DynSep:
Dynamic Configuration for Cutting Plane **Sep**arators via Reinforcement Learning on Incremental Graph

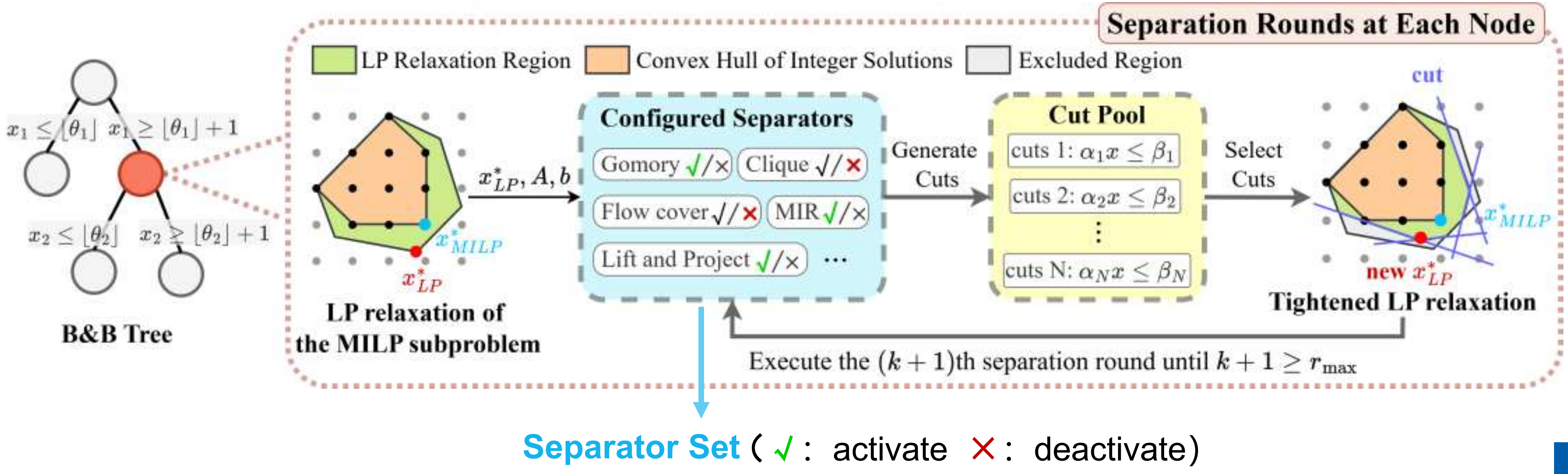
Presenter: Mingxuan Ye, Ph.D. Student
Advisor: Prof. Feng Wu, Prof. Jie Wang
MIRA Lab, University of Science and Technology of China

2025/11/05

DynSep

Background: How Cuts Are Born in MILP Solver

- **Cutting planes (cuts)** are essential for solving **mixed-integer linear programming (MILP)** problems, as they tighten the feasible solution space and accelerate the solving process.
- **Separators:** **cutting plane algorithms** built in MILP solver (e.g., Gomory cuts, Clique cuts).
- **Separator configuration:** Determine which separators to activate and how aggressively to activate. The configuration of separators determines the quality of candidate cuts and thus the solver's convergence behavior.
- **Multiple Separation Rounds in MILP Solver:**



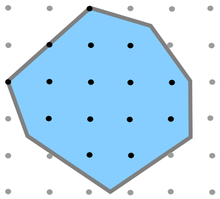
[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

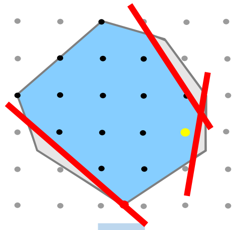
DynSep

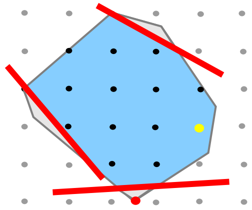
Motivation: The Optimal Separators Choice Keep Changing

Cut passes induce the evolving MILPs...

Original MILP → MILP added Cuts from 1st Separation Round → MILP added Cuts from 2nd Separation Round







Decision: Select Activated Separators at Each Round

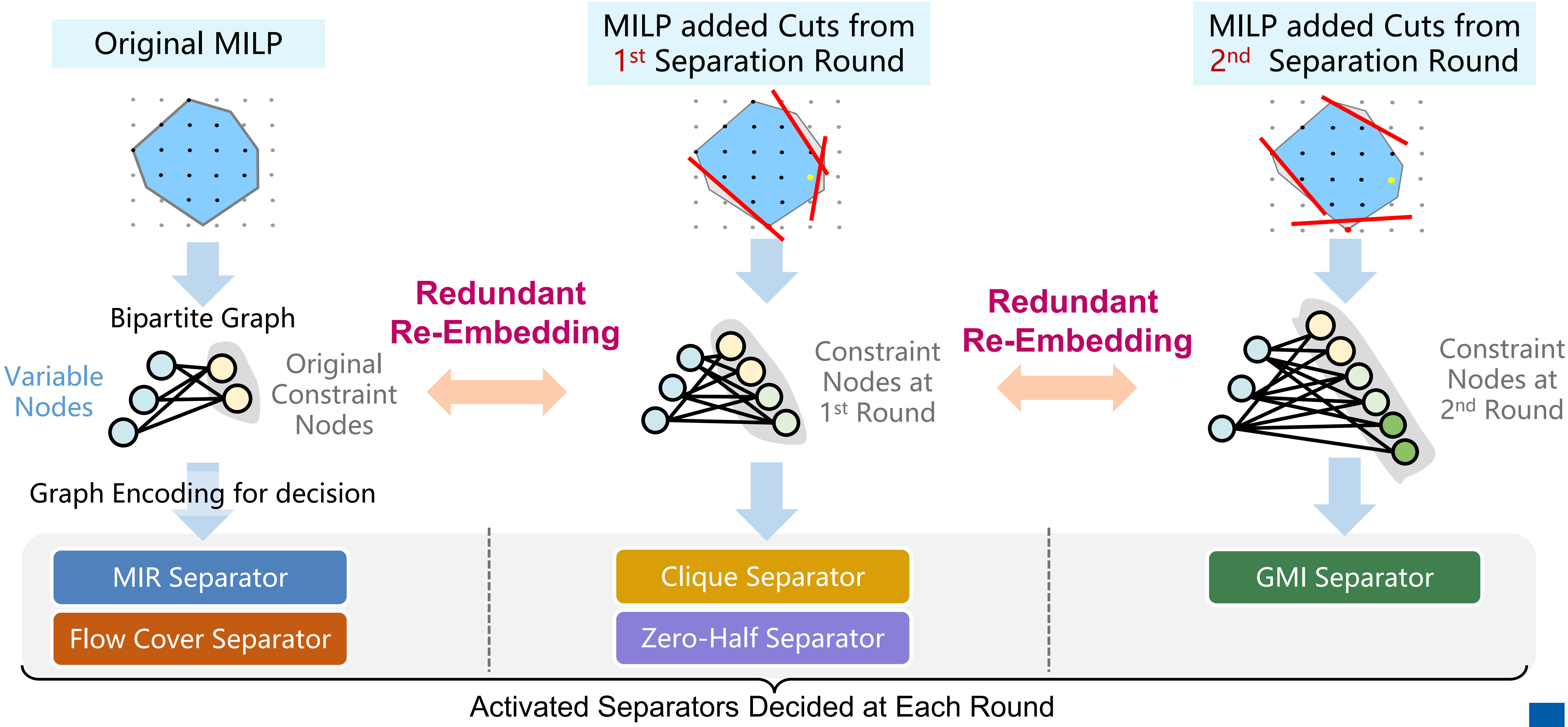
MIR Separator	Clique Separator	GMI Separator
Flow Cover Separator	Zero-Half Separator	

Evolving MILPs Require Round-Aware Separator Reconfiguration

[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

DynSep

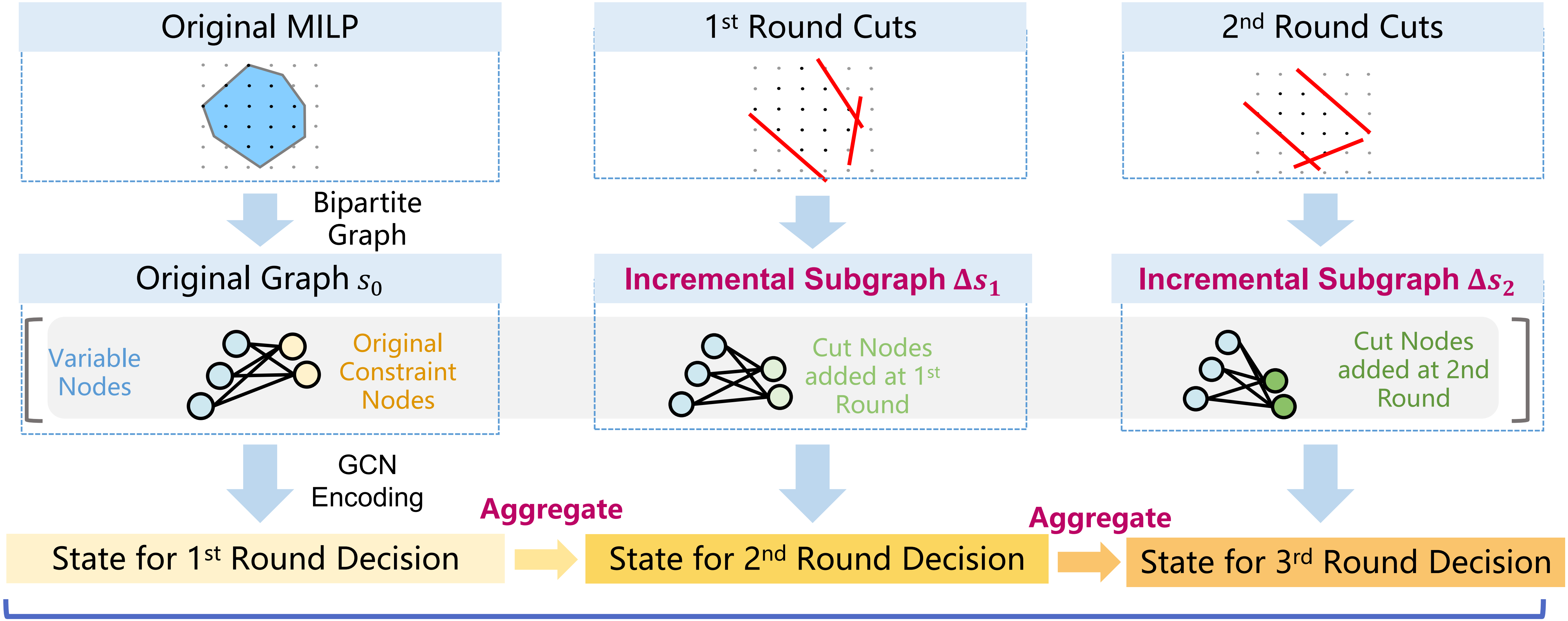
Motivation: Eolving Graph & Redundant Re-Embedding



[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

DynSep

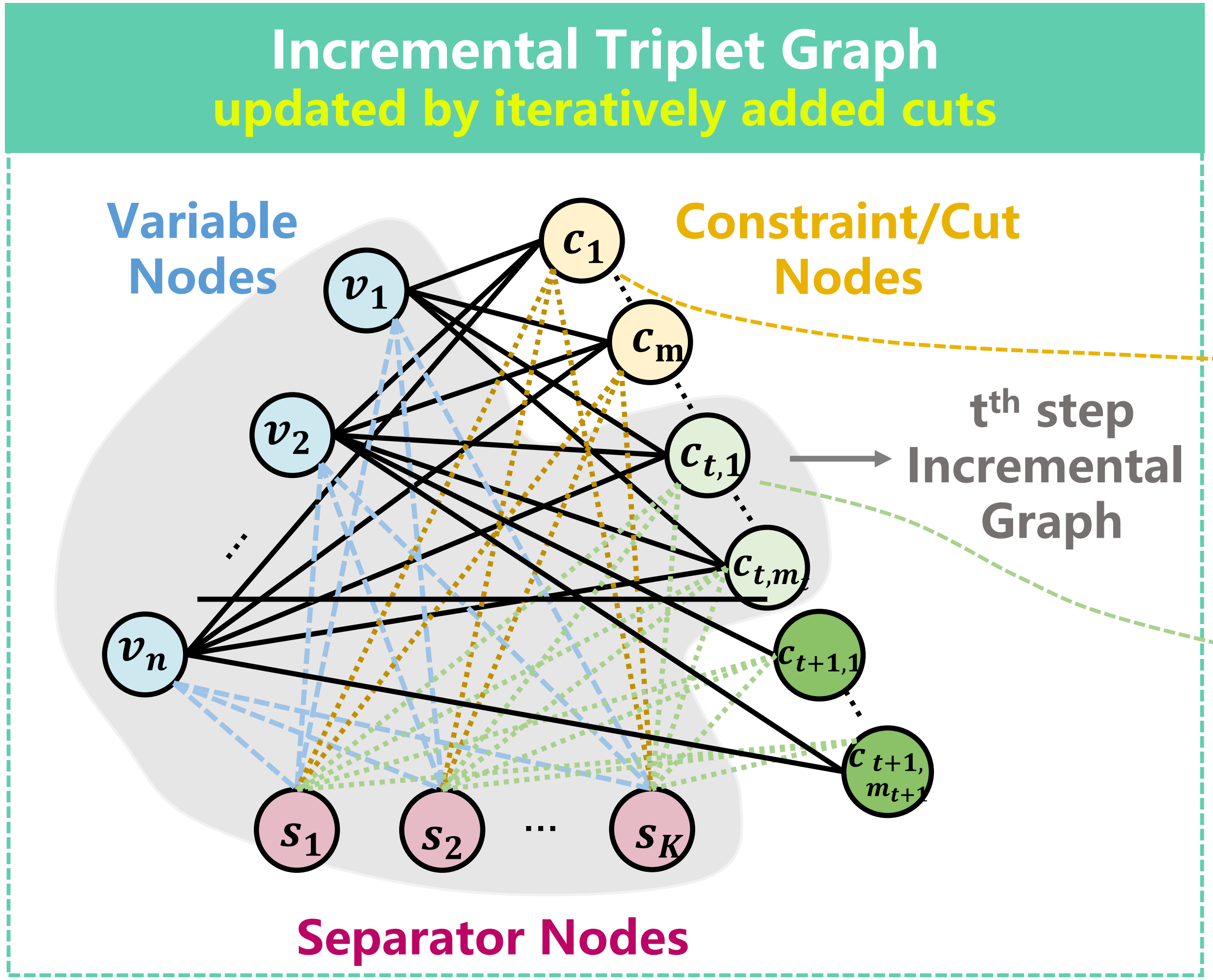
Insight: Computing the Incremental Graph at each decision



Aggregate current-round incremental changes with history to form the full decision state

[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

Method-Step 1: Triplet Graph Modeling



$$\delta_1:\ a_{11}x_1 + \cdots + a_{1n}x_n \le b_1$$
$$\vdots$$
$$\delta_m:\ a_{m1}x_1 + \cdots + a_{mn}x_n \le b_m$$

m rows of original constraints

$$\vdots$$

$$\delta_{t,1}:\ a_{(t,1),1}x_1 + \cdots + a_{(t,1),n}x_n \le b_{t,1}$$
$$\vdots$$
$$\delta_{t,m_t}:\ a_{(t,m_t),1}x_1 + \cdots + a_{(t,m_t),n}x_n \le b_{t,m_t}$$

Newly added cuts at t-th round

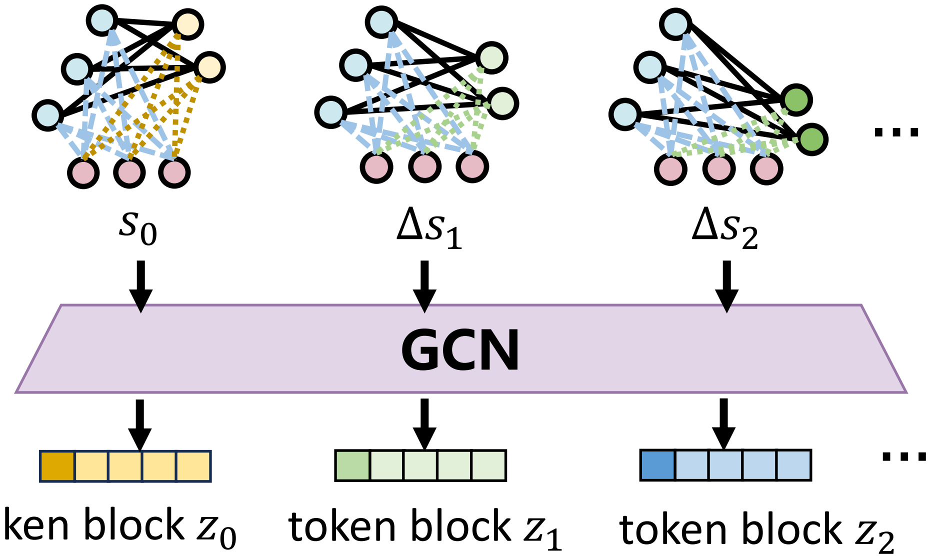
$$x_i \in \mathbb{Z}, \forall i \in I$$

[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

DynSep

Method-Step 2: Sequential Encoding & Tokenizer the Incremental Graph

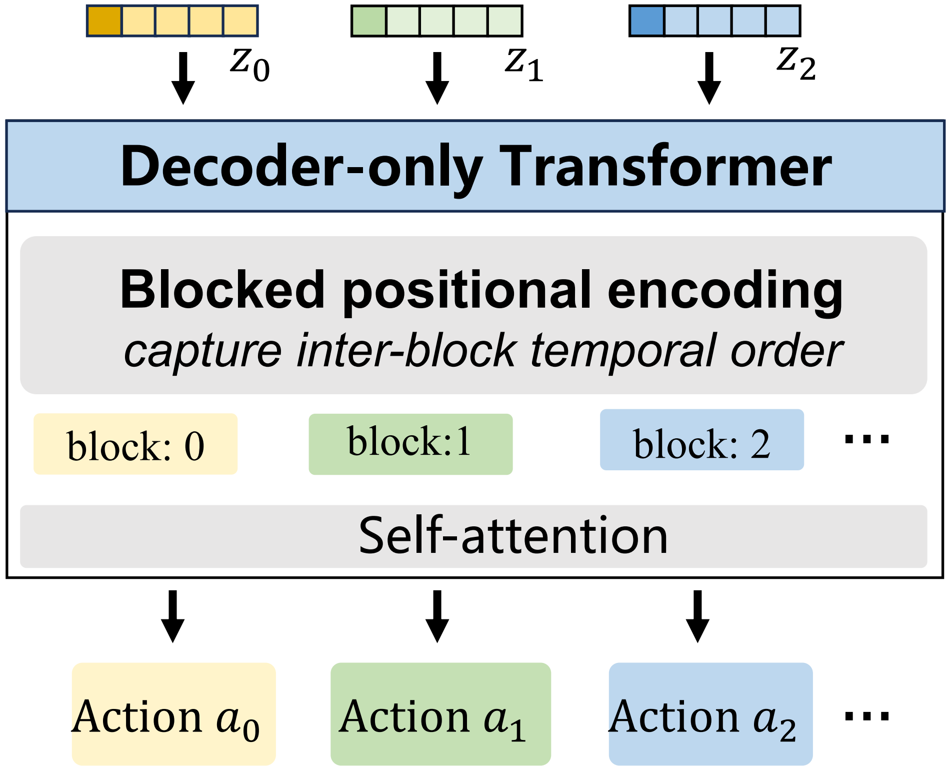
- Encode each round's incremental subgraph as a **token block** and arrange a temporal token sequence



$$z_t = \text{concat}(\text{global graph embedding}, \text{K separator−node embeddings})$$

Token sequence： $< z_0, z_1, z_2, \ldots >$

- Autoregressive decision model with blocked PE;
- captures inter-block order and retains **permutation equivariance** inside each block



[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

DynSep

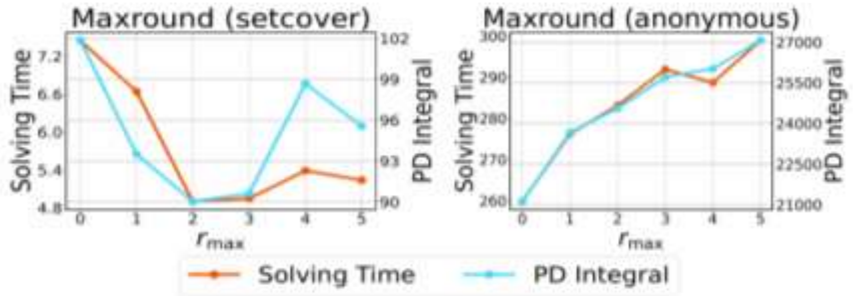
Method-Step 3: Decision Modeling

- Evolving MILPs Require **Round-Aware** Separator configuration:

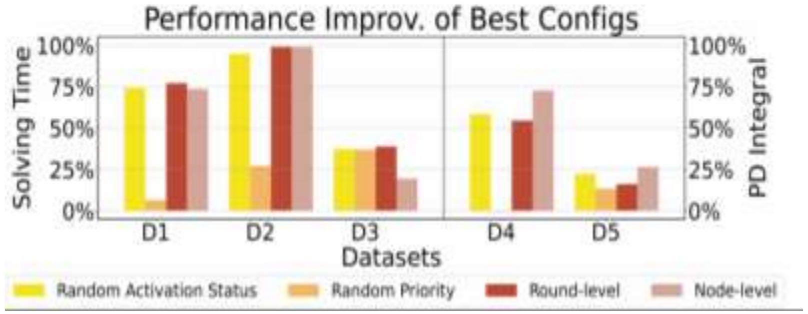
1

Decide when to stop separation round:

Performance is highly sensitive to r_{max}; More rounds do not necessarily perform better.

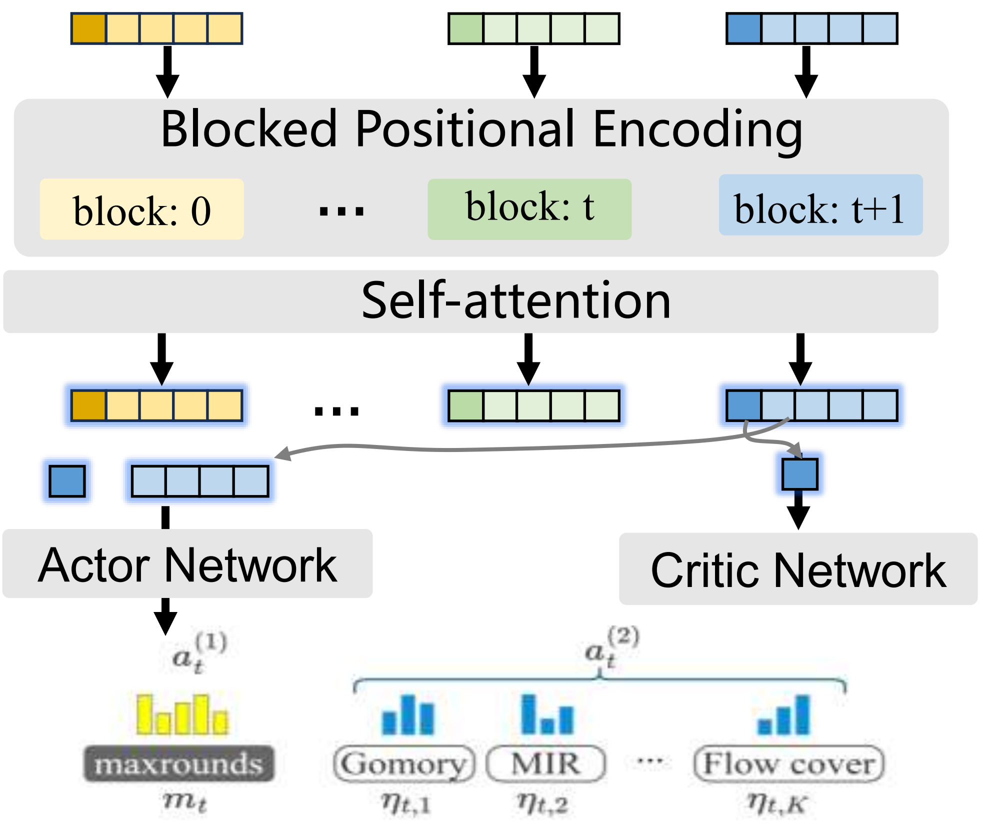


2



Decide which separators to activate:

Proper round-aware configurations yield obvious performance gains.

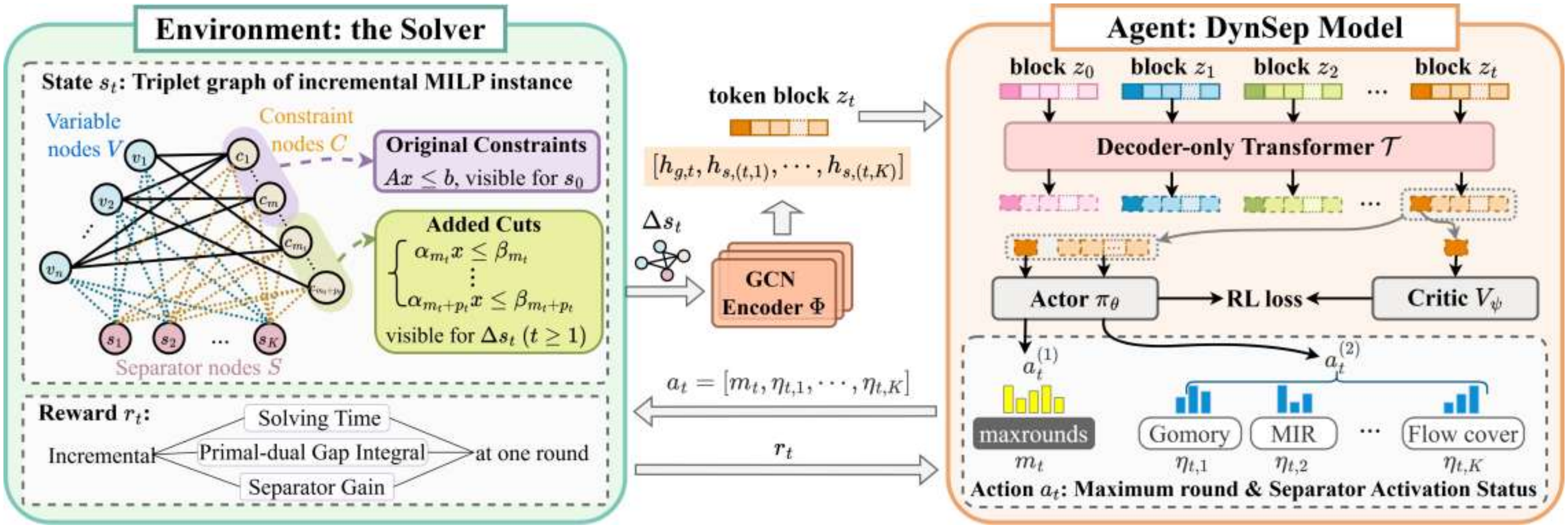


One-to-one mapping:
graph token ↔ maximum round decision
separator tokens ↔ configuration decisions

[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

DynSep

Method: Overall RL Formulation



DynSep

Experiments: Main Evaluation

- **64%** time speed-up on easy/medium datasets
- **16%** PD-integral reduction on hard datasets

Table 1: **Comparative evaluation on easy, medium, and hard datasets.** Best performance is in bold, with the greatest improvement (Improv.) both bolded and underlined. Sizes of nine benchmarks are in parentheses, with n and m representing the average numbers of variables and constraints, respectively. The values report the mean (standard deviation) of time and PD integral metrics.

	Easy: Set Covering (n = 1000, m = 500)			Easy: Max Independent Set (n = 500, m = 1953)			Easy: Multiple Knapsack (n = 720, m = 72)		
Method	Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓	Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓	Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓
NoCuts	7.45 (5.87)	NA	101.86 (55.59)	15.32 (5.82)	NA	146.4 (56.99)	13.84 (28.79)	NA	25.21 (26.6)
Default	5.24 (1.79)	29.66	95.56 (36.86)	30.4 (8.02)	-98.43	289.51 (103.81)	2.01 (1.82)	85.48	18.6 (10.49)
Search(50)	1.7 (0.44)	77.18	36.77 (8.48)	4.42 (3.89)	71.15	23.28 (19.02)	4.12 (5.52)	70.23	13.38 (7.36)
Prune	7.45 (5.14)	0.00	66.83 (45.97)	5.03 (3.18)	67.17	30.93 (22.39)	0.64 (0.48)	95.38	9.89 (3.82)
L2Sep(R1)	6.56 (4.35)	11.95	62.12 (39.08)	5.42 (3.86)	64.62	34.21 (25.97)	7.45 (12.07)	46.17	12.99 (8.66)
L2Sep(R2)	7.35 (4.88)	1.34	70.00 (43.57)	5.36 (3.76)	65.01	33.94 (25.33)	9.14 (12.89)	33.96	14.40 (8.72)
LLM4Sepasel	11.73 (12.09)	-57.45	110.73 (91.14)	5.13 (4.19)	66.51	27.18 (20.17)	4.8 (5.51)	65.32	17.79 (8.9)
DynSep (Ours)	**1.51 (0.27)**	**79.73**	**33.88 (9.34)**	**0.53 (0.20)**	**96.54**	**9.66 (2.40)**	**0.52 (0.24)**	**96.24**	**9.71 (5.39)**

	Medium: Corlat (n = 466, m = 486)			Medium: MIK (n = 413, m = 346)			Hard: Anonymous (n = 37881, m = 49603)		
Method	Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓	Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓	Time(s) ↓	PD integral ↓	Improv. ↑ (PD Int., %)
NoCuts	74.66 (122.23)	NA	2687.68 (6209.48)	190.28 (113.97)	NA	887.85 (859.76)	259.77 (75.71)	21117.12 (9234.01)	NA
Default	111.55 (132.19)	-49.41	10573.14 (13070.46)	16.65 (18.06)	91.25	82.80 (56.24)	298.92 (4.09)	27069.58 (4892.8)	-28.19
Search(50)	55.74 (97.19)	25.34	2910.77 (6585.5)	24.99 (20.56)	86.87	89.27 (55.85)	270.68 (65.52)	24028.68 (9007.57)	-13.79
Prune	89.09 (125.52)	-19.33	2615.71 (5814.74)	300.01 (0.0)	-57.67	2237.28 (1023.8)	241.75 (100.61)	17304.91 (9563.3)	18.05
L2Sep(R1)	91.14 (124.12)	-22.07	3124.07 (6914.50)	15.50 (17.60)	91.85	61.09 (44.50)	**239.52 (94.82)**	16970.35 (10108.40)	19.64
L2Sep(R2)	89.84 (124.30)	-20.33	3113.29 (6927.16)	11.13 (9.09)	94.15	**44.69 (25.06)**	240.54 (93.80)	16850.57 (10052.83)	20.20
LLM4Sepasel	64.03 (110.63)	14.24	2921.73 (6860.21)	17.94 (17.76)	90.57	85.66 (65.34)	284.57 (34.79)	25384.48 (8100.56)	-20.21
DynSep (Ours)	**22.96 (38.93)**	**69.25**	**2233.42 (3868.43)**	**10.99 (9.44)**	**94.22**	134.15 (44.21)	241.89 (100.75)	**15656.7 (8996.14)**	**25.86**

	Hard: Load Balancing (n = 61000, m = 64304)			Hard: MIPLIB mixed neos (n = 6958, m = 5660)			Hard: MIPLIB mixed supportcase (n = 19766, m = 19910)		
Method	Time(s) ↓	PD integral ↓	Improv. ↑ (PD Int., %)	Time(s) ↓	PD integral ↓	Improv. ↑ (PD Int., %)	Time(s) ↓	PD integral ↓	Improv. ↑ (PD Int., %)
NoCuts	300.11 (0.02)	15093.26 (940.68)	NA	275.04 (43.23)	14618.53 (12214.63)	NA	181.26 (120.25)	12959.99 (10506.47)	NA
Default	300.14 (0.02)	15187.19 (936.38)	-0.62	282.98 (29.49)	18500.5 (9386.15)	-26.56	244.75 (105.8)	21561.09 (10434.42)	-66.37
Search(50)	300.04 (0.05)	3783.52 (448.59)	74.93	274.23 (44.64)	15619.98 (11969.47)	-6.85	133.36 (131.32)	10241.17 (10794.69)	20.98
Prune	300.07 (0.12)	10597.31 (671.55)	29.79	249.37 (87.7)	14464.45 (12569.32)	1.05	158.63 (141.48)	9827.52 (11433.13)	24.17
L2Sep(R1)	**300.02 (0.03)**	10548.89 (4474.08)	30.11	242.83 (99.02)	10383.49 (11808.13)	28.97	162.18 (138.25)	11318.55 (11796.53)	12.67
L2Sep(R2)	300.03 (0.10)	10860.13 (4348.96)	28.05	242.90 (98.90)	13989.09 (12116.88)	4.31	166.23 (134.71)	11489.15 (11849.13)	11.35
LLM4Sepasel	300.04 (0.06)	4769.47 (709.05)	68.40	276.34 (47.32)	14109.36 (13706.18)	3.48	256.48 (100.88)	22618.21 (10234.21)	-74.52
DynSep (Ours)	300.04 (0.08)	**3720.26 (499.37)**	**75.35**	**235.19 (112.26)**	**8511.58 (12413.9)**	**41.78**	**132.50 (130.32)**	**9212.24 (9840.56)**	**28.92**

[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

Experiments: Ablation Study

Table 2: Ablation study comparing five variants of DynSep on three datasets.

	Easy: Multiple Knapsack (n = 720, m = 72)			Medium: Corlat (n = 466, m = 486)			Hard: MIPLIB mixed neos (n = 6958, m = 5660)		
Method	Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓	Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓	Time(s) ↓	PD integral ↓	Improv. ↑ (PD Int., %)
NoCuts	13.84 (28.79)	NA	25.21 (26.6)	74.66 (122.23)	NA	2687.68 (6209.48)	275.04 (43.23)	14618.53 (12214.63)	NA
Default	2.01 (1.82)	85.48	18.6 (10.49)	111.55 (132.19)	-49.41	10573.14	282.98 (29.49)	18500.5 (9386.15)	-26.56
w/o MaxR	0.67 (0.46)	95.16	10.07 (5.35)	28.28 (53.99)	62.12	2688.01 (5424.21)	263.63 (63.0)	9029.4 (12114.3)	38.23
w/o TF	0.64 (0.53)	95.38	9.82 (5.24)	25.82 (61.58)	65.42	2552.91 (6146.98)	245.04 (95.2)	9448.22 (11932.01)	35.37
w/o DynG	0.70 (1.23)	94.94	10.03 (5.86)	28.05 (50.97)	62.43	2635.47 (5086.84)	270.66 (50.83)	9233.22 (12003.39)	36.84
w/o DynG&TF	0.58 (0.41)	95.81	**9.65 (5.28)**	110.31 (131.6)	-47.75	10396.1 (12887.24)	300.0 (0.0)	17330.56 (9660.1)	-18.55
w/o BlockPE	0.93 (1.81)	93.28	11.12 (9.16)	34.4 (65.33)	53.92	2870.01 (5533.02)	242.39 (99.79)	**8291.49 (12533.67)**	**43.28**
DynSep (Ours)	**0.52 (0.24)**	**96.24**	9.71 (5.39)	**22.96 (38.93)**	**69.25**	**2233.42 (3868.43)**	**235.19 (112.26)**	8511.58 (12413.9)	41.78

[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

DynSep

Experiments: Generalization

Table 3: Evaluate the generalization ability of DynSep on MIS.

Method	Set Covering ($n = 1000,\ m = 1000,\ 2\times$) Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓	Set Covering ($n = 1000,\ m = 2000,\ 4\times$) Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓
NoCuts	78.91 (76.53)	NA	579.09 (507.14)	282.27 (52.86)	NA	3109.87 (1013.74)
Default	11.54 (3.55)	85.38	255.69 (92.95)	19.74 (8.06)	93.01	**606.90 (310.79)**
Search(30)	116.95 (92.25)	-48.821	912.87 (647.22)	292.54 (35.54)	-3.64	3827.701 (1057.08)
Prune	107.86 (90.55)	-36.69	826.38 (660.27)	295.01 (29.73)	-4.51	4805.02 (1302.02)
L2Sep(R1)	105.48 (92.68)	-33.67	803.32 (656.89)	294.85 (31.41)	-4.46	4728.1 (1325.35)
L2Sep(R2)	116.43 (95.6)	-47.55	905.98 (694.71)	293.77 (33.96)	-4.07	4481.42 (1377.69)
LLM4Sepasel	149.28 (105.45)	-89.18	1115.22 (767.04)	295.4 (29.34)	-4.65	3881.22 (1001.51)
DynSep (Ours)	**4.03 (0.64)**	**94.89**	**104.62 (20.19)**	**19.05 (26.92)**	**93.25**	629.28 (1106.36)

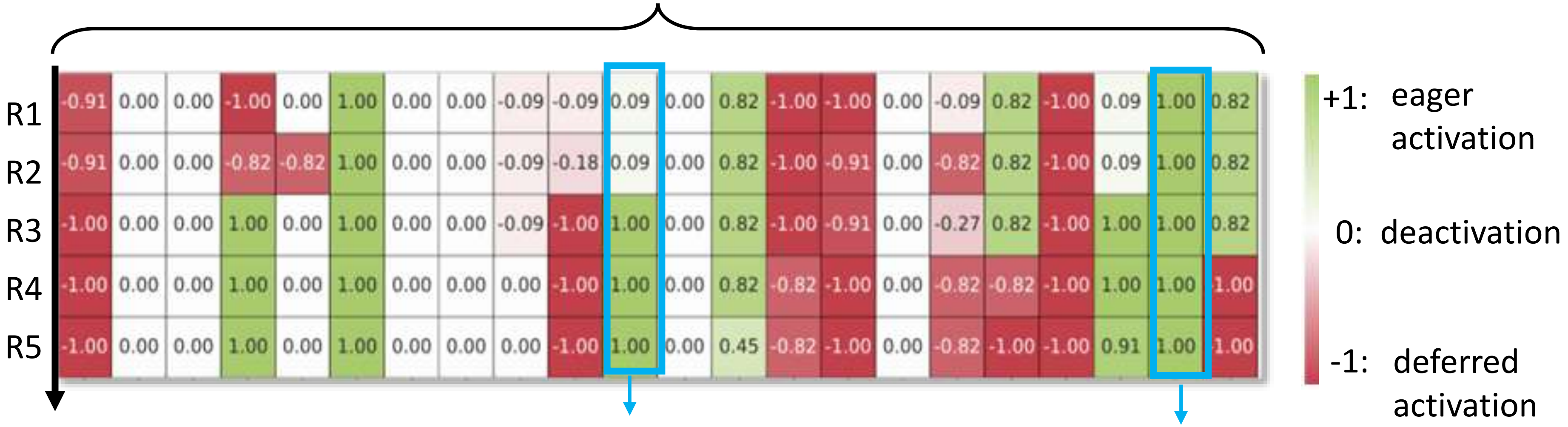
Method	Max Independent Set ($n = 1000,\ m = 3946,\ 4\times$) Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓	Max Independent Set ($n = 1500,\ m = 5940,\ 9\times$) Time(s) ↓	Improv. ↑ (time, %)	PD integral ↓
NoCuts	195.53 (95.78)	NA	1056.83 (544.51)	300.01 (0.01)	NA	2226.72 (370.66)
Default	88.17 (66.05)	54.91	813.36 (512.14)	177.19 (91.28)	40.94	1782.96 (887.23)
Search(30)	151.51 (98.42)	22.51	462.18 (339.23)	299.08 (9.17)	0.31	1251.49 (332.08)
Prune	105.83 (86.46)	45.88	396.8 (318.97)	292.94 (26.42)	2.36	1312.04 (343.31)
L2Sep(R1)	144.67 (94.46)	26.01	546.58 (370.45)	299.34 (6.19)	0.22	1504.9 (354.98)
L2Sep(R2)	138.82 (92.89)	29.00	530.09 (367.16)	299.49 (5.2)	0.17	1494.7 (346.61)
LLM4Sepasel	60.68 (42.03)	68.97	222.88 (143.29)	264.44 (58.96)	11.86	942.49 (337.13)
DynSep (Ours)	**5.47 (19.09)**	**97.20**	**36.75 (71.49)**	**26.66 (77.49)**	**91.11**	**151.39 (364.6)**

[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

DynSep

Experiments: Visualization

Knapsack family: Round-wise Activation Status of 22 SCIP Separators



Gomory: activate later
(effective in later rounds)

MIR: effective for
knapsack-structured instances

[1] **Mingxuan Ye**, Jie Wang*, et.al. Dynamic Configuration for Cutting Plane Separators via Reinforcement Learning on Incremental Graph. **NeurIPS 2025.**

Thanks for your