TU WIEN
FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Diplomarbeitspräsentation

dbai
Database and Artificial Intelligence Group

Masterstudium:
Software Engineering & Internet Computing

Algorithm Selection for the Graph Coloring Problem

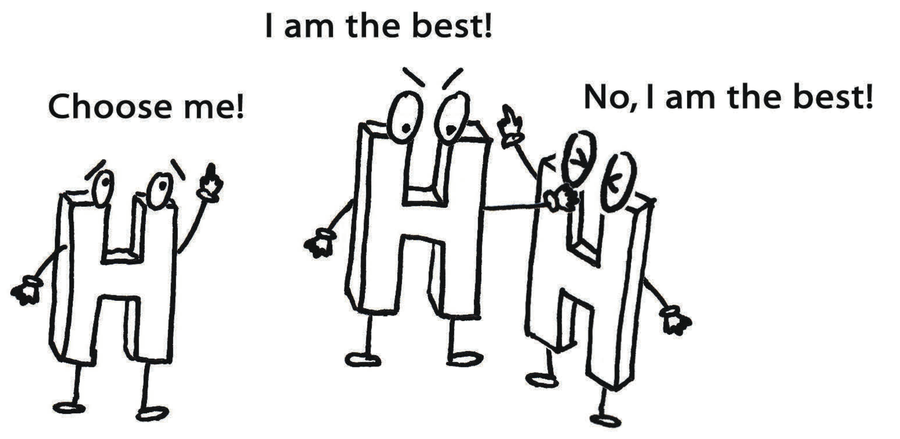
Martin Schwengerer

Technische Universität Wien
Institut für Informationssysteme
Database and Artificial Intelligence Group
Betreuer: Priv.-Doz. Dr. Nysret Musliu

Motivation

- Some problems (called NP-hard problems) cannot be solved efficiently
- ⇒ focus on heuristic algorithms
- But: None of them is perfect on all problems (known as "No Free Lunch" theorem [1])

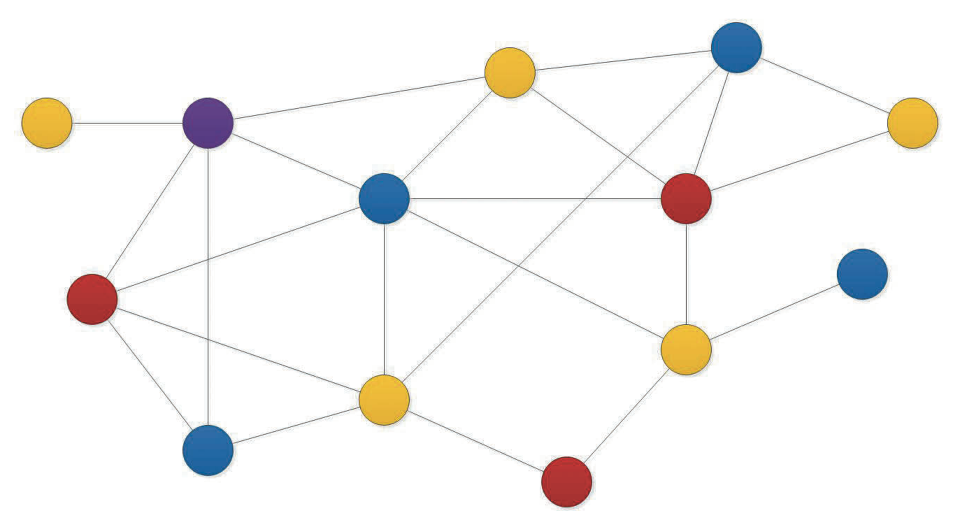
Problem: Which heuristic should be used?



One approach: Select always the algorithm from which we expect the best performance.

Example Problem: GCP

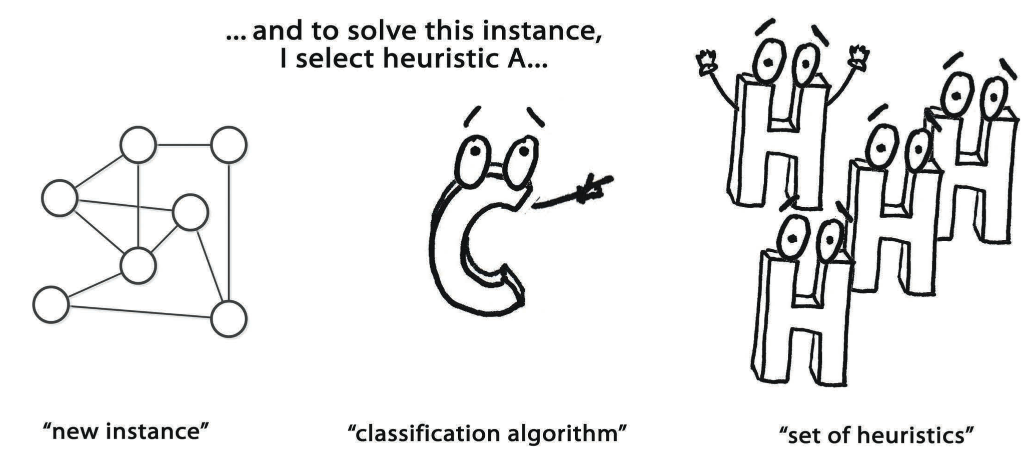
- The Graph Coloring Problem (GCP) is a well-known NP-hard problem
- Input: Graph $G = (V, E)$
- Objective: assign each node a color such that
  - no adjacent nodes have the same color and
  - the total number of colors is minimized.



- There exist different heuristic approaches for GCP like *tabu search*, *simulated annealing*, *genetic algorithms*, *ant colony optimization*, ...

Our Approach:

Use Machine Learning techniques for automated algorithm selection for the GCP!

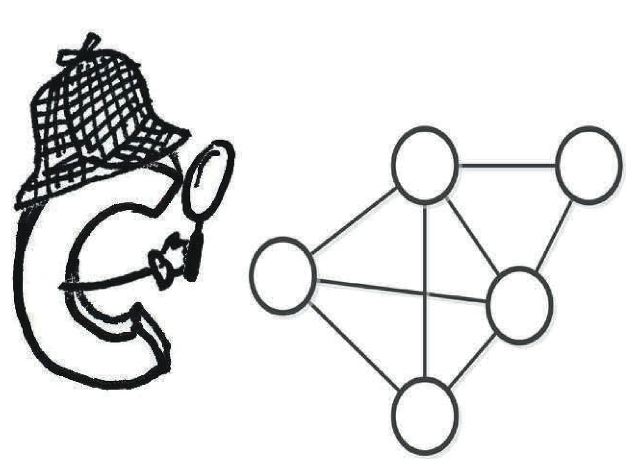


1. Identify characteristic features of a graph
2. Evaluate the performance of several state-of-the-art solvers for the GCP
3. Train classification algorithms to predict the best algorithm for a new GCP instance.

Step 1: Identify Instance Features

We identified **78** features of a GCP instance that can be calculated in polynomial time based on:

- Graph Size
- Node degree
- Clustering Coefficient
- Clique Size
- Greedy Coloring Algorithms
- Local Search Attributes
- Lower- and upper bounds
- Tree Decomposition



Step 2: Evaluation of Several State-Of-The-Art Heuristics

We tested **7** heuristic algorithms:

- Foo-PartialCol (FPC),
- Hybrid Evolutionary Algorithm (HEA),
- Iterated Local Search (ILS),
- Multi-Agent Fusion Search (MAFS),
- MACOL,
- MMT, and
- TABUCOL (TABU)

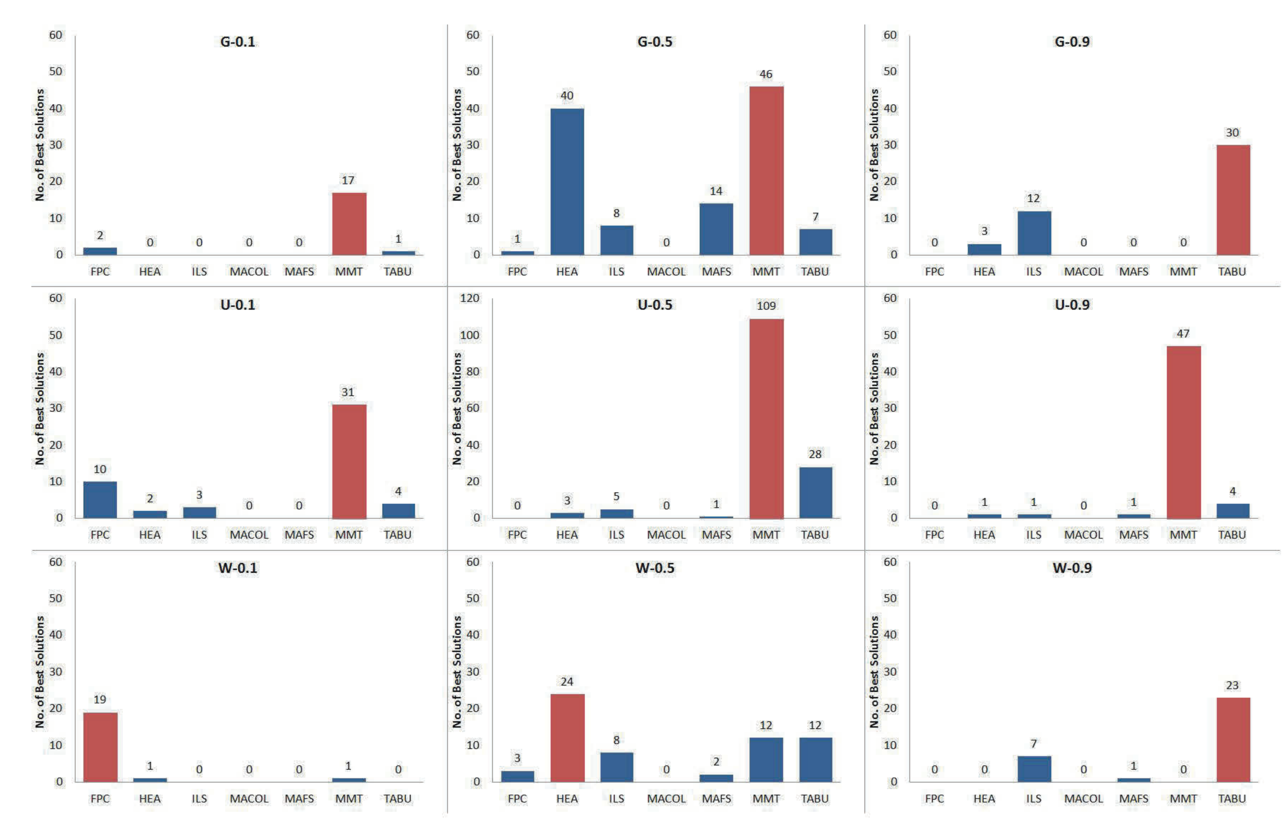
on **3** public available instance sets

- **1265** graphs

Total runtime: roughly 90.000 hours

Results:

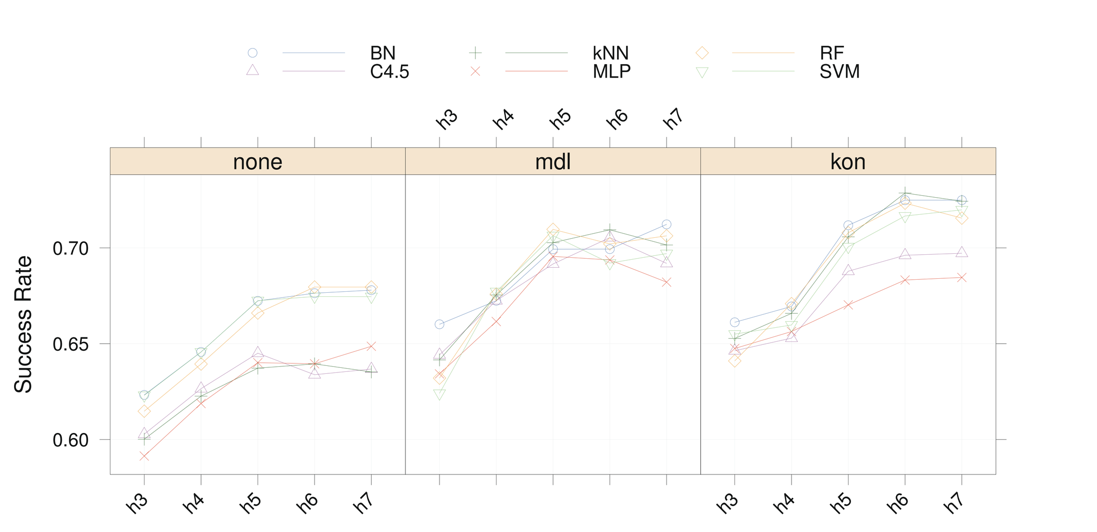
- no heuristics dominates all others on all graphs
- some heuristics show better performance on graphs with certain features



Number of best solutions obtained by the different heuristics. The red bar denotes that this algorithm achieved the highest number within the subset of instances.

Step 3: Training Phase

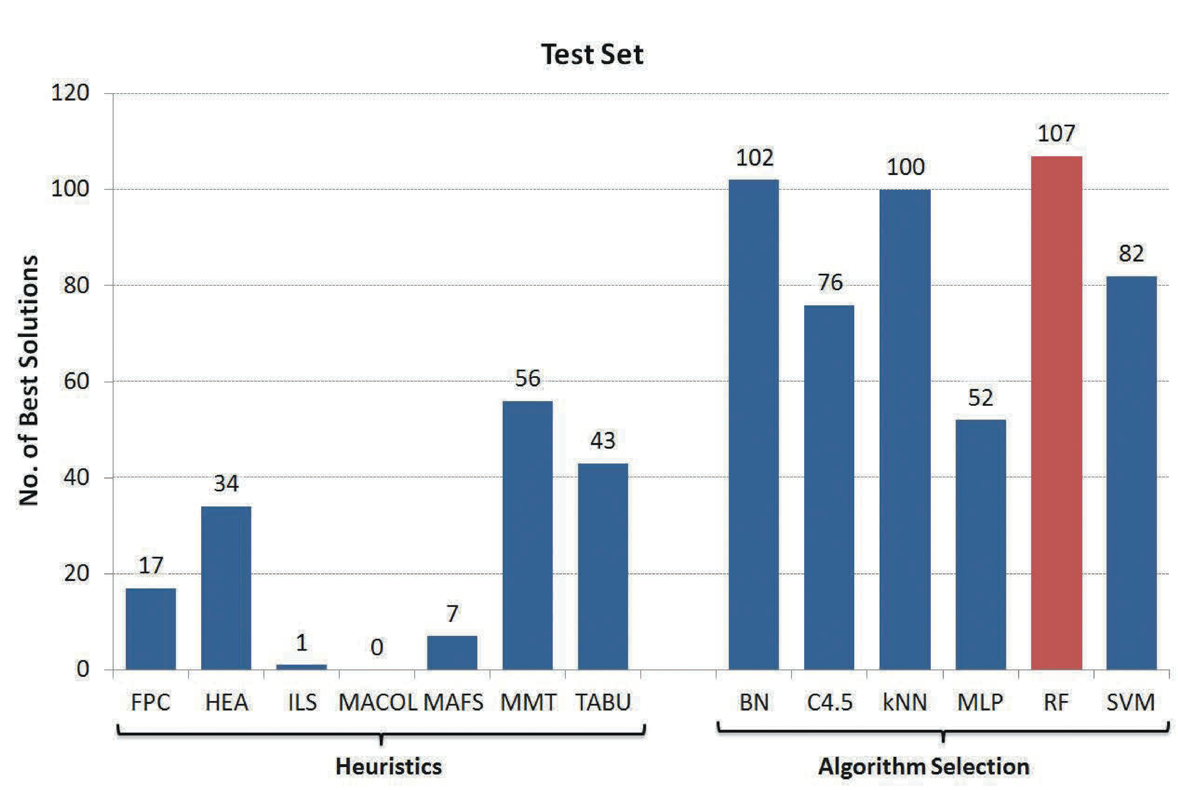
- Trained **6** well-known classification algorithms:
  - k-nearest Neighbor (kNN),
  - C4.5 Decision Trees (C4.5),
  - Bayes Networks (BN),
  - Multi-Layer Perceptrons (MLP),
  - Random Forest (RF), and
  - Support-Vector Machines (SVM),
- Different discretization methods (MDL, KON),
- Analyzed the impact of using a subset of heuristics on the overall quality of the prediction.



Performance of different classification algorithms on data without discretization (none), discretized with the classical MDL criteria (mdl) and with Kononenko's criteria (kon). The x-axis represents algorithms set among which the classifier can choose whereby h*x* represents the best *x* algorithm according to our evaluation.

Evaluation & Results

We compared our automated algorithm selection solvers with the existing solvers on **152** new generated instances:



Number of best solutions per solver. The red bar denotes the approach that shows on the highest number of instances the best performance.

Solver	No. Best Solution	$s(c,I,A)$ (%)	$err(k,i)$ (%)	Rank avg	Rank stdev	F1
Heuristics						
FPC	17	11.18	25.43	3.39	1.53	919
HEA	34	22.37	15.25	2.74	1.43	1065
ILS	1	0.66	21.97	3.99	1.56	784
MACOL	0	0.00	28.13	5.17	1.23	588
MAFS	7	4.61	31.71	5.34	1.94	585
MMT	56	36.84	**4.63**	2.88	1.99	1077
TABU	43	28.29	19.47	2.57	1.25	1094
Algorithm Selection						
BN	102	67.11	5.85	1.58	1.02	1360
C4.5	76	50.00	4.90	2.26	1.62	1204
IBK	100	65.79	**4.88**	1.61	1.17	1357
MLP	52	34.21	22.92	2.64	1.54	1091
RF	**107**	**70.39**	6.44	**1.50**	1.07	**1386**
SVM	82	53.95	9.37	2.10	1.58	1240
Best (heu)	56	36.84	**4.63**	2.57	1.25	1094
Best (AS)	**107**	**70.39**	4.88	**1.50**	1.07	**1386**

Performance metrics of the algorithm selection and the existing solver on the *test set*.

Results:

- Classification algorithms predicts for up to **70.39%** of the graphs the most suited algorithm
- Improvement of **+33.55%** compared with the best solver

References

- D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, vol. 1, no. 1, pp. 67–82, 1997.

Kontakt: mschweng@kr.tuwien.ac.at