Learning Cut Selection for Mixed-Integer Linear Programming via Hierarchical Sequence Modeln

Published in ICLR 2023

Wang Zhihai, Li Xijun, Wang Jie, Kuang Yufei, Yuan Mingxuan, Zeng Jia, Zhang Yongdong, Wu Feng

Lecture of Mathieu Lacroix

03/28/2023

Outline

Outline

Solving MILP with Branch-and-cut

Branch-and-cut algorithm

1. Solve the root node with strengthening inequalities added dynamically (cutting plane algorithm)
2. Follow with a branch-and-bound algorithm (keeping the generated inequalities but no new ones)

Cuttting plane algorithm

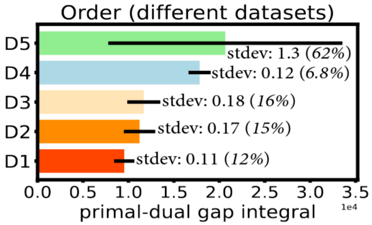
Repeat:

1. Solve current linear relaxation
2. Generate strengthening cust (cut generation)
3. select only a subset of inequalities (cut selection)

This work uses machine learning for improving cut selection.

MILP Resolution depends on the order of the constraints

In cutting plane, the order in which the strengthening cuts are added matters!



- Different datasets
- Solve each instance 10 times (one round of cutting plane?) by adding all generated cuts but with different random orders.

Remark

Same thing if only a fraction of candidate cuts are added.

Outline

Idea

Objectives

Given a set of candidate cuts, determine

- (P1) which cuts should be selected
- (P2) how many cuts should be selected
- (P3) in which order the selected cuts should be added

Done by learning!

Reinforcement learning

Markdown decision process

- State: (current LP, optimal solution, set of candidate cuts)
- Action: choose an order subset of candidate cuts (one cut selection)
- Reward:
  - dual bound improvement
  - final reward: (negative) solving time or (negative) primal dual integral gap
- Stops after T rounds of cutting plane.

State

Each constraint of the current LP is represented as a vector of size 13.

Table 7: The designed cut features of a generated cut $\boldsymbol{\alpha}^T \mathbf{x} \leq \beta$. (Suppose $\mathbf{c}$ denotes the objective coefficient.)

Feature	Description	Number
cut coefficients	the mean, max, min, std of cut coefficients	4
objective coefficients	the mean, max, min, std of the objective coefficients	4
parallelism	the parallelism between the objective and the cut $\frac{\mathbf{c}^T \boldsymbol{\alpha}}{\lvert\mathbf{c}\rvert \lvert\boldsymbol{\alpha}\rvert}$	1
efficacy	the Euclidean distance of the cut hyperplane to the current LP solution	1
support	the proportion of non-zero coefficients of the cut	1
integral support	the proportion of non-zero coefficients with respect to integer variables of the cut	1
normalized violation	the violation of the cut to the current LP solution $\max\{0, \frac{\boldsymbol{\alpha}^T \mathbf{x}^*_{LP} - \beta}{\lvert\beta\rvert}\}$	1

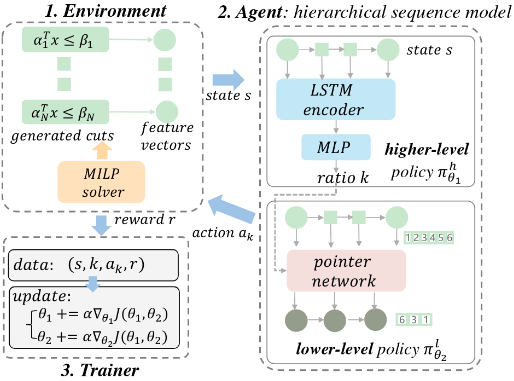
State

Sequence of such constraint vectors (order given by the order of the constraints in the current LP)

Hierarchical learning

- First model to define the ratio of candidate cuts to keep
- Second model to define select the candidate cuts (with order)

ML Model



1. Environment
$\alpha_1^T x \le \beta_1$
$\alpha_N^T x \le \beta_N$
generated cuts
feature vectors
MILP solver
state s
2. Agent: hierarchical sequence model
state s
LSTM encoder
MLP
higher-level policy $\pi_{\theta_1}^h$
ratio k
1 2 3 4 5 6
pointer network
6 3 1
lower-level policy $\pi_{\theta_2}^l$
action a_k
reward r
data: (s, k, a_k, r)
update:
$\theta_1 \mathrel{+}= \alpha \nabla_{\theta_1} J(\theta_1, \theta_2)$
$\theta_2 \mathrel{+}= \alpha \nabla_{\theta_2} J(\theta_1, \theta_2)$
3. Trainer

Comparison with scoring models

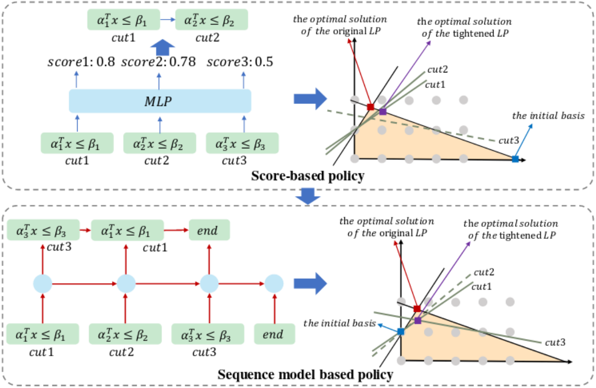


Figure 4: Illustration of selecting cuts using a sequence to sequence model compared to using a scoring function. The sequence model has two main advantages. First, it captures the interaction among cuts by selecting cuts one by one. Consequently, it selects cut3 and cut1 that complement each other nicely, leading to more tightened LP relaxation. Second, it naturally captures the order of selected cuts. Better order of selected cuts may lead to a better initial basis, thus solving the LP relaxation faster (Li et al., 2022) (see Section 3).

Training

Algorithm 2 Pseudo code for training the HEM

Initialize Hierarchical sequence model $\pi_{[\theta_1,\theta_2]}$, MILP instances $\mathcal{D}$, training dataset $\mathcal{D}_{\text{train}}$, batch size N_b, training epochs N_e, policy learning rate α
for N_e epochs **do**
  Empty the training dataset $\mathcal{D}_{\text{train}}$
  for N_b steps **do**
    Randomly sample a MILP s_0 from $\mathcal{D}$
    Take action k and a_k at state s_0 with the policy π
    Receive reward r and add (s_0, k, a_k, r) to $\mathcal{D}_{\text{train}}$
  end for
  Compute hierarchical policy gradient using $\mathcal{D}_{\text{train}}$ as in proposition 1
  Update the parameters, $\theta_1 = \theta_1 + \alpha\nabla_{\theta_1} J([\theta_1, \theta_2])$, $\theta_2 = \theta_2 + \alpha\nabla_{\theta_2} J([\theta_1, \theta_2])$
end for

Proposition 1. *Given the cut selection policy* $\pi_\theta(a_k|s) = \mathbb{E}_{k\sim\pi^h_{\theta_1}(\cdot|s)}[\pi^l_{\theta_2}(a_k|s,k)]$ *and the training objective (3), the hierarchical policy gradient takes the form of*

$$\nabla_{\theta_1} J([\theta_1,\theta_2]) = \mathbb{E}_{s\sim\mu,k\sim\pi^h_{\theta_1}(\cdot|s)}[\nabla_{\theta_1}\log(\pi^h_{\theta_1}(k|s))\mathbb{E}_{a_k\sim\pi^l_{\theta 2}(\cdot|s,k)}[r(s,a_k)]],$$

$$\nabla_{\theta_2} J([\theta_1,\theta_2]) = \mathbb{E}_{s\sim\mu,k\sim\pi^h_{\theta_1}(\cdot|s),a_k\sim\pi^l_{\theta_2}(\cdot|s,k)}[\nabla_{\theta_2}\log\pi^l_{\theta_2}(a_k|s,k)r(s,a_k)].$$

Outline

Datasets

- Easy datasets:
  - Set covering
  - Max independant set
  - Multiple knapsack
- Medium datasets:
  - MIK
  - CORLAT
- Hard datasets:
  - Load balancing problem
  - Anonymous problem
  - Two subsets of MIPLIB 2017

Competitors

Comparison with the following algorithms

- SCIP with no cuts
- SCIP default
- SCIP with random selection of 20% of candidate cuts
- Score-based policy (SBP): Implementation of a (slightly difference) learned scrore function (but original work was for one cut selection).

All algorithms use SCIP 8.0 (even theirs). They perform only one round of cutting at root node.

	Easy: Set Covering ($n = 1000,\ m = 500$)			Easy: Maximum Independent Set ($n = 500,\ m = 1953$)			Easy: Multiple Knapsack ($n = 720,\ m = 72$)		
Method	Time(s) $\downarrow$	Improvement (Time, %) $\uparrow$	PD integral $\downarrow$	Time(s) $\downarrow$	Improvement (Time, %) $\uparrow$	PD integral $\downarrow$	Time(s) $\downarrow$	Improvement (Time, %) $\uparrow$	PD integral $\downarrow$
NoCuts	6.31 (4.61)	NA	56.99 (38.89)	8.78 (6.66)	NA	71.31 (51.74)	9.88 (22.24)	NA	16.41 (14.16)
Default	4.41 (5.12)	29.90	55.63 (42.21)	3.88 (5.04)	55.80	29.44 (35.27)	9.90 (22.24)	-0.20	16.46 (14.25)
Random	5.74 (5.19)	8.90	67.08 (46.58)	6.50 (7.09)	26.00	52.46 (53.10)	13.10 (35.51)	-32.60	20.00 (25.14)
NV	9.86 (5.43)	-56.50	99.77 (53.12)	7.84 (5.54)	10.70	61.60 (43.95)	13.04 (36.91)	-32.00	21.75 (24.71)
Eff	9.65 (5.45)	-53.20	95.66 (51.71)	7.80 (5.11)	11.10	61.04 (41.88)	9.99 (19.02)	-1.10	20.49 (22.11)
SBP	1.91 (0.36)	69.60	38.96 (8.66)	2.43 (5.55)	72.30	21.99 (40.86)	7.74 (12.36)	21.60	16.45 (16.62)
HEM (Ours)	**1.85 (0.31)**	**70.60**	**37.92 (8.46)**	**1.76 (3.69)**	**80.00**	**16.01 (26.21)**	**6.13 (9.61)**	**38.00**	**13.63 (9.63)**

	Medium: MIK ($n = 413,\ m = 346$)			Medium: Corlat ($n = 466,\ m = 486$)			Hard: Load Balancing ($n = 61000,\ m = 64304$)		
Method	Time(s) $\downarrow$	PD integral $\downarrow$	Improvement $\uparrow$ (PD integral, %)	Time(s) $\downarrow$	PD integral $\downarrow$	Improvement $\uparrow$ (PD integral, %)	Time(s) $\downarrow$	PD integral $\downarrow$	Improvement $\uparrow$ (PD integral, %)
NoCuts	300.01 (0.009)	2355.87 (996.08)	NA	103.30 (128.14)	2818.40 (5908.31)	NA	300.00 (0.12)	14853.77 (951.42)	NA
Default	179.62 (122.36)	844.40 (924.30)	64.10	75.20 (120.30)	2412.09 (5892.88)	14.40	300.00 (0.06)	9589.19 (1012.95)	35.40
Random	289.86 (28.90)	2036.80 (933.17)	13.50	84.18 (124.34)	2501.98 (6031.43)	11.20	300.00 (0.09)	13621.20 (1162.02)	8.30
NV	299.76 (1.32)	2542.67 (529.49)	-7.90	90.26 (128.33)	3075.70 (7029.55)	-9.10	300.00 (0.05)	13933.88 (971.10)	6.20
Eff	298.48 (5.84)	2416.57 (642.41)	-2.60	104.38 (131.61)	3155.03 (7039.99)	-11.90	300.00 (0.07)	13913.07 (969.95)	6.30
SBP	286.07 (41.81)	2053.30 (740.11)	12.80	70.41 (122.17)	2023.87 (5085.96)	28.20	300.00 (0.10)	12535.30 (741.43)	15.60
HEM (Ours)	**176.12 (125.18)**	**785.04 (790.38)**	**66.70**	**58.31 (110.51)**	**1079.99 (2653.14)**	**61.68**	**300.00 (0.04)**	**9496.42 (1018.35)**	**36.10**

	Hard: Anonymous ($n = 37881,\ m = 49603$)			Hard: MIPLIB mixed neos ($n = 6958,\ m = 5660$)			Hard: MIPLIB mixed supportcase ($n = 19766,\ m = 19910$)		
Method	Time(s) $\downarrow$	PD integral $\downarrow$	Improvement $\uparrow$ (PD integral, %)	Time(s) $\downarrow$	PD integral $\downarrow$	Improvement $\uparrow$ (PD integral, %)	Time(s) $\downarrow$	PD integral $\downarrow$	Improvement $\uparrow$ (PD integral, %)
NoCuts	246.22 (94.90)	18297.30 (9769.42)	NA	253.65 (80.29)	14652.29 (12523.37)	NA	170.00 (131.60)	9927.96 (11334.07)	NA
Default	244.02 (97.72)	17407.01 (9736.19)	4.90	256.58 (76.05)	14444.05 (12347.09)	1.42	164.61 (135.82)	9672.34 (10668.24)	2.57
Random	243.49 (98.21)	16850.89 (10227.87)	7.80	255.88 (76.65)	14006.48 (12698.76)	4.41	165.88 (134.40)	10034.70 (11052.73)	-1.07
NV	242.01 (98.68)	16873.66 (9711.16)	7.80	263.81 (64.10)	14379.05 (12306.35)	1.86	**161.67 (131.43)**	8967.00 (9690.30)	9.68
Eff	244.94 (93.47)	17137.87 (9456.34)	6.30	260.53 (68.54)	14021.74 (12859.41)	4.30	167.35 (134.99)	9941.55 (10943.48)	-0.14
SBP	245.71 (92.46)	18188.63 (9651.85)	0.59	256.48 (78.59)	13531.00 (12898.22)	7.65	165.61 (135.25)	7408.65 (7903.47)	25.37
HEM (Ours)	**241.68 (97.23)**	**16077.15 (9108.21)**	**12.10**	**248.66 (89.46)**	**8678.76 (12337.00)**	**40.77**	162.96 (138.21)	**6874.80 (6729.97)**	**30.75**

Experiment 2

Compare HEM with just the inner model (pointer network with end token)

	Easy: Maximum Independent Set ($n = 500,\ m = 1953$)			Medium: Corlat ($n = 466,\ m = 486$)			Hard: MIPLIB mixed neos ($n = 6958,\ m = 5660$)		
Method	Time(s) ↓	Improvement ↑ (Time, %)	PD integral ↓	Time(s) ↓	PD integral ↓	Improvement ↑ (PD integral, %)	Time(s) ↓	PD integral ↓	Improvement ↑ (PD integral, %)
NoCuts	8.78 (6.66)	NA	71.31 (51.74)	103.30 (128.14)	2818.40 (5908.31)	NA	253.65 (80.29)	14652.29 (12523.37)	NA
Default	3.88 (5.04)	55.81	29.44 (35.27)	75.20 (120.30)	2412.09 (5892.88)	14.42	256.58 (76.05)	14444.05 (12347.09)	1.42
SBP	2.43 (5.55)	72.32	21.99 (40.86)	70.41 (122.17)	2023.87 (5085.96)	28.19	256.48 (78.59)	13531.00 (12898.22)	7.65
HEM w/o H	1.88 (4.20)	78.59	16.70 (28.15)	63.14 (115.26)	1939.08 (5484.83)	31.20	249.21 (88.09)	13614.29 (12914.76)	7.08
HEM (Ours)	**1.76 (3.69)**	**79.95**	**16.01 (26.21)**	**58.31 (110.51)**	**1079.99 (2653.14)**	**61.68**	**248.66 (89.46)**	**8678.76 (12337.00)**	**40.77**

Note Hierarchie is important (predict first the ratio and then select cut)
Question Why not subdividing into 3 models (ratio, subset order)?

Experiment 3

- HEM-ratio: the ratio is fixed
- HEM-ratio-order: the selected cuts are added in the order they are generated (no order selection in cut selection).

	Easy: Maximum Independent Set ($n = 500,\ m = 1953$)			Medium: Corlat ($n = 466,\ m = 486$)			Hard: MIPLIB mixed neos ($n = 6958,\ m = 5660$)		
Method	Time(s) ↓	Improvement ↑ (Time, %)	PD integral ↓	Time(s) ↓	PD integral ↓	Improvement ↑ (PD integral, %)	Time(s) ↓	PD integral ↓	Improvement ↑ (PD integral, %)
NoCuts	8.78 (6.66)	NA	71.31 (51.74)	103.30 (128.14)	2818.40 (5908.31)	NA	253.65 (80.29)	14652.29 (12523.37)	NA
Default	3.88 (5.04)	55.81	29.44 (35.27)	75.20 (120.30)	2412.09 (5892.88)	14.42	256.58 (76.05)	14444.05 (12347.09)	1.42
SBP	2.43 (5.55)	72.32	21.99 (40.86)	70.41 (122.17)	2023.87 (5085.96)	28.19	256.48 (78.59)	13531.00 (12898.22)	7.65
HEM-ratio-order	2.30 (5.18)	73.80	21.19 (38.52)	70.94 (122.93)	1416.66 (3380.10)	49.74	245.99 (93.67)	14026.75 (12683.60)	4.27
HEM-ratio	2.26 (5.06)	74.26	20.82 (37.81)	67.27 (117.01)	1251.60 (2869.87)	55.59	**244.87 (95.56)**	13659.93 (12900.59)	6.77
HEM (Ours)	**1.76 (3.69)**	**79.95**	**16.01 (26.21)**	**58.31 (110.51)**	**1079.99 (2653.14)**	**61.68**	248.66 (89.46)	**8678.76 (12337.00)**	**40.77**

Note ratio and order are important

Experiment 4

Test on bigger instances

	Easy: Maximum Independent Set ($n = 500,\ m = 1953$)			Medium: Corlat ($n = 466,\ m = 486$)			Hard: MIPLIB mixed neos ($n = 6958,\ m = 5660$)		
Method	Time(s) ↓	Improvement ↑ (Time, %)	PD integral ↓	Time(s) ↓	PD integral ↓	Improvement ↑ (PD integral, %)	Time(s) ↓	PD integral ↓	Improvement ↑ (PD integral, %)
NoCuts	8.78 (6.66)	NA	71.31 (51.74)	103.30 (128.14)	2818.40 (5908.31)	NA	253.65 (80.29)	14652.29 (12523.37)	NA
Default	3.88 (5.04)	55.81	29.44 (35.27)	75.20 (120.30)	2412.09 (5892.88)	14.42	256.58 (76.05)	14444.05 (12347.09)	1.42
SBP	2.43 (5.55)	72.32	21.99 (40.86)	70.41 (122.17)	2023.87 (5085.96)	28.19	256.48 (78.59)	13531.00 (12898.22)	7.65
HEM-ratio-order	2.30 (5.18)	73.80	21.19 (38.52)	70.94 (122.93)	1416.66 (3380.10)	49.74	245.99 (93.67)	14026.75 (12683.60)	4.27
HEM-ratio	2.26 (5.06)	74.26	20.82 (37.81)	67.27 (117.01)	1251.60 (2869.87)	55.59	**244.87 (95.56)**	13659.93 (12900.59)	6.77
HEM (Ours)	**1.76 (3.69)**	**79.95**	**16.01 (26.21)**	**58.31 (110.51)**	**1079.99 (2653.14)**	**61.68**	248.66 (89.46)	**8678.76 (12337.00)**	**40.77**

Note HEM generalizes well