Mixed Integer Linear Programming Based Large Neighborhood Search Approaches for the Directed Feedback Vertex Set Problem*

Maria Bresich [1] Johannes Varga [1] Günther R. Raidl [1] Steffen Limmer [2]

[1]Institute of Logic and Computation, TU Wien, Austria,
mbresich, jvarga, raidl@ac.tuwien.ac.at

[2]Honda Research Institute Europe, Germany,
steffen.limmer@honda-ri.de

META'2023, Marrakech, Morocco
November 3, 2023

*This project was financially supported by Honda Research Institute Europe GmbH.

Introduction

Definition (Directed Feedback Vertex Set Problem)

Given: Directed graph $G = (V, E)$
Task: Find $F \subseteq V$ of minimum cardinality, s.t. $G[V \setminus F]$ is acyclic.

F ... directed feedback vertex set (DFVS)
$G[V \setminus F]$ ... directed acyclic graph (DAG)

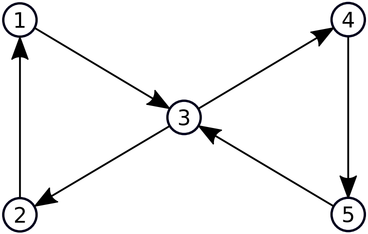


Figure: Example of a DFVS problem instance.

Introduction

Definition (Directed Feedback Vertex Set Problem)

Given: Directed graph $G = (V, E)$
Task: Find $F \subseteq V$ of minimum cardinality, s.t. $G[V \setminus F]$ is acyclic.

F ... directed feedback vertex set (DFVS)
$G[V \setminus F]$ ... directed acyclic graph (DAG)

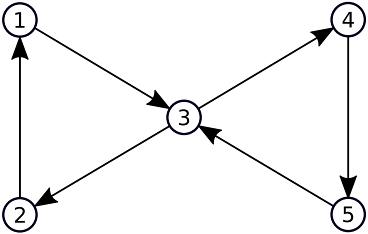


Figure: Example of a DFVS problem instance.

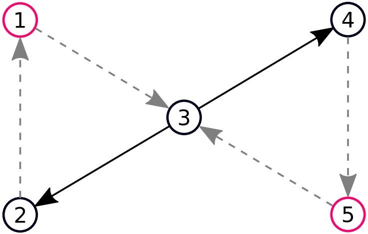


Figure: Suboptimal solution $F = \{1, 5\}$.

Definition (Directed Feedback Vertex Set Problem)

Given: Directed graph $G = (V, E)$

Task: Find $F \subseteq V$ of minimum cardinality, s.t. $G[V \setminus F]$ is acyclic.

F ... directed feedback vertex set (DFVS)

$G[V \setminus F]$ ... directed acyclic graph (DAG)

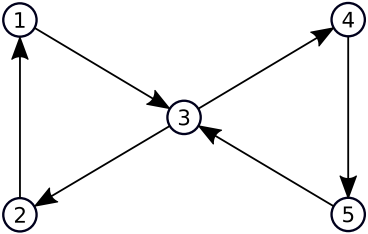


Figure: Example of a DFVS problem instance.

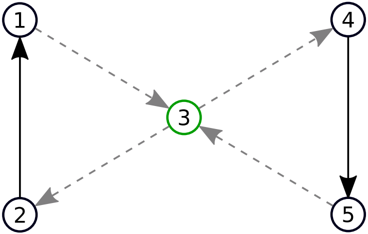


Figure: Optimal solution $F^* = \{3\}$.

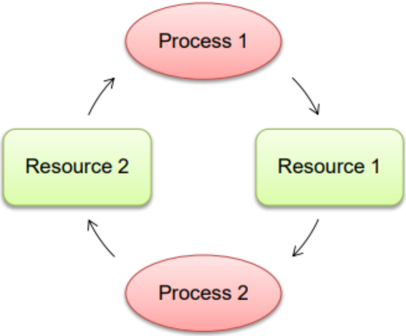
Problem Motivation

Applications:

- Deadlock detection and recovery
- Program verification
- Package dependencies

But:

- NP-complete problem



Related Work

DFVS problem:

- Simulated annealing (SA) metaheuristic by Galinier et al. (2013): SA-FVSP
- Extension of SA-FVSP with nonuniform neighborhood sampling (SA-FVSP-NNS) by Tang et al. (2017)
- Heuristic solvers from the Parameterized Algorithms and Computational Experiments (PACE) 2022 challenge[†]

Undirected weighted FVS problem:

- MILS$^+$ by Melo et al. (2021)
    - multi-start iterated local search (MILS) + MIP-based local search

[†] https://pacechallenge.org/2022/results/

Solving Approach

Exact: Mixed Integer Linear Programming (MILP)

Heuristic: Large Neighborhood Search (LNS)

Idea: combine LNS and MILP

- Large neighborhoods with complex move operators
  - Destroy operator
  - Repair operator
- MILP formulation: optimally solve subproblem in repair operator
- Important parameter: degree of destruction k
- Initial solution: construction heuristic + local search

MILP Formulations

Formulation inspired by subtour elimination constraints from Miller, Tucker, and Zemlin (MTZ):

- derived from formulation by Melo et al. (2021)

Formulation based on cycle elimination constraints (CECs):

- initial model strengthened with clique constraints based on cycles of length two (2-cycles)
- lazy constraint generation for more general CECs

Enlarge-DAG Neighborhood Structure

- destroy: move multiple vertices from DFVS to DAG
- repair: solve smaller DFVS subproblem

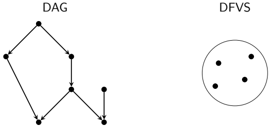


Figure: Initial state with current solution.

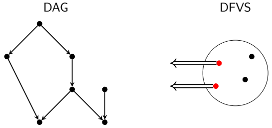


Figure: Element selection.

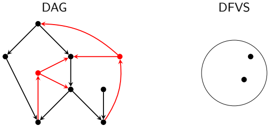


Figure: Enlarged DAG.

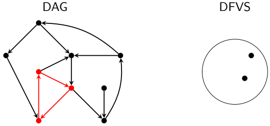


Figure: Enlarged DAG with exemplary cycle.

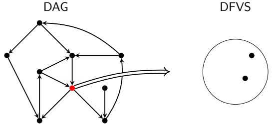


Figure: Solving the subproblem.

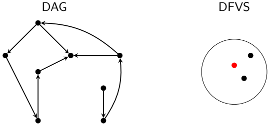


Figure: Final state with new solution.

Selection Strategies

Degree of Destruction k

k = number of selected vertices in the destroy operator

Simple selection:

- fixed_degree(x): constant value x
- random selection: from a predefined range

Advanced dynamic selection:

- 5 strategies
- based on graph properties and/or MILP formulation
- rules to predict suitable values for each instance

Selection Strategies

Degree of Destruction k

#2-cycles:

2-cycles Partition		Instances	k	
From	To		CEC	MTZ
0	100	22	50	25
101	10000	11	200	75000
10001	100000	16	200	5000
100001	200000	14	2000	1000
200001	1000000	15	2000	500
1000001	1200000	12	3000	1000
1200001	∞	10	50000	3000

2 approaches:

- uninformed preselection
- depending on instance characteristics:
  - number of 2-cycles
  - graph density

Computational Study

Setup

- implementation in `Julia 1.7.1`
- `Gurobi 9.5.1` via `JuMP`
  - memory limit of 20 GB
  - time limit of 90 s
- general time limit of 550 s
- Intel Xeon E5540 with 2.53 GHz

Table: Data sets used for the computational study.

Data Set	Size	Number of Vertices		Number of Edges	
		$n_{\min}$	$n_{\max}$	$m_{\min}$	$m_{\max}$
pace-public	100	843	875713	2103	5105039
pace-private	100	1024	2394385	3473	5021410
fsp-data	40	50	1000	100	30000

Computational Study

Benchmark Instances

Table: Data sets used for the computational study.

Data Set	Size	Number of Vertices		Number of Edges	
		n_{min}	n_{max}	m_{min}	m_{max}
pace-public	100	843	875713	2103	5105039
pace-private	100	1024	2394385	3473	5021410
fsp-data	40	50	1000	100	30000

Evaluation:

- solution quality: $100\% \frac{|F_i^*|}{|F_i|}$
  - instance i, solution F_i, best known solution F_i^*
- geometric mean

Results - Selection Strategies for k

Dynamic Selection

Selection Strategy	Formulation	Average Solution Quality [%]		Best Known Solutions	
		pace-public	pace-private	pace-public	pace-private
#2-cycles	CEC	96.15	98.83	12	26
	MTZ	94.57	97.25	13	17
	dynamic	**96.21**	**98.96**	15	26
best_triple	CEC	94.53	96.33	**32**	**39**
	MTZ	93.68	96.48	13	15
#2-cycles_best_triple	CEC	95.84	97.48	17	27
	MTZ	94.34	97.10	13	17
#2-cycles_regression	CEC	95.62	97.53	15	25
	MTZ	93.89	95.83	8	12
	dynamic	95.63	97.44	14	21
#vertices_regression	CEC	94.81	96.77	9	27
	MTZ	93.62	95.81	6	10
	dynamic	94.80	96.79	9	20

Results - Selection Strategies for k

Dynamic Selection

Selection Strategy	Formulation	Average Solution Quality [%]		Best Known Solutions	
		pace-public	pace-private	pace-public	pace-private
#2-cycles	CEC	96.15	98.83	12	26
	MTZ	94.57	97.25	13	17
	dynamic	**96.21**	**98.96**	15	26
best_triple	CEC	94.53	96.33	**32**	**39**
	MTZ	93.68	96.48	13	15
#2-cycles_best_triple	CEC	95.84	97.48	17	27
	MTZ	94.34	97.10	13	17
#2-cycles_regression	CEC	95.62	97.53	15	25
	MTZ	93.89	95.83	8	12
	dynamic	95.63	97.44	14	21
#vertices_regression	CEC	94.81	96.77	9	27
	MTZ	93.62	95.81	6	10
	dynamic	94.80	96.79	9	20

Comparison with Literature

SA-based metaheuristics:

- SA-FVSP (Galinier et al. (2013))
- SA-FVSP (Tang et al. (2017))
- SA-FVSP-NNS (Tang et al. (2017))

Benchmark instances:

Table: Data sets used for the computational study.

Data Set	Size	Number of Vertices		Number of Edges	
		$n_{\min}$	$n_{\max}$	$m_{\min}$	$m_{\max}$
pace-public	100	843	875713	2103	5105039
pace-private	100	1024	2394385	3473	5021410
fsp-data	40	50	1000	100	30000

Comparison with Literature

Results

MILP-based LNS Algorithms		Average Solution Quality [%]	Best Known Solutions
Selection Strategy	Formulation		
#2-cycles	CEC	**96.37**	18
	MTZ	95.49	15
	dynamic	96.01	15
best_triple	CEC	94.77	**19**
	MTZ	93.38	15
#2-cycles_best_triple	CEC	96.23	**19**
	MTZ	95.54	15
#2-cycles_regression	CEC	95.71	18
	MTZ	94.80	15
	dynamic	95.57	15
#vertices_regression	CEC	95.58	18
	MTZ	94.53	15
	dynamic	95.42	15
SA Algorithms			
SA-FVSP‡		**99.77**	**27**
SA-FVSP§		63.24	1
SA-FVSP-NNS§		70.40	1

‡ by Galinier et al. (2013)

§ by Tang et al. (2017)

Conclusion & Future Work

- LNS outperforms MILP solving when having a time limit
- Degree of destruction important for performance of MILP-based LNS
- Dynamic MILP selection sometimes beneficial
  $\rightarrow$ still room for improvement
- Investigate machine learning approaches for algorithm configuration
- Extend graph reduction procedure

Thank you!

References I

Xuan Cai, Jingwei Huang, and Guoqiang Jian. Search algorithm for computing minimum feedback vertex set of a directed graph. *Jisuanji Gongcheng/Computer Engineering*, 32(4):67–69, 2006.

Philippe Galinier, Eunice Lemamou, and Mohamed Wassim Bouzidi. Applying local search to the feedback vertex set problem. *Journal of Heuristics*, 19(5):797–818, 2013.

Hanoch Levy and David W. Low. A contraction algorithm for finding small cycle cutsets. *Journal of Algorithms*, 9(4):470–493, 1988.

Rafael A. Melo, Michell F. Queiroz, and Celso C. Ribeiro. Compact formulations and an iterated local search-based matheuristic for the minimum weighted feedback vertex set problem. *European Journal of Operational Research*, 289(1):75–92, 2021.

Sungju Park and Sheldon B. Akers. An efficient method for finding a minimal feedback arc set in directed graphs. In *[Proceedings] 1992 IEEE International Symposium on Circuits and Systems*, volume 4, pages 1863–1866. IEEE, 1992. ISBN 9780780305939.

Zhipeng Tang, Qilong Feng, and Ping Zhong. Nonuniform neighborhood sampling based simulated annealing for the directed feedback vertex set problem. *IEEE Access*, 5:12353–12363, 2017.

MILP Formulations

MTZ

$$
\begin{aligned}
\min \quad & |V| - \sum_{v \in V} y_v && && (1)\\
\text{s.t.} \quad & \Phi_u - \Phi_v + |V_s| \cdot y_v \leq |V_s| - 1 && \forall (u,v) \in E_s && (2)\\
& y_v \leq \Phi_v && \forall v \in V && (3)\\
& y_s = 1 && && (4)\\
& \Phi_s = 0 && && (5)\\
& y_v \in \{0,1\} && \forall v \in V && (6)\\
& 0 \leq \Phi_v \leq |V_s| - 1 && \forall v \in V && (7)
\end{aligned}
$$

MILP Formulations
CEC

$$\min \ |V| - \sum_{v \in V} y_v \tag{8}$$

$$\text{s.t.} \ \sum_{v \in C} y_v \le |C| - 1 \qquad \forall C \in \mathcal{C} \tag{9}$$

$$\sum_{v \in K} y_v \le 1 \qquad \forall K \in \mathcal{K} \tag{10}$$

$$y_v \in \{0, 1\} \qquad \forall v \in V \tag{11}$$

Construction Heuristic & Local Search

Idea:

- create initial solution with construction heuristic (CH)
- improve with local search (LS)

CH: based on greedy function (Cai et al. (2006)) and topological ordering

$$h(v) = \deg^-(v) + \deg^+(v) - \lambda \cdot |\deg^-(v) - \deg^+(v)| \tag{12}$$

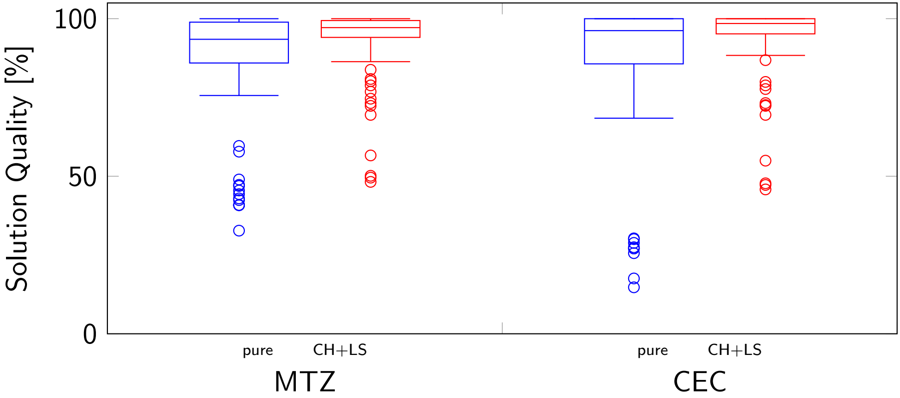
LS: one-flip neighborhood

- move 1 vertex from DFVS to DAG

Results - MILP Formulations

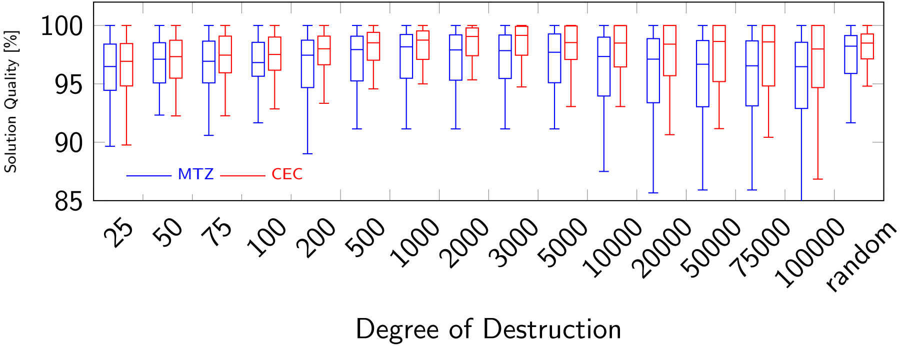
pace-public

Formulation	Avg. Solution Quality [%]	Best Known Solutions
MTZ_{pure}	84.93	18
MTZ_{CH+LS}	92.58	18
CEC_{pure}	82.59	38
CEC_{CH+LS}	93.43	38



Results - Selection Strategies for k

Simple Selection on pace-public



Selection Strategy	MILP	Avg. Solution Quality [%]	Best Known Solutions
fixed_degree(25)	MTZ	93.42	8
	CEC	94.20	7
fixed_degree(75)	MTZ	92.92	7
	CEC	95.10	11
random	MTZ	93.72	6
	CEC	94.87	6

Benchmark Instances

Table: Data sets used for the computational study.

Data Set	Size	Number of Vertices		Number of Edges	
		$n_{\min}$	$n_{\max}$	$m_{\min}$	$m_{\max}$
pace-public	100	843	875713	2103	5105039
pace-private	100	1024	2394385	3473	5021410
fsp-data	40	50	1000	100	30000
fsp-data_50	10	50	50	100	900
fsp-data_100	10	100	100	200	1400
fsp-data_500	10	500	500	1000	7000
fsp-data_1000	10	1000	1000	3000	30000

Benchmark Instances - pace-public

pace-public instances: vertex number

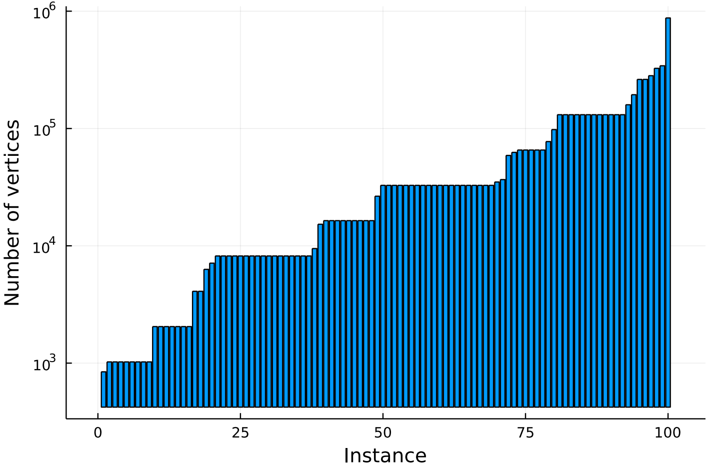


Figure: The number of vertices of each instance.

Benchmark Instances - pace-public

pace-public instances: edge number

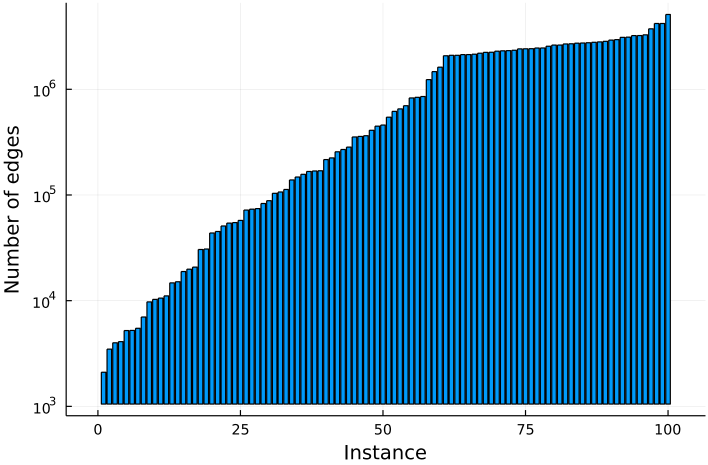


Figure: The number of edges of each instance.

Benchmark Instances - pace-public

pace-public instances: edges in 2-cycles

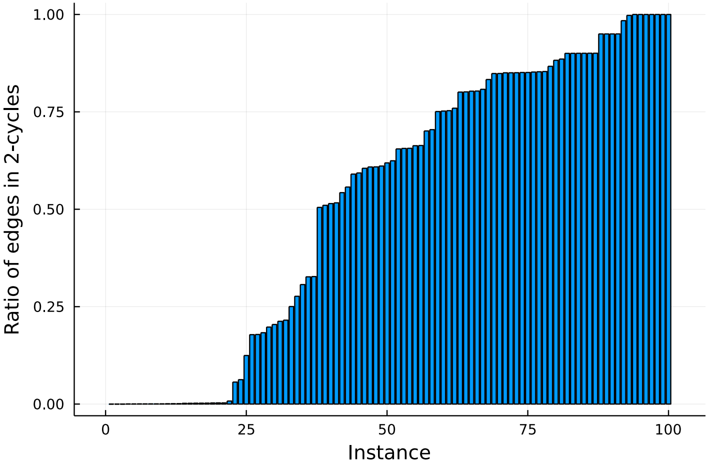


Figure: The ratio of edges involved in 2-cycles for each instance.

Graph Reduction

5 reduction rules inspired by Levy and Low (1988) and Park and Akers (1992)

- reducing $> 75\%$ of tested instances
- reductions of up to 100%
- average runtime: < 1 second

Partitioning into strongly connected components (SCCs)

- splits DFVS problem into smaller subproblems

Graph Reduction

5 reduction rules inspired by Levy and Low (1988) and Park and Akers (1992)

Rule IN/OUT0:

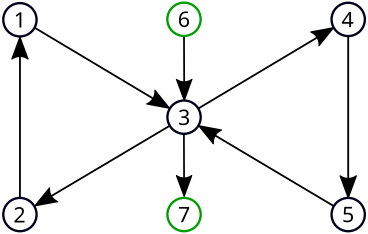


Figure: Initial graph.

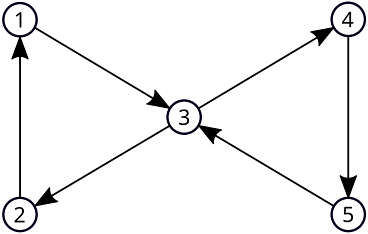


Figure: Reduced graph.

Graph Reduction

5 reduction rules inspired by Levy and Low (1988) and Park and Akers (1992)

Rule IN1:

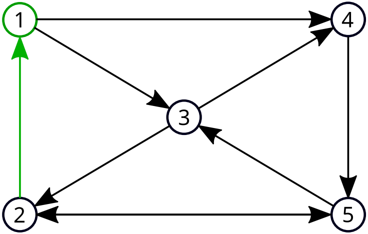


Figure: Initial graph.

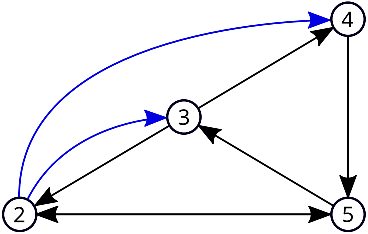


Figure: Reduced graph.

5 reduction rules inspired by Levy and Low (1988) and Park and Akers (1992)

Rule OUT1:

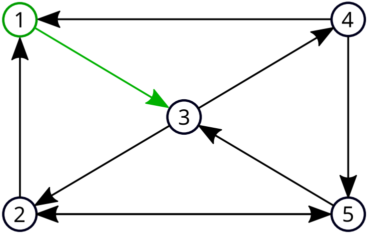


Figure: Initial graph.

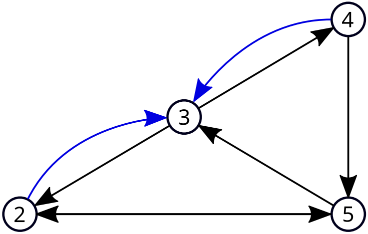


Figure: Reduced graph.

Graph Reduction

5 reduction rules inspired by Levy and Low (1988) and Park and Akers (1992)

Rule LOOP:

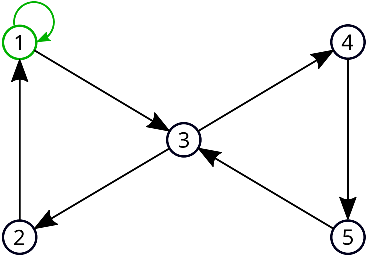


Figure: Initial graph.

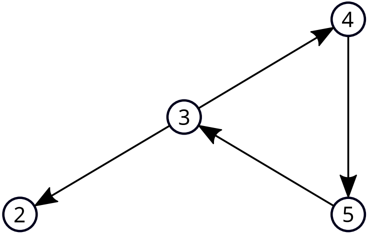


Figure: Reduced graph.

Graph Reduction

5 reduction rules inspired by Levy and Low (1988) and Park and Akers (1992)

Loop generation:

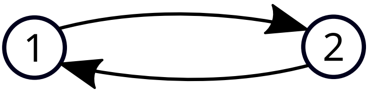


Figure: Initial graph.

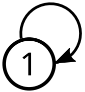


Figure: Reduced graph.

Graph Reduction

5 reduction rules inspired by Levy and Low (1988) and Park and Akers (1992)

Rule SCC:

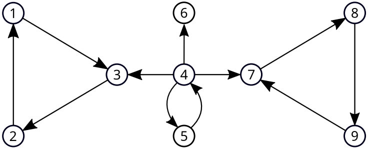


Figure: Initial graph.

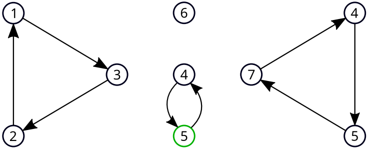


Figure: Partitioned graph.

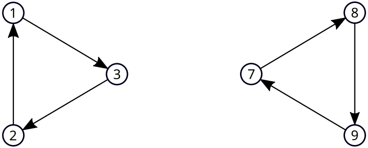


Figure: Reduced graph.

Dynamic Selection Strategies for *k*

- #2-cycles
- best_triple
- #2-cycles_best_triple
- #2-cycles_regression
- #vertices_regression

Dynamic Selection Strategies for k

#2-cycles:

- predefine partitions for the number of 2-cycles
- preselect the best performing degree of destruction for each partition
- differentiate between MILP formulations

2-cycles Partition		Instances	k	
From	To		CEC	MTZ
0	100	22	50	25
101	10000	11	200	75000
10001	100000	16	200	5000
100001	200000	14	2000	1000
200001	1000000	15	2000	500
1000001	1200000	12	3000	1000
1200001	∞	10	50000	3000

Dynamic Selection Strategies for k

best_triple:

- preselect 3 values: best-mean, mode, most-best
- random selection in each LNS iteration
- differentiate between MILP formulations
- independent of graph properties

Formulation	Degree of Destruction k		
	best-mean	mode	most-best
CEC	75	3000	75000
MTZ	25	25	1000 5000

Dynamic Selection Strategies for k

#2-cycles_best_triple:

- combination of #2-cycles and best_triple
- reuse 2-cycle partitions
- preselect 3 values for each partition and each MILP formulation: best-2cycle-mean, most-2cycle-best, best-mean
- random selection in each LNS iteration

2-cycles Partition		Degree of Destruction k					
From	To	best-2cycle-mean		most-2cycle-best		best-mean	
		CEC	MTZ	CEC	MTZ	CEC	MTZ
0	100	50	25	75	25	75	25
101	10000	200	75000	500	500	75	25
10001	100000	200	5000	2000	1000	75	25
100001	200000	2000	1000	5000	5000	75	25
200001	1000000	2000	500	1000	500	75	25
1000001	1200000	3000	1000	2000	500	75	25
1200001	∞	50000	3000	2000	1000	75	25

Dynamic Selection Strategies for k

#2-cycles_regression:

- linear correlation between the base-10 logarithmic value of the number of 2-cycles and the base-10 logarithmic value of the lowest best degree of destruction
- function defined by regression line
- differentiate between MILP formulations

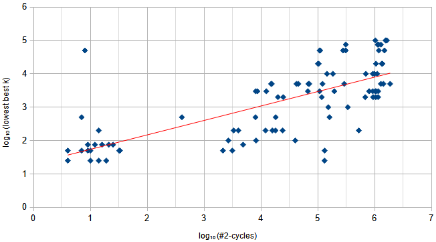


Figure: CEC-based formulation.

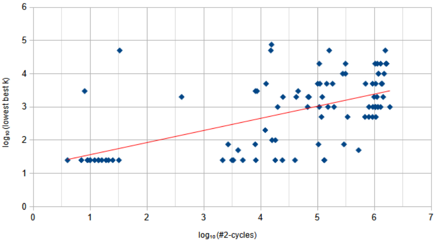


Figure: MTZ-based formulation.

Dynamic Selection Strategies for k

#2-cycles_regression:

- linear correlation between the base-10 logarithmic value of the number of 2-cycles and the base-10 logarithmic value of the lowest best degree of destruction
- function defined by regression line
- differentiate between MILP formulations

$$k = \begin{cases} 15.85 \cdot z^{0.365} & \text{for the MTZ model} \\ 20.14 \cdot z^{0.433} & \text{for the CEC model,} \end{cases} \tag{13}$$

$z \ldots$ number of 2-cycles

Dynamic Selection Strategies for k

#vertices_regression:

- linear correlation between the base-10 logarithmic value of the number of vertices and the base-10 logarithmic value of the lowest best degree of destruction
- function defined by regression line
- differentiate between MILP formulations

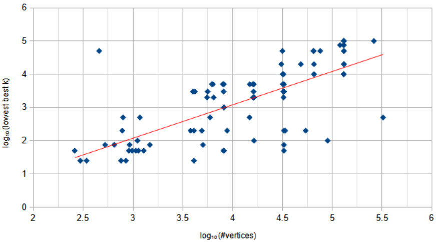


Figure: CEC-based formulation.

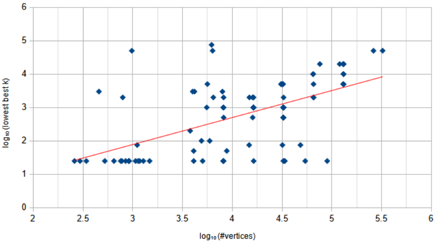


Figure: MTZ-based formulation.

Dynamic Selection Strategies for k

#vertices_regression:

- linear correlation between the base-10 logarithmic value of the number of vertices and the base-10 logarithmic value of the lowest best degree of destruction
- function defined by regression line
- differentiate between MILP formulations

$$k = \begin{cases} 0.2917 \cdot |V|^{0.808} & \text{for the MTZ model} \\ 0.1159 \cdot |V|^{1.004} & \text{for the CEC model} \end{cases} \tag{14}$$

Dynamic Selection of MILP Formulation

- predefine partitions for the product of the number of 2-cycles and the graph density
- preselect the MILP formulation for each partition
- differentiate between direct MILP and hybrid LNS

MILP-based LNS:

2-cycles × Density Partition		MILP Formulation
From	To	
0	5	CEC
5	20	MTZ
20	700	CEC
700	1000	MTZ
1000	3000	CEC
3000	7000	MTZ
7000	∞	CEC

Dynamic Selection of MILP Formulation

- predefine partitions for the product of the number of 2-cycles and the graph density
- preselect the MILP formulation for each partition
- differentiate between direct MILP and hybrid LNS

Direct MILP:

2-cycles × Density Partition		MILP Formulation
From	To	
0.00	0.04	MTZ
0.04	0.40	CEC
0.40	20.00	MTZ
20.00	50.00	CEC
50.00	150.00	MTZ
150.00	500.00	CEC
500.00	∞	MTZ