o‘zapft is: Tap Your Networking Algorithm‘s Big Data!

Andreas Blenk
Patrick Kalmbach
  *Patrick.Kalmbach@tum.de*
Wolfgang Kellerer

Stefan Schmid
  *schmiste@cs.aau.dk*
Network Algorithms are Ubiquitious

Routing

Cache/Content Placement

Congestion Control

Two classes of optimization problems: Packing and Covering problems
The Limitation – Fire and Forget

Algorithms repeatedly solve similar problems from scratch. This is not only boring for the algorithm but also a waste of information and resources.

The Opportunity – Tap into your Algorithm’s Big Data
Traditional vs. Proposed System

Traditional System

Problem Instances → Machine Learning → Solution Information → Optimization Algorithm → Problem Solutions

Problem Instances → Optimization Algorithm → Problem Solutions

learn from (offline)

produce

Data Available at: Patrick Kalmbach, Johannes Zerwas, Michael Manhart, Andreas Blenk, Stefan Schmid, and Wolfgang Kellerer. 2017. Data on "o’zapft is: Tap Your Network Algorithm’s Big Data!". (2017). https://doi.org/10.14459/2017md1361589

Patrick Kalmbach (TUM) | o’zapft is: Tap Your Networking Algorithm's Big Data!
Potentials

Search Space Reduction reduction/Initial Solutions

Predict Value of Objective Function
Facility Location (Controller Placement) – Guess Initial solutions

**Problem**: Given a network and a number of controllers, where to place the controllers?
Data Generation – Facility Location

Optimization Algorithms:
- Random Placement (Rnd)
- Mixed Integer Program (MIP)
- Greedy (Gdy)

Substrates
- Barabasi-Albert (BA) [2] 40 nodes
- Whole Topology Zoo (TZ) [1] 20-800 nodes

Objective:
Minimize maximum latency between node and controller

Learning to Place

Library:
- Sci-Kit Learn [1]

Features:
- Node degree
- Closeness
- Betweenness
- Spectral Features

Measures:
- F1 Score

Classifier:
- Logistic Regression
- Support Vector Machine
- Random Forest
- Extra Tree
- AdaBoost

Model Training and Selection:
- 80% Training
- 20% Testing
- Feature Extraction
- Min Max
- Balance Classes
- 2-Fold Stratified Cross Validation
- Best Parameters
- Final Results

Patrick Kalmbach (TUM) | o‘zapft is: Tap Your Networking Algorithm’s Big Data!
Use Case: Facility Location

**Problem**: Place Facility on network node such that maximum latency is minimized

Node $n_i$ → Calculate Node Features → Feature Vector $x_i$ → Machine Learning → $P(n_i = 1 | x_i)$

\[ \text{argmax}_{n_i} \{ P(n_i = 1 | x_i) \} \]

\[ o'zapft \text{ is-Exact} \]

\[ o'zapft \text{ is-MIP0.25} \]

\[ o'zapft \text{ is-MIP0.5} \]
Use Case: Facility Location

Large reduction of solution space with only small performance degradation
# Search Space Reduction

<table>
<thead>
<tr>
<th>Substrate</th>
<th>Kentucky Data Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Nodes</td>
<td>734</td>
</tr>
<tr>
<td>Number Facilities</td>
<td>10</td>
</tr>
</tbody>
</table>

MIP

- $o'zapft is-$
  - MIP0.5: $9.2 \times 10^{-4}$
  - MIP0.25: $7.7 \times 10^{-7}$
Impact of Complex Features

**Good classification performance for features with low complexity**
Conclusion

It is possible to predict the behavior of networking algorithms from past problem-solution-pairs.

- **Search Space Reduction**
  - Reduction/Initial Solutions
  - Predict Value of Objective Function

![Graph showing search space reduction and initial solutions](Image)

![Graph showing prediction of objective function](Image)
Future Work

• Transfer Learning
• Investigate the size of the minimal search space
• Investigate whether heuristics improve on reduced search space
• Investigating the applicability of Deep Learning