The Maximum Independent Set Problem is Easy

Darren Strash
Shonan Meeting 144 | March 5, 2019
The Maximum Independent Set Problem is Easy
(Except When it Isn’t)

Darren Strash
Shonan Meeting 144 | March 5, 2019
Applications

Computer vision:

→ Image segmentation

† [Link to website](http://www.ntu.edu.sg/home/asjfcai/Benchmark_Website/benchmark_index.html)
Applications

**Computer vision:**

→ Image segmentation

[Image segmentation examples]

**Map labeling:**

→ Maximize nonoverlapping labels

† [http://www.ntu.edu.sg/home/asjfcai/Benchmark Website/benchmark_index.html](http://www.ntu.edu.sg/home/asjfcai/Benchmark Website/benchmark_index.html)
Applications

Computer vision:

→ Image segmentation

Map labeling:

→ Maximize nonoverlapping labels

Tracking submarines:

→ Coordinate information from multiple sensors

† http://www.ntu.edu.sg/home/asjfcai/Benchmark_Website/benchmark_index.html

* By Rama [CeCILL (http://www.cecill.info/licences/Licence_CeCILL_V2-en.html) or CC BY-SA 2.0 fr (https://creativecommons.org/licenses/by-sa/2.0/fr/deed.en)], from Wikimedia Commons
Large real-world networks

Graphs with millions/billions of nodes and “structure”

→ social networks, web-crawl graphs, co-citation networks

→ sparse, many low-degree vertices
Large real-world networks

Graphs with millions/billions of nodes and “structure”
→ social networks, web-crawl graphs, co-citation networks
→ sparse, many low-degree vertices

Branch-and bound is limited to hundreds (maybe thousands) of vertices
→ graph C4000.5 solved with 1 year of computation!
Large real-world networks

Graphs with millions/billions of nodes and “structure”
→ social networks, web-crawl graphs, co-citation networks
→ sparse, many low-degree vertices

Branch-and bound is limited to hundreds (maybe thousands) of vertices
→ graph C4000.5 solved with 1 year of computation!

Sparse graphs should be worse...
Large real-world networks

$\alpha(G)$ is (roughly) linear for sparse graphs...

→ linear search depth is infeasible for branch and bound
Large real-world networks

$\alpha(G)$ is (roughly) linear for sparse graphs...

→ linear search depth is infeasible for branch and bound

...enter inexact algorithms!

<table>
<thead>
<tr>
<th>Graph</th>
<th>$n$</th>
<th>Avg.</th>
<th>Max.</th>
<th>Min.</th>
</tr>
</thead>
<tbody>
<tr>
<td>enron</td>
<td>69 244</td>
<td>62 811</td>
<td>62 811</td>
<td>62 811</td>
</tr>
<tr>
<td>gowalla</td>
<td>196 591</td>
<td>112 369</td>
<td>112 369</td>
<td>112 369</td>
</tr>
<tr>
<td>citation</td>
<td>268 495</td>
<td>150 380</td>
<td>150 380</td>
<td>150 380</td>
</tr>
<tr>
<td>cnr-2000*</td>
<td>325 557</td>
<td>229 981</td>
<td>229 991</td>
<td>229 976</td>
</tr>
<tr>
<td>google</td>
<td>356 648</td>
<td>174 072</td>
<td>174 072</td>
<td>174 072</td>
</tr>
<tr>
<td>coPapers</td>
<td>434 102</td>
<td>47 996</td>
<td>47 996</td>
<td>47 996</td>
</tr>
<tr>
<td>skitter*</td>
<td>554 930</td>
<td>328 519</td>
<td>328 520</td>
<td>328 519</td>
</tr>
<tr>
<td>amazon</td>
<td>735 323</td>
<td>309 774</td>
<td>309 778</td>
<td>309 769</td>
</tr>
<tr>
<td>in-2004*</td>
<td>1 382 908</td>
<td>896 581</td>
<td>896 585</td>
<td>896 580</td>
</tr>
</tbody>
</table>
Branch and Reduce

[Akiba and Iwata, 2016]
Branch-and-reduce algorithms

Branch: Select vertex, remove neighbors

Reduce

Undo reductions, backtrack
Branch-and-reduce algorithms

Branch: Select vertex, remove neighbors

→ Effective in theory: $O^*(1.1996^n)$ [Xiao and Nagamochi, 2017]
Reduction rules

Degree 0
Reduction rules

Degree 0

Degree 1
Reduction rules

Degree 0

Degree 1

\[ u \]
Reduction rules

Degree 0

Degree 1

Degree 2
Reduction rules

Degree 0

Degree 1

Degree 2

vertex folding

Contract into single vertex

Shonan Meeting 144: March 5, 2019
Reductions

- LP-relaxation
  \[ \text{Maximize } \sum x_v \text{ where } x_u + x_v \leq 1. \text{ If } x_v = 1, \text{ then in some MIS.} \]

- Unconfined
  \[ \text{Some MIS exists without “unconfined” vertices} \]

- Twin
  \[ \text{Generalization of vertex folding} \]

- Diamond, alternative, . . .
And more…

Reductions

• LP-relaxation

  → Maximize $\sum x_v$ where $x_u + x_v \leq 1$. If $x_v = 1$, then in some MIS.

• Unconfined

  → Some MIS exists without “unconfined” vertices

• Twin

  → Generalization of vertex folding

• Diamond, alternative, …

Other techniques

• Packing constraints

  → Maintain constraints that update throughout recursion

• Branching rules, vertex ordering, …
Works well!

On LAW, SNAP, KONECT graphs...

- Solves in less than 1 second:
  - Citation networks ($\approx 200,000$ vertices)
  - Web crawl graphs ($\approx 500,000$ vertices)
  - Social networks ($\approx 3,000,000$ vertices)
Works well!

On LAW, SNAP, KONECT graphs...

- Solves in less than 1 second:
  - Citation networks ($\approx 200,000$ vertices)
  - Web crawl graphs ($\approx 500,000$ vertices)
  - Social networks ($\approx 3,000,000$ vertices)

- Some take much longer:
  - as-skitter (big) (1,170,580 vertices) 48 min
  - web-Stanford (163,390 vertices) 13 hours
  - Many networks remain unsolved
Works well!

On LAW, SNAP, KONECT graphs...

- Solves in less than 1 second:
  - Citation networks ($\approx 200,000$ vertices)
  - Web crawl graphs ($\approx 500,000$ vertices)
  - Social networks ($\approx 3,000,000$ vertices)

- Some take much longer:
  - as-skitter (big) (1,170,580 vertices) 48 min
  - web-Stanford (163,390 vertices) 13 hours
  - Many networks remain unsolved

$\rightarrow$ why?
Works well!

On LAW, SNAP, KONECT graphs...

- Solves in less than 1 second:
  - Citation networks ($\approx 200,000$ vertices)
  - Web crawl graphs ($\approx 500,000$ vertices)
  - Social networks ($\approx 3,000,000$ vertices)

- Some take much longer:
  - as-skitter (big) (1,170,580 vertices) 48 min
  - web-Stanford (163,390 vertices) 13 hours
  - Many networks remain unsolved

→ why?

→ what can be done about it?
Works well!

On LAW, SNAP, KONECT graphs...

- Solves in less than 1 second:
  - Citation networks ($\approx 200,000$ vertices)
  - Web crawl graphs ($\approx 500,000$ vertices)
  - Social networks ($\approx 3,000,000$ vertices)

- Some take much longer:
  - as-skitter (big) (1,170,580 vertices) 48 min
  - web-Stanford (163,390 vertices) 13 hours
  - Many networks remain unsolved

→ why?

reductions are powerful

→ why?

what can be done about it?

Remove redundant computation
Combine reductions and heuristic search
The power of simple reductions

[Strash, 2016]
An explanation

- Solves in less than 1 second:
  - Citation networks (≈200,000 vertices)
  - Web crawl graphs (≈500,000 vertices)
  - Social networks (≈3,000,000 vertices)

→ why?
An explanation

- Solves in less than 1 second:
  - Citation networks (≈200,000 vertices)
  - Web crawl graphs (≈500,000 vertices)
  - Social networks (≈3,000,000 vertices)

→ After reductions, nearly all graphs are empty.

→ why?
An explanation

- Solves in less than 1 second:
  - Citation networks ($\approx 200,000$ vertices)
  - Web crawl graphs ($\approx 500,000$ vertices)
  - Social networks ($\approx 3,000,000$ vertices)

→ After reductions, nearly all graphs are empty.

- 80% instances solved with two reductions ($< 1$ sec)
  - Vertex folding
  - Isolated clique removal

→ why?

11  Shonan Meeting 144: March 5, 2019  Darren Strash
Heuristic: Guess “likely candidates” [Lamm et al. 2016]

- Branch-and-reduce selects **one solution vertex** at a time.
  → Limits the number of reductions in next recursion call.

Exponential time!
Heuristic: Guess “likely candidates” [Lamm et al. 2016]

- Branch-and-reduce selects **one solution vertex** at a time → Limits the number of reductions in next recursion call

- Can we guess **many vertices** that are likely in an MIS? → Remove and continue applying reductions

Exponential time!

Fast, but inexact
Evolutionary algorithm [EvoMIS] [Lamm et al. 2015]

- Start with an initial independent set $I$
- Swap whole blocks of independent set nodes using separators and graph partitioning

Update solutions to $A$ and $B$ with local search
→ next generation
Evolutionary algorithm [EvoMIS] [Lamm et al. 2015]

- Start with an initial independent set $I$
- Swap whole blocks of independent set nodes using *separators* and *graph partitioning*

Update solutions to $A$ and $B$ with *local search*

$\rightarrow$ next generation

- Finds large independent sets in large sparse networks.
Evolutionary algorithm [EvoMIS] [Lamm et al. 2015]

- Start with an initial independent set $I$
- Swap whole blocks of independent set nodes using separators and graph partitioning

Update solutions to $A$ and $B$ with local search
→ next generation

- Finds large independent sets in large sparse networks.

Idea: Select low-degree vertices from “fittest” independent set.
ReduMIS: Near-optimal on “difficult networks”

- Finds exact MIS faster, when exact algorithm is slow:
  - as-skitter (big) 48 min → 21 min
  - web-Stanford 13 hours → 5 min
  - bcsstk30 8.6 hours → 2.4 sec
  - brack2 13 min → 9.4 sec
  - col 2 hours → 28 sec
ReduMIS: Near-optimal on “difficult networks”

- Finds exact MIS faster, when exact algorithm is slow:
  - as-skitter (big) \(48 \text{ min} \rightarrow 21 \text{ min}\)
  - web-Stanford \(13 \text{ hours} \rightarrow 5 \text{ min}\)
  - bcsstk30 \(8.6 \text{ hours} \rightarrow 2.4 \text{ sec}\)
  - brack2 \(13 \text{ min} \rightarrow 9.4 \text{ sec}\)
  - col \(2 \text{ hours} \rightarrow 28 \text{ sec}\)

- Finds exact MIS, for large networks with known MIS size
ReduMIS: Near-optimal on “difficult networks”

- Finds exact MIS faster, when exact algorithm is slow:
  - as-skitter (big) 48 min → 21 min
  - web-Stanford 13 hours → 5 min
  - bcsstk30 8.6 hours → 2.4 sec
  - brack2 13 min → 9.4 sec
  - col 2 hours → 28 sec

- Finds exact MIS, for large networks with known MIS size
- Consistently finds same value for large graphs with unknown MIS size
  - cnr-2000 → 230,036
  - skitter → 328,626
  - amazon → 309,794
  - ny → 131,502
ReduMIS: Finds larger solutions faster

- Consistently finds larger solutions on social, Web, and road networks

→ averaged over 5 runs.
ReduMIS: Finds larger solutions faster

- Consistently finds larger solutions on social, Web, and road networks
  → averaged over 5 runs.
ReduMIS: Finds larger solutions faster

- Consistently finds larger solutions on social, Web, and road networks
  → averaged over 5 runs.

- Consistent, even as we scale to graphs on **10M to 100M nodes**
- However, finds worse solutions for huge meshes
• Start with some maximal independent set
• (1,2)-swaps → remove one vertex, add two
• avoid swapping “recently” swapped vertices
• When not possible, perturb the solution
Iterated Local Search [ARW] [Andrade et al. 2012]

- Start with some maximal independent set
- (1,2)-swaps → remove one vertex, add two
- avoid swapping “recently” swapped vertices
- When not possible, perturb the solution

![Diagram](image-url)
Accelerating Local Search  

- **Problem:** Too much time to wait for high-quality solution.

[Dahlum et al. 2016]
Accelerating Local Search [Dahlum et al. 2016]

- **Problem:** Too much time to wait for high-quality solution.
- **Solution:** Speed up local search with **online** reductions, and removing **high-degree** vertices.

![Graphs showing exact and inexact approaches](image)

![Graph showing solution size vs. time](image)
Accelerating Local Search  

**Problem:** Too much time to wait for high-quality solution.

**Solution:** Speed up local search with *online* reductions, and removing *high-degree* vertices.

Exact, reduce graph size

Inexact, for speed up

![Graph comparison](image1.jpg)

![Graph comparison](image2.jpg)
Accelerating Local Search  

[Dahlum et al. 2016]

- **Problem:** Too much time to wait for high-quality solution.
- **Solution:** Speed up local search with **online** reductions, and removing **high-degree** vertices.

---

**Exact, reduce graph size**

**Inexact, for speed up**

300x faster!
**Linear-time reductions**  [Chang et al. 2017]

**Problem:** Vertex folding is slow with high-degree neighbors
Linear-time reductions [Chang et al. 2017]

**Problem:** Vertex folding is slow with high-degree neighbors

**Solution:** Avoid it...

```
            or
            or...
```

```
  . . .
  or
  or...
```
Linear-time reductions [Chang et al. 2017]

**Problem:** Vertex folding is slow with high-degree neighbors

**Solution:** Avoid it...

![Diagram](image)

**Repeat:** Add small degree vertex to solution + reduce

or...

or...
Linear-time reductions [Chang et al. 2017]

(a) soc-pokec

(b) indochina

(c) webbase

(d) it-2004
### Scalable Reductions

**Problem:** Effective reductions are slow

| Graph name       | Graph size $n$ | Linear $|K|$ | Time | NearLinear $|K|$ | Time | VCSolver $|K|$ | Time |
|------------------|----------------|---------|------|-------------|------|-------------|------|
| uk-2002          | 19M            | 11.7M   | 1.5  | 4.0M        | 28.0 | 0.2M        | 336.9|
| arabic-2005      | 23M            | 15.6M   | 2.6  | 6.7M        | 246.1| 0.6M        | 1033.2|
| gsh-2015-tpd     | 31M            | 2.0M    | 11.6 | 1.2M        | 97.4 | 0.4M        | 372.3|
| uk-2005          | 39M            | 28.2M   | 2.5  | 5.9M        | 60.5 | 0.8M        | 541.4|
| it-2004          | 41M            | 27.1M   | 3.3  | 11.3M       | 1544.6| 1.6M        | 6749.0|
| sk-2005          | 51M            | *       | *    | *           | *    | 3.2M        | 10010.5|
| uk-2007-05       | 106M           | *       | *    | *           | *    | 3.5M        | 18829.4|
| webbase-2001     | 118M           | 51.7M   | 13.0 | 17.3M       | 121.1| 0.7M        | 4207.8|
| asia.osm         | 12M            | 626.7K  | 0.8  | 594.4K      | 1.4  | 15.2K       | 204.7|
| road_usa         | 24M            | 2.5M    | 2.5  | 2.4M        | 4.1  | 0.2M        | 310.0|
| europe.osm       | 51M            | 1500.0K | 4.1  | 1329.9K     | 6.1  | 8.4K        | 302.4|
| rgg26            | 67M            | 67.1M   | 1.0  | 51.3M       | 172.6| 49.6M       | 9887.7|
| rhg              | 100M           | *       | *    | *           | *    | 0           | 124.0|
| del24            | 17M            | 16.8M   | 0.2  | 15.6M       | 12.7 | 12.4M       | 4789.5|
| del26            | 67M            | 67.1M   | 0.7  | 62.5M       | 53.3 | 49.9M       | 20728.7|
Scalable Reductions

[Hespe et al. 2018]

**Problem:** Effective reductions are slow

<table>
<thead>
<tr>
<th>Graph name</th>
<th>$n$</th>
<th></th>
<th>$K$</th>
<th>time</th>
<th></th>
<th>$K$</th>
<th>time</th>
<th>$K$</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td>uk-2002</td>
<td>19M</td>
<td>11.7M</td>
<td>1.5</td>
<td>4.0M</td>
<td>28.0</td>
<td>0.2M</td>
<td>336.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>arabic-2005</td>
<td>23M</td>
<td>15.6M</td>
<td>2.6</td>
<td>6.7M</td>
<td>246.1</td>
<td>0.6M</td>
<td>1033.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gsh-2015-tpd</td>
<td>31M</td>
<td>2.0M</td>
<td>11.6</td>
<td>1.2M</td>
<td>97.4</td>
<td>0.4M</td>
<td>372.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uk-2005</td>
<td>39M</td>
<td>28.2M</td>
<td>2.5</td>
<td>5.9M</td>
<td>60.5</td>
<td>0.8M</td>
<td>541.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>it-2004</td>
<td>41M</td>
<td>27.1M</td>
<td>3.3</td>
<td>11.3M</td>
<td>1544.6</td>
<td>1.6M</td>
<td>6749.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sk-2005</td>
<td>51M</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>3.2M</td>
<td>10010.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uk-2007-05</td>
<td>106M</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>3.5M</td>
<td>18829.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>webbase-2001</td>
<td>118M</td>
<td>51.7M</td>
<td>13.0</td>
<td>17.3M</td>
<td>121.1</td>
<td>0.7M</td>
<td>4207.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>asia.osm</td>
<td>12M</td>
<td>626.7K</td>
<td>0.8</td>
<td>594.4K</td>
<td>1.4</td>
<td>15.2K</td>
<td>204.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>road_usa</td>
<td>24M</td>
<td>2.5M</td>
<td>2.5</td>
<td>2.4M</td>
<td>4.1</td>
<td>0.2M</td>
<td>310.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>europe.osm</td>
<td>51M</td>
<td>1500.0K</td>
<td>4.1</td>
<td>1329.9K</td>
<td>6.1</td>
<td>8.4K</td>
<td>302.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rgg26</td>
<td>67M</td>
<td>67.1M</td>
<td>1.0</td>
<td>51.3M</td>
<td>172.6</td>
<td>49.6M</td>
<td>9887.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rhg</td>
<td>100M</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>0</td>
<td>124.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>del24</td>
<td>17M</td>
<td>16.8M</td>
<td>0.2</td>
<td>15.6M</td>
<td>12.7</td>
<td>12.4M</td>
<td>4789.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>del26</td>
<td>67M</td>
<td>67.1M</td>
<td>0.7</td>
<td>62.5M</td>
<td>53.3</td>
<td>49.9M</td>
<td>20728.7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Scalable Reductions

[Hespe et al. 2018]

Solutions:

Only check parts of graph that change
Scalable Reductions [Hespe et al. 2018]

Solutions:

Only check parts of graph that change

Parallelize

Cannot do both reductions at the same time
Scalable Reductions

[Hespe et al. 2018]

**Solutions:**

Only check parts of graph that change

Parallelize

Cannot do both reductions at the same time

Stop reductions if they are ineffective
## Scalable Reductions

[Hespe et al. 2018]

<table>
<thead>
<tr>
<th>NearLinear</th>
<th>VCSolver</th>
<th>ParFastKer</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>K</td>
<td>$, time</td>
</tr>
<tr>
<td>4.0M, 28.0</td>
<td>0.2M, 336.9</td>
<td>0.3M, 11.8, 28.4</td>
</tr>
<tr>
<td>6.7M, 246.1</td>
<td>0.6M, 1033.2</td>
<td>0.6M, 25.7, 40.2</td>
</tr>
<tr>
<td>1.2M, 97.4</td>
<td>0.4M, 372.3</td>
<td>0.5M, 32.0, 11.7</td>
</tr>
<tr>
<td>5.9M, 60.5</td>
<td>0.8M, 541.4</td>
<td><strong>0.9M</strong>, 53.3, 10.1</td>
</tr>
<tr>
<td>11.3M, 1544.6</td>
<td>1.6M, 6749.0</td>
<td>1.7M, 151.8, 44.4</td>
</tr>
<tr>
<td>*</td>
<td>3.2M, 10010.5</td>
<td>3.5M, 178.3, 56.1</td>
</tr>
<tr>
<td>*</td>
<td>3.5M, 18829.4</td>
<td>3.7M, 372.4, 50.6</td>
</tr>
<tr>
<td><strong>17.3M</strong>, 121.1</td>
<td><strong>0.7M</strong>, 4207.8</td>
<td><strong>0.9M</strong>, 54.9, 76.6</td>
</tr>
<tr>
<td>594.4K, 1.4</td>
<td>15.2K, 204.7</td>
<td><strong>34.9K</strong>, 1.2, 169.8</td>
</tr>
<tr>
<td>2.4M, 4.1</td>
<td>0.2M, 310.0</td>
<td><strong>0.2M</strong>, 4.1, 76.0</td>
</tr>
<tr>
<td>1329.9K, 6.1</td>
<td>8.4K, 302.4</td>
<td><strong>14.2K</strong>, 4.9, 61.3</td>
</tr>
<tr>
<td>51.3M, 172.6</td>
<td>49.6M, 9887.7</td>
<td><strong>19.8M</strong>, 150.3, 65.8</td>
</tr>
<tr>
<td>*</td>
<td>0, 124.0</td>
<td>16, 64.6, 1.9</td>
</tr>
<tr>
<td>15.6M, 12.7</td>
<td>12.4M, 4789.5</td>
<td>12.9M, 51.5, 93.1</td>
</tr>
<tr>
<td>62.5M, 53.3</td>
<td>49.9M, 20728.7</td>
<td>51.7M, 179.0, 115.8</td>
</tr>
</tbody>
</table>
The weighted case
Weighted variant

> 2 months ago:

Cannot solve on graphs with 500 vertices

One LP reduction — untested
New reductions [Lamm et al. 2019]

heaviest

\[ u \]
New reductions [Lamm et al. 2019]

heaviest

Contract into single vertex

sum heavier, but each lighter

[u]

heaviest

[heaviest]

[heaviest]

[Lamm et al. 2019]
New reductions [Lamm et al. 2019]

- Contract into single vertex
- Heaviest
- Sum heavier, but each lighter

Lamm et al. 2019

Shonan Meeting 144: March 5, 2019

Darren Strash
New reductions \cite{Lamm2019}

- Contract into single vertex

- Sum heavier, but each lighter

- Sum heavier, but sum of 2 lighter
New reductions [Lamm et al. 2019]

- Contract into single vertex
- Heaviest
- Sum heavier, but each lighter
- Sum heavier, but sum of 2 lighter
- ???
Meta-reductions

Theorem.

\[ \mathcal{U} \]

\[ \mathcal{I} \]
Meta-reductions

Theorem. \[ \omega(u) \geq \omega(\mathcal{I})? \]

Choose \[ u \]
Meta-reductions

**Theorem.**

Choose \( u \)

\[ w(u) \geq w(\mathcal{I}) \]

**Theorem.**

Choose \( u \)

\[ w(u) < w(\mathcal{I}) \]
Meta-reductions

Theorem. Choose $u$

\[ w(u) \geq w(I) \]

Choose $u$

Theorem.

\[ w(u) < w(I) \]

$I$ uniquely larger? contract $u'$
<table>
<thead>
<tr>
<th>Graph</th>
<th>B &amp; R&lt;sub&gt;full&lt;/sub&gt;</th>
</tr>
</thead>
</table>
|                  | <i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i><i>
### Practice / Application: Viability for map labeling

<table>
<thead>
<tr>
<th>Graph</th>
<th>$B$ &amp; $R_{\text{full}}$</th>
<th>$t_{\text{max}}$</th>
<th>$w_{\text{max}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>alabama-AM2</td>
<td>0.79 174 309</td>
<td></td>
<td></td>
</tr>
<tr>
<td>alabama-AM3</td>
<td>80.78 185 707</td>
<td></td>
<td></td>
</tr>
<tr>
<td>district-of-columbia-AM1</td>
<td>4.13 196 475</td>
<td></td>
<td></td>
</tr>
<tr>
<td>district-of-columbia-AM2</td>
<td>233.70 147 450</td>
<td></td>
<td></td>
</tr>
<tr>
<td>district-of-columbia-AM3</td>
<td>918.07 92 714</td>
<td></td>
<td></td>
</tr>
<tr>
<td>florida-AM2</td>
<td>0.02 230 595</td>
<td></td>
<td></td>
</tr>
<tr>
<td>florida-AM3</td>
<td>324.38 226 767</td>
<td></td>
<td></td>
</tr>
<tr>
<td>georgia-AM3</td>
<td>14.35 214 918</td>
<td></td>
<td></td>
</tr>
<tr>
<td>greenland-AM3</td>
<td>47.25 13 069</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hawaii-AM2</td>
<td>10.89 125 284</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hawaii-AM3</td>
<td>1 177.95 129 812</td>
<td></td>
<td></td>
</tr>
<tr>
<td>idaho-AM3</td>
<td>61.26 76 831</td>
<td></td>
<td></td>
</tr>
<tr>
<td>kansas-AM3</td>
<td>18.99 87 925</td>
<td></td>
<td></td>
</tr>
<tr>
<td>kentucky-AM2</td>
<td>42.05 97 397</td>
<td></td>
<td></td>
</tr>
<tr>
<td>kentucky-AM3</td>
<td>3346.94 96 634</td>
<td></td>
<td></td>
</tr>
<tr>
<td>louisiana-AM3</td>
<td>20.17 60 024</td>
<td></td>
<td></td>
</tr>
<tr>
<td>maryland-AM3</td>
<td>11.08 45 496</td>
<td></td>
<td></td>
</tr>
<tr>
<td>massachusetts-AM2</td>
<td>0.48 140 095</td>
<td></td>
<td></td>
</tr>
<tr>
<td>massachusetts-AM3</td>
<td>23.97 145 631</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mexico-AM3</td>
<td>289.14 97 663</td>
<td></td>
<td></td>
</tr>
<tr>
<td>new-hampshire-AM3</td>
<td>8.75 116 060</td>
<td></td>
<td></td>
</tr>
<tr>
<td>north-carolina-AM3</td>
<td>11.55 49 562</td>
<td></td>
<td></td>
</tr>
<tr>
<td>oregon-AM2</td>
<td>0.09 165 047</td>
<td></td>
<td></td>
</tr>
<tr>
<td>oregon-AM3</td>
<td>474.15 164 941</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pennsylvania-AM3</td>
<td>38.76 143 870</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rhode-island-AM2</td>
<td>16.79 184 543</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rhode-island-AM3</td>
<td>931.05 163 808</td>
<td></td>
<td></td>
</tr>
<tr>
<td>utah-AM3</td>
<td>285.22 98 847</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vermont-AM3</td>
<td>443.88 55 577</td>
<td></td>
<td></td>
</tr>
<tr>
<td>virginia-AM2</td>
<td>0.77 295 867</td>
<td></td>
<td></td>
</tr>
<tr>
<td>virginia-AM3</td>
<td>786.05 233 572</td>
<td></td>
<td></td>
</tr>
<tr>
<td>washington-AM2</td>
<td>2.20 305 619</td>
<td></td>
<td></td>
</tr>
<tr>
<td>washington-AM3</td>
<td>532.25 271 747</td>
<td></td>
<td></td>
</tr>
<tr>
<td>west-virginia-AM3</td>
<td>854.73 47 927</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Graph</th>
<th>$B$ &amp; $R_{\text{full}}$</th>
<th>$t_{\text{max}}$</th>
<th>$w_{\text{max}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>as-skitter</td>
<td>746.93 123 904 741</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ca-AstroPh</td>
<td>0.03 796 556</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ca-CondMat</td>
<td>0.02 1143 480</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ca-GrQC</td>
<td>0.00 289 481</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ca-HepPh</td>
<td>0.02 579 675</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ca-HepTh</td>
<td>0.01 560 662</td>
<td></td>
<td></td>
</tr>
<tr>
<td>email-Enron</td>
<td>0.03 2457 547</td>
<td></td>
<td></td>
</tr>
<tr>
<td>email-EuAll</td>
<td>0.19 25330 331</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2p-Gnutella04</td>
<td>0.01 667 539</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2p-Gnutella05</td>
<td>0.01 556 559</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2p-Gnutella06</td>
<td>0.01 547 591</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2p-Gnutella08</td>
<td>0.01 435 893</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2p-Gnutella09</td>
<td>0.01 568 472</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2p-Gnutella24</td>
<td>0.02 1970 329</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2p-Gnutella25</td>
<td>0.02 1697 310</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2p-Gnutella30</td>
<td>0.03 2785 957</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2p-Gnutella31</td>
<td>0.04 4750 671</td>
<td></td>
<td></td>
</tr>
<tr>
<td>roadNet-CA</td>
<td>774.56 111 408 830</td>
<td></td>
<td></td>
</tr>
<tr>
<td>roadNet-PA</td>
<td>32.06 61 686 106</td>
<td></td>
<td></td>
</tr>
<tr>
<td>roadNet-TX</td>
<td>33.49 78 606 965</td>
<td></td>
<td></td>
</tr>
<tr>
<td>soc-Epinions1</td>
<td>0.11 5668 401</td>
<td></td>
<td></td>
</tr>
<tr>
<td>soc-LiveJournal1</td>
<td>270.96 283 948 671</td>
<td></td>
<td></td>
</tr>
<tr>
<td>soc-Slashdot0811</td>
<td>0.18 5650 791</td>
<td></td>
<td></td>
</tr>
<tr>
<td>soc-Slashdot0902</td>
<td>0.21 5953 582</td>
<td></td>
<td></td>
</tr>
<tr>
<td>soc-pokec-relationships</td>
<td>1404.57 75 717 984</td>
<td></td>
<td></td>
</tr>
<tr>
<td>web-BerkStan</td>
<td>831.75 43 766 431</td>
<td></td>
<td></td>
</tr>
<tr>
<td>web-Google</td>
<td>3.16 56313 384</td>
<td></td>
<td></td>
</tr>
<tr>
<td>web-NotreDame</td>
<td>28.11 25 957 800</td>
<td></td>
<td></td>
</tr>
<tr>
<td>web-Stanford</td>
<td>4.69 17 799 469</td>
<td></td>
<td></td>
</tr>
<tr>
<td>wiki-Talk</td>
<td>3.36 235 875 181</td>
<td></td>
<td></td>
</tr>
<tr>
<td>wiki-Vote</td>
<td>0.06 500 436</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

and more graphs
Conclusion

Reduction efficiency is important in practice

Reductions are effective in practice

Reductions + heuristics are a winning combination

Next? → transfer to theory
Conclusion

Reduction efficiency is important in practice

Reductions are effective in practice

Reductions + heuristics are a winning combination

Next? → transfer to theory

Thank you!