ParaILP: A Parallel Local Search Framework for Integer Linear Programming with Cooperative Evolution Mechanism (Technical-Appendix)

Peng Lin[1,2] , **Mengchuan Zou**[1] , **Zhihan Chen**[1,2] , **Shaowei Cai**[1,2]*

[1] Key Laboratory of System Software (Chinese Academy of Sciences) and State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, China

[2] School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing, China

{linpeng, zoumc, chenzh, caisw}@ios.ac.cn,

1 Stability of ParaILP

To examine the stability of ParaILP which involves randomness, we execute ParaILP 10 times using 10 different seeds on the benchmark for 10s, 60s, and 300s time limits.

For all 10 times, we denote the average primal integral *P(T)* of each time by $\mathrm{avg}_{\mathrm{P(T)}}$, and the standard deviation of the primal integral by $\mathrm{std}_{\mathrm{P(T)}}$. As shown by the results in Table 1, for the time limits of 10s and 60s, the values of $\frac{\mathrm{std}_{\mathrm{P(T)}}}{\mathrm{avg}_{\mathrm{P(T)}}}$ for ParaILP are less than 0.5%; for the 300s, the values of $\frac{\mathrm{std}_{\mathrm{P(T)}}}{\mathrm{avg}_{\mathrm{P(T)}}}$ for ParaILP are less than 1.3%, indicating ParaILP exhibits stable performance.

Table 1: Performance of ParaILP with 10 different seeds.

Time Limit	$\mathrm{avg}_{\mathrm{P(T)}}$	$\mathrm{std}_{\mathrm{P(T)}}$	$\mathrm{std}_{\mathrm{P(T)}}/\mathrm{avg}_{\mathrm{P(T)}}$
10s	0.58593	0.00245	0.00418
60s	0.51331	0.00228	0.00445
300s	0.46316	0.00576	0.01244

2 Result on Real-world Benchmarks

We test our solver on two practical problems, including the bin packing and the scheduling problem, which are challenging combinatorial optimization problems, and also have various applications in real-world industry. The benchmarks include 1 standard bin packing benchmark provided by [Falkenauer, 1996], and 2 standard scheduling benchmarks provided by Taillard's instances [Taillard, 1993].

- **BBP**: the Bin Packing problem, This benchmark consists of 60 instances with 500 and 1000 items to pack, encoded by the modeling method proposed in [Delorme *et al.*, 2016].
- **JSP**: the Job-shop Scheduling problem. This benchmark consists of 80 instances encoded by the modeling method proposed in [Ku and Beck, 2016].
- **OSP**: the Open-shop Scheduling problem. This benchmark consists of 60 instances encoded by the modeling method proposed in [Naderi and Zandieh, 2014].

As shown in Table 2, 3, 4 and 5, ParaILP is significantly better than the state-of-the-art academic solvers FiberSCIP and HiGHS, and is competitive with the state-of-the-art commercial solver Gurobi.

*Corresponding author

Table 2: Performance evaluation between SOTA academic solvers HiGHS, FiberSCIP and ParaILP in terms *#Feas*.

Benchmarks	#ins	HiGHS	FiberSCIP	ParaILP
time limit 10 seconds				
JSP	80	28	**68**	50
OSP	60	48	56	**60**
BPP	60	40	0	**60**
time limit 60 seconds				
JSP	80	40	63	**70**
OSP	60	54	**60**	**60**
BPP	60	40	17	**60**
time limit 300 seconds				
JSP	80	49	65	**79**
OSP	60	**60**	59	**60**
BPP	60	50	54	**60**

Table 3: Performance evaluation between SOTA academic solvers HiGHS, FiberSCIP and ParaILP in terms *#Win*.

Benchmarks	#ins	HiGHS	FiberSCIP	ParaILP
time limit 10 seconds				
JSP	80	0	26	**44**
OSP	60	22	27	**52**
BPP	60	0	0	**60**
time limit 60 seconds				
JSP	80	1	25	**44**
OSP	60	29	24	**50**
BPP	60	0	0	**60**
time limit 300 seconds				
JSP	80	3	27	**49**
OSP	60	39	29	**40**
BPP	60	0	**33**	27

References

[Delorme *et al.*, 2016] Maxence Delorme, Manuel Iori, and Silvano Martello. Bin packing and cutting stock problems: Mathematical models and exact algorithms. *European Journal of Operational Research*, 255(1):1–20, 2016.

[Falkenauer, 1996] Emanuel Falkenauer. A hybrid grouping genetic algorithm for bin packing. *Journal of heuristics*, 2:5–30, 1996.

[Ku and Beck, 2016] Wen-Yang Ku and J Christopher Beck. Mixed integer programming models for job shop schedul-

Table 4: Performance evaluation between SOTA commercial solver Gurobi (both the exact and heuristic version) and ParaILP in terms *#Feas*.

Benchmarks	#ins	Gurobi$_{\mathrm{comp}}$	Gurobi$_{\mathrm{heur}}$	ParaILP
time limit 10 seconds				
JSP	80	37	31	**50**
OSP	60	**60**	49	**60**
BPP	60	**60**	**60**	**60**
time limit 60 seconds				
JSP	80	50	55	**70**
OSP	60	**60**	**60**	**60**
BPP	60	**60**	**60**	**60**
time limit 300 seconds				
JSP	80	54	70	**79**
OSP	60	**60**	**60**	**60**
BPP	60	**60**	**60**	**60**

Table 5: Performance evaluation between SOTA commercial solver Gurobi (both the exact and heuristic version) and ParaILP in terms *#Win*.

Benchmarks	#ins	Gurobi$_{\mathrm{comp}}$	Gurobi$_{\mathrm{heur}}$	ParaILP
time limit 10 seconds				
JSP	80	8	13	**30**
OSP	60	34	27	**46**
BPP	60	1	0	**59**
time limit 60 seconds				
JSP	80	12	**38**	22
OSP	60	**40**	38	**40**
BPP	60	13	27	**33**
time limit 300 seconds				
JSP	80	12	**59**	13
OSP	60	**46**	43	33
BPP	60	**39**	38	7

ing: A computational analysis. *Computers & Operations Research*, 73:165–173, 2016.

[Naderi and Zandieh, 2014] B Naderi and M Zandieh. Modeling and scheduling no-wait open shop problems. *International Journal of Production Economics*, 158:256–266, 2014.

[Taillard, 1993] Eric Taillard. Benchmarks for basic scheduling problems. *european journal of operational research*, 64(2):278–285, 1993.