUESTIONS?