

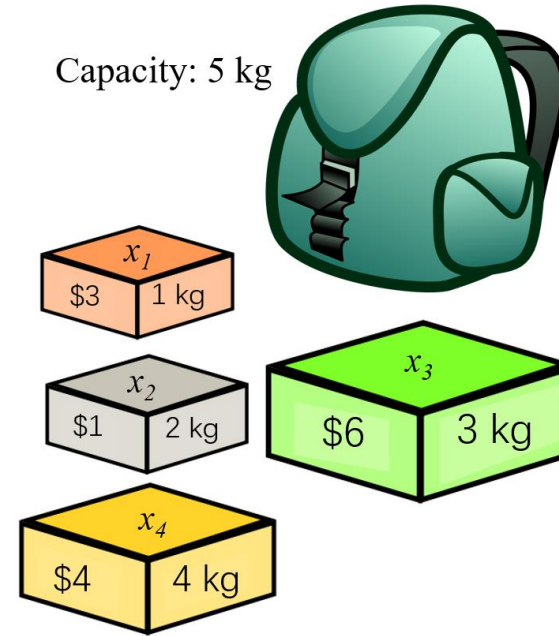
Lightweight Look-ahead-based Value Heuristics for Constraint Optimization Problems

Hongbo Li

Shanghai University of Finance and Economics

Constraint Optimization Problem

- A Constraint Optimization Problem (COP) P is a 4-tuple $P = \langle X, D, C, F \rangle$, where
 - X is a set of **variables**
 - D is a set of **domains**, where the domain $dom(x) \in D$ specifies the set of possible values for the variable $x \in X$
 - C is a set of **constraints**, where each constraint $c \in C$ specifies the allowed combinations of values for a subset of variables
 - F is an **objective function**, which maps every complete assignment of P to a real number.
- A complete assignment S that satisfies all constraints is a feasible solution to P .
- A feasible solution S^* is an optimal solution if $F(S^*) \leq F(S)$ for any other feasible solution S to P .



- $X: \{x_1, x_2, x_3, x_4\}$
- $D: dom(x_i) = \{0, 1\}$ for $i = 1$ to 4
- $C: x_1 + 2x_2 + 3x_3 + 4x_4 \leq 5$
- $F: \text{maximize } obj = 3x_1 + x_2 + 6x_3 + 4x_4$

Branch-and-Bound Search

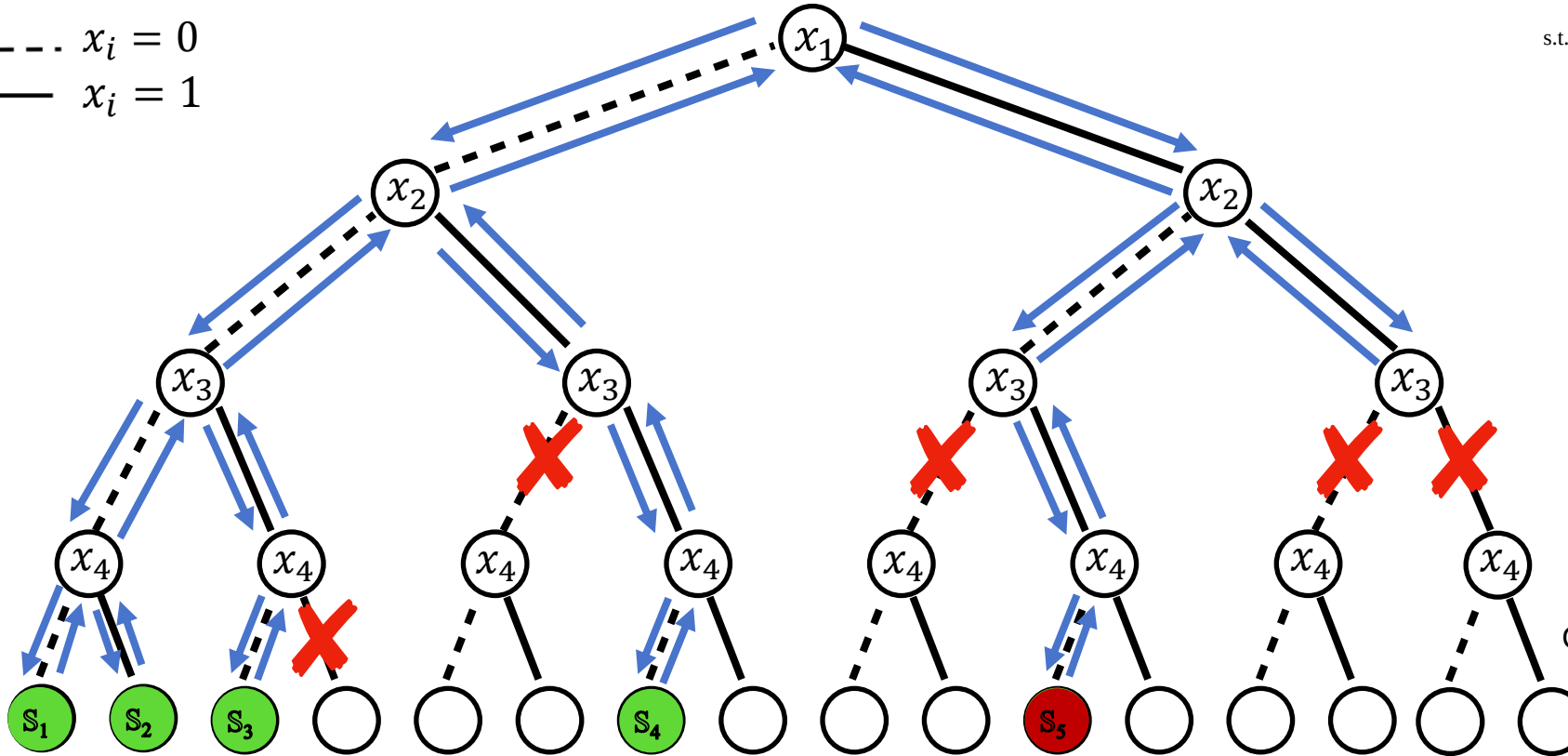
- Backtracking
 - Depth-first tree search
 - The main idea is to use the **current best feasible solution** found as an upper bound to force the next feasible solution must be **strictly better** than the current best one.
 - When no more solution can be found, optimality is proved and the **last feasible solution** found can be returned as the optimal solution of the COP.
- Search Heuristics
 - variable selection
 - value selection
 - variable-value selection
- Constraint Propagation
 - pruning search space

Branch-and-Bound Search

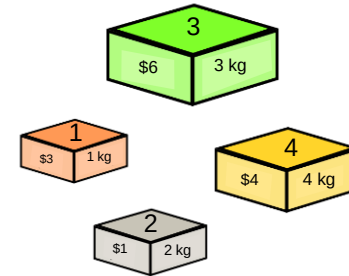
- - - - $x_i = 0$
 ——— $x_i = 1$

$$\text{s.t. } x_1 + 2x_2 + 3x_3 + 4x_4 \leq 5$$

$$x_i \in \{0,1\} \text{ for } i = 1, \dots, 4$$



Capacity: 5 kg



$$3x_1 + x_2 + 3x_3 + 4x_4 \geq 0$$

$$3x_1 + x_2 + 3x_3 + 4x_4 \geq 4$$

$$3x_1 + x_2 + 3x_3 + 4x_4 \geq 6$$

$$3x_1 + x_2 + 3x_3 + 4x_4 \geq 7$$

$$3x_1 + x_2 + 3x_3 + 4x_4 \geq 9$$

Optimal solution

$S_1 = \{0,0,0,0\}, \text{obj}=0$

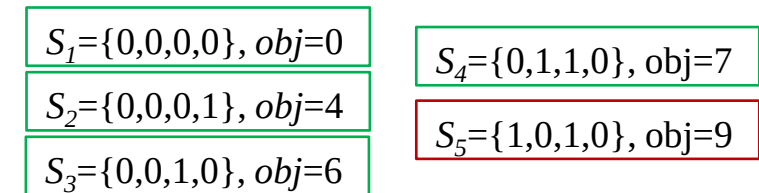
$S_2 = \{0,0,0,1\}, \text{obj}=4$

$S_3 = \{0,0,1,0\}, \text{obj}=6$

$S_4 = \{0,1,1,0\}, \text{obj}=7$

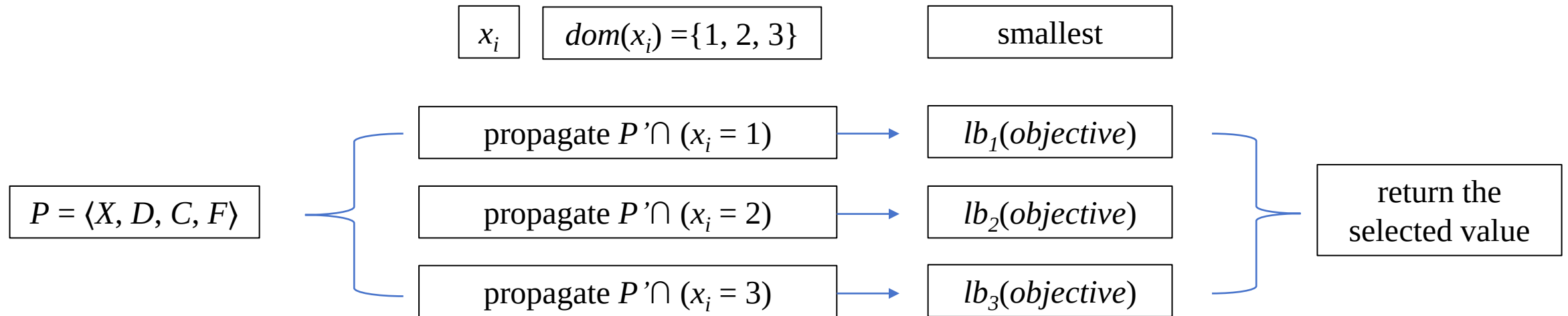
$S_5 = \{1,0,1,0\}, \text{obj}=9$

-
- 8 nodes to find an optimal solution
- $S_1 = \{1, 0, 0, 0\}, \text{obj} = 3$
- $S_2 = \{1, 0, 0, 1\}, \text{obj} = 7$
- $S_3 = \{1, 0, 1, 0\}, \text{obj} = 9$



- **Bound-Impact Value Selector (BIVS)**^[1]

- It **iterates over all values** in $dom(x)$ and evaluates each value by **examining its impact** on the objective variable.
- It **propagates** the instantiation of each value on a copy of the COP P' , and selects the value that **yields the best bounds**.
- The smallest lower bound in minimization problems
- The largest upper bound in maximization problems.

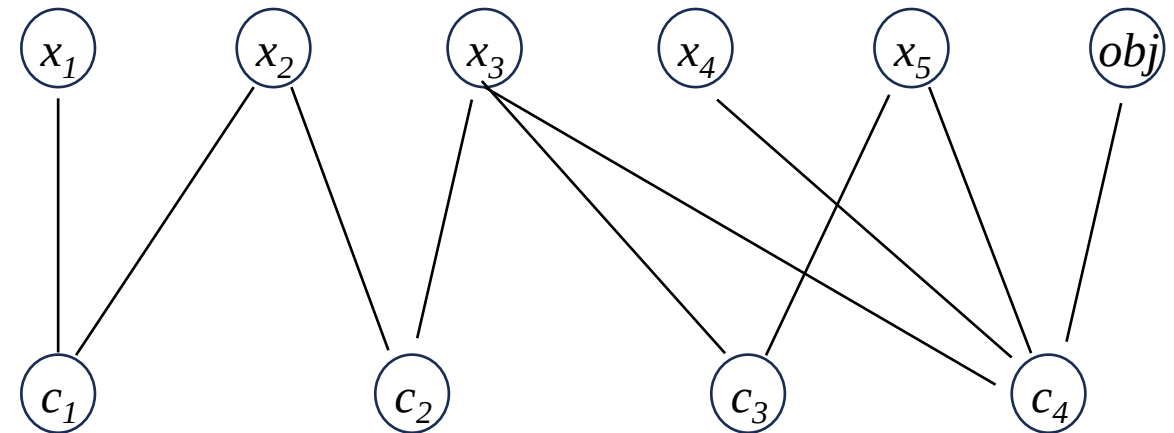


[1] J. G. Fages and C. Prud' Homme. Making the first solution good. In Proc. ICTAI' 17. IEEE, 2017.

- **Restricted Fixpoint**^[2]

- RF restricts propagation to a **subset** of C .
- It constructs a **bipartite variable-constraint graph**, where variables and constraints form two disjoint vertex sets, and an edge connects a variable to a constraint if the variable appears in that constraint.
- When evaluating the values of a variable x , RF only propagates the constraints **along the shortest paths** from x to the objective variable.

- When propagating x_1 , RF propagates through c_1, c_2, c_4 .
- When propagating x_3 , RF propagates through only c_4 .



[2] Augustin Delecluse and Pierre Schaus. Black-box value heuristics for solving optimization problems with constraint programming. In Proc. of CP'24, 2024.

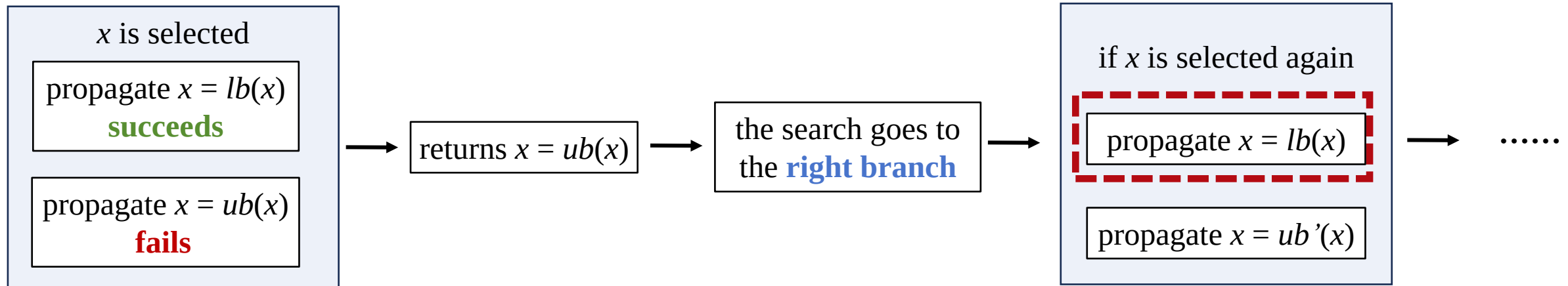
Access Before Look-Ahead

- **The issue:** the look-ahead process in BIVS or RF might be computationally **expensive**, making them less efficient to apply throughout the **entire** branch-and-bound search.
- **Access Before Look-Ahead**
 - **The motivation:** the look-ahead-based heuristics may fail to distinguish between candidate values in certain scenarios
 - **The key idea:** to first examine the impacts of the lower and upper bounds of the selected variable to determine whether a full look-ahead is worthwhile
 - **If the impacts of the two bounds are identical**, then the full look-ahead process is skipped, and the smallest value is returned
 - Otherwise, the look-ahead process is invoked to evaluate all values in the domain

Access Before Look-Ahead

• ABLA & ABLA2

- Propagating either the lower bound or the upper bound of x may lead to a failure, and **processing conflicts with priority** may benefit the search
- We always examine the lower bound first, and then the upper bound
- ABLA directly returns **the boundary value** where a conflict is detected during the assessment
- ABLA2 returns the value **only** when a conflict occurs on **the lower bound**; a conflict on the upper bound is treated as one of the propagation outcomes



- 173 COP instances from MiniZinc Challenges 2024-2025
- Choco solver 4.10.8
- Metrics: primal gap^[3]
- Luby^[4] restart with initial failure limit set to 500
- Nogood-Recording^[5]
- Baselines
 - Value heuristics: MV, BIVS^[1], RF^[2], Reverse Look-ahead + RF (RLARF)^[2]
 - Variable heuristics: activity-based search (ABS)^[6], dom/wdeg^[7], pick/dom^[8], FRBA4^[9]

[3] Timo Berthold. Measuring the impact of primal heuristics. *Operations Research Letters*, 41(6):611–614, 2013

[4] Luby, M.; Sinclair, A.; and Zuckerman, D.: Optimal speedup of Las Vegas algorithms. *Information Processing Letters*.

[5] Lecoutre, C.; Sais, L.; Tabary, S.; and Vidal, V.: Recording and Minimizing Nogoods from Restarts. *Journal on Satisfiability, Boolean Modeling and Computation*.

[6] Michel, L.; and Van Hentenryck, P.: Activity-based Search for Black-box Constraint Programming Solvers. *Proc. CPAIOR'12*.

[7] Boussemart, F.; Hemery, F.; Lecoutre, C.; and Sais, L.: Boosting Systematic Search by Weighting Constraints. *Proc. ECAI'04*.

[8] Gilles Audemard, Christophe Lecoutre, and Charles Prud'Homme. Guiding backtrack search by tracking variables during constraint propagation. In *Proc. of CP'23*, 2023.

[9] Y. Zhang and H. Li. Right branches matter in failure-based variable ordering heuristics. In *Proc. of AAAI'26*, 2026

Experiments

proportion of decisions requiring a look-ahead out of all decisions

proportion of decisions that ABLA decides to skip look-ahead out of all look-ahead decisions

The proportion of cases where the value selection decisions made with skipping look-ahead is the same as that of performing the look-ahead

When ABLA decides to skip, the proportion of cases where all values in the variable's domain have the same impact on the objective variable

	n_1	BIVS		RF	
		ABLA	ABLA2	ABLA	ABLA2
Proportion of Look-Ahead Decisions	$1 - 10^4$	98% (28)	99% (26)	99% (32)	99% (32)
	$10^4 - 10^5$	99% (18)	95% (22)	97% (19)	96% (17)
	$10^5 - 10^6$	86% (37)	90% (36)	85% (34)	85% (35)
	$10^6 - 10^7$	91% (59)	90% (61)	92% (50)	91% (53)
	$10^7 - \infty$	99% (29)	99% (26)	99% (26)	99% (24)
	Average	93% (171)	94% (171)	94% (161)	93% (161)
Proportion of Skipping Decisions	$1 - 10^4$	6% (28)	6% (26)	28% (32)	30% (32)
	$10^4 - 10^5$	10% (18)	8% (22)	40% (19)	35% (17)
	$10^5 - 10^6$	17% (37)	13% (36)	36% (34)	40% (35)
	$10^6 - 10^7$	13% (59)	13% (61)	48% (50)	49% (53)
	$10^7 - \infty$	14% (29)	13% (26)	58% (26)	54% (24)
	Average	12% (171)	11% (171)	42% (161)	43% (161)
Proportion of Decision Agreement When Skipping	$1 - 10^4$	89% (28)	91% (26)	92% (32)	95% (32)
	$10^4 - 10^5$	97% (18)	97% (22)	99% (19)	93% (17)
	$10^5 - 10^6$	98% (37)	98% (36)	92% (34)	92% (35)
	$10^6 - 10^7$	98% (59)	98% (61)	95% (50)	94% (53)
	$10^7 - \infty$	97% (29)	95% (26)	98% (26)	98% (24)
	Average	97% (171)	97% (171)	95% (161)	94% (161)
Proportion of Uniform Impact When Skipping	$1 - 10^4$	85% (28)	86% (26)	87% (32)	86% (32)
	$10^4 - 10^5$	88% (18)	88% (22)	85% (19)	84% (17)
	$10^5 - 10^6$	81% (37)	83% (36)	82% (34)	83% (35)
	$10^6 - 10^7$	88% (59)	88% (61)	91% (50)	90% (53)
	$10^7 - \infty$	81% (29)	80% (26)	92% (26)	93% (24)
	Average	85% (171)	85% (171)	88% (161)	88% (161)

Experiments

Time	Metrics	Heuristics	BIVS			RF			MV	RLARF
			ABLA	ABLA2	BIVS	ABLA	ABLA2	RF		
1800s	Primal Gap	Pick	20.89	20.52	24.02	22.22	22.17	25.47	30.38	29.91
		FRBA4	21.51	20.75	24.57	23.38	23.45	26.97	30.27	28.63
	#Solved	Pick	156	157	155	157	157	157	156	156
		FRBA4	155	156	155	154	154	154	154	155
	#Optimum	Pick	51	54	47	52	51	52	55	53
		FRBA4	54	58	50	57	57	56	57	56
3600s	Primal Gap	Pick	18.33	16.91	18.29	18.45	19.48	23.49	27.35	26.27
		FRBA4	17.55	17.79	20.42	20.92	21.00	25.19	27.10	26.17
	#Solved	Pick	157	159	159	159	158	158	157	158
		FRBA4	158	158	156	155	155	155	156	156
	#Optimum	Pick	56	56	53	55	58	56	58	55
		FRBA4	55	61	57	63	66	61	60	60

Conclusion

- There are about **10%** value selection decisions that obtain a skipping decision
- The skipping decisions are **highly reliable**
- ABLA2 performs better than ABLA in finding better solutions in general, as well as the baselines

Q&A

Thank You

Hongbo Li
lihongbo@mail.sufe.edu.cn