南京大学 人工智能学院
SCHOOL OF ARTIFICIAL INTELLIGENCE, NANJING UNIVERSITY

LAMDA Learning And Mining from DatA
http://www.lamda.nju.edu.cn

Macro Placement by Wire-Mask-Guided Black-Box Optimization

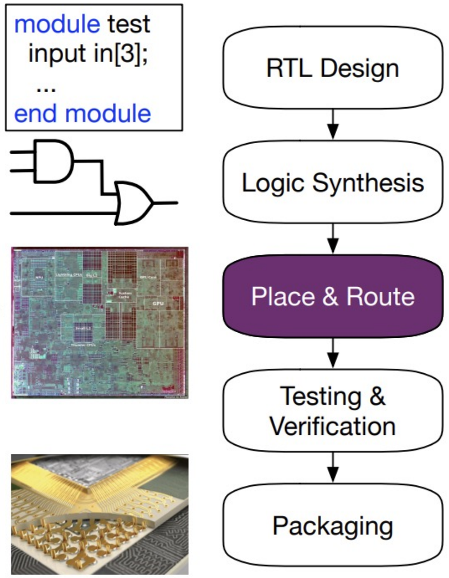
Yunqi Shi, Ke Xue, Lei Song, and **Chao Qian***

School of Artificial Intelligence
Nanjing University, China

Advances in Neural Information Processing Systems 36 (NeurIPS'23)

Why we are the best?

The problem we solve



Macro Placement

A vital stage in chip design

The baselines we beat

Google nature

ICML International Conference On Machine Learning

NEURAL INFORMATION PROCESSING SYSTEMS

TCAD, DAC **best paper**

human experts

Significant improvement

(e.g., 80% wirelength improvement over [Google, Nature'21])

The contributions to community

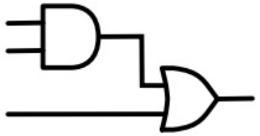
- **Bring EAs back to the state-of-the-art for macro placement**
- **Reaffirm the potential of EAs for chip design**

Chip design

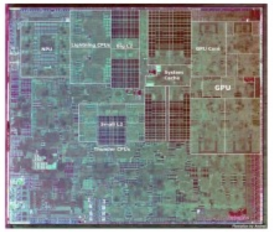
Register transfer level (RTL)

```
module test
 input in[3];
 ...
end module
```

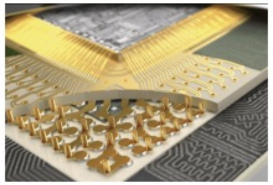
Netlist



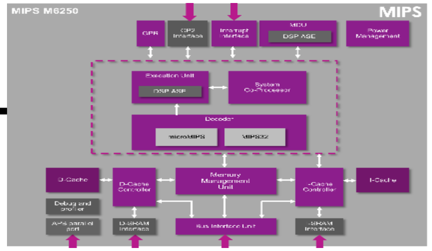
Chip layout



Wafer



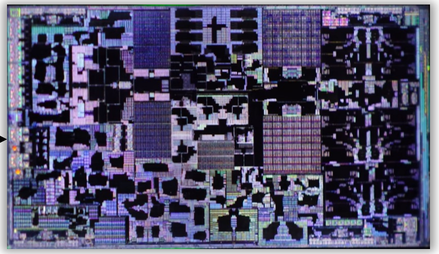
RTL Design → Logic Synthesis → Place & Route → Testing & Verification → Packaging



Design Document

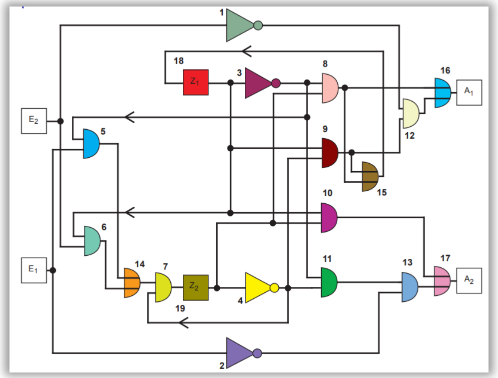
What we study

Macro placement: The first step in place & route, which significantly influences the subsequent stages

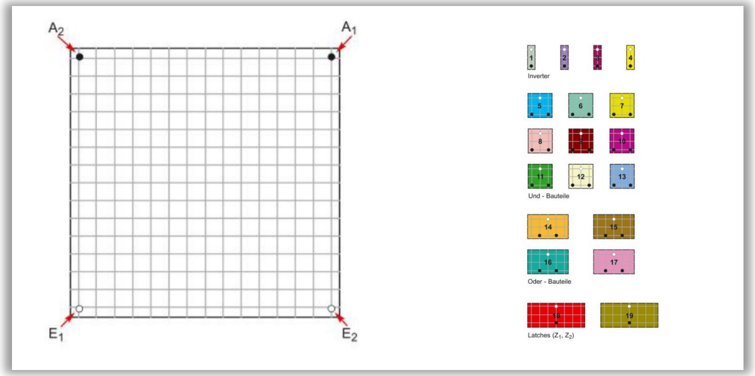


Apple A17 Pro

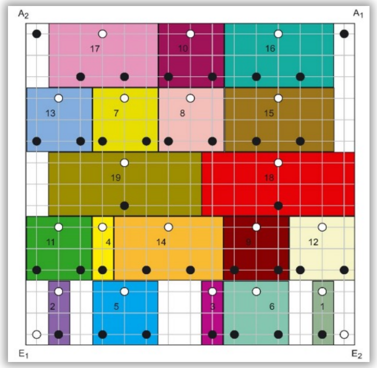
Macro placement



Netlist



Chip canvas To-place macros



Placement result

Performance:

Power

Timing

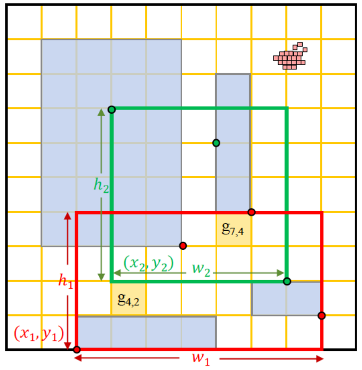
Congestion

Minimization objective: Half Perimeter Wire Length

$$\mathrm{HPWL}(\boldsymbol{s}, H) = \sum\nolimits_{e_j \in E} (w_j + h_j)$$

Sum of the half perimeter of nets' bounding boxes

Non-differentiable!



$$\mathrm{HPWL} = w_1 + h_1 + w_2 + h_2$$

Constraint:

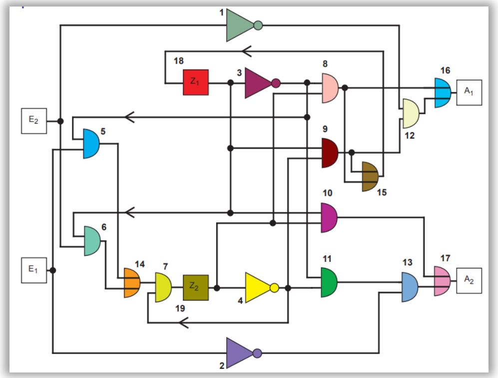
No-overlapping

High-dimensional:

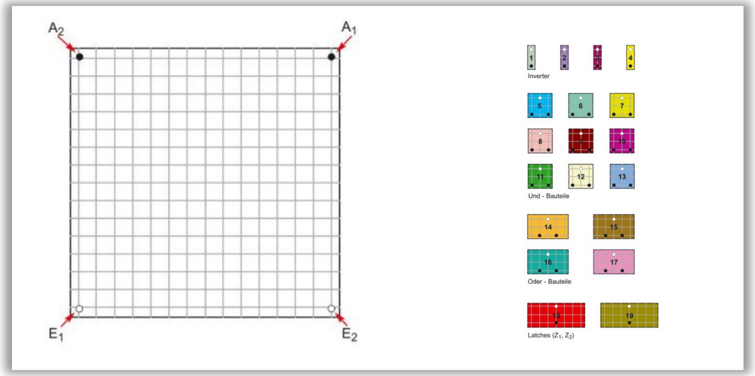
Thousands of macros

Some figures are from [Jens Vygen, Combinatorial Optimization and Applications in VLSI Design]

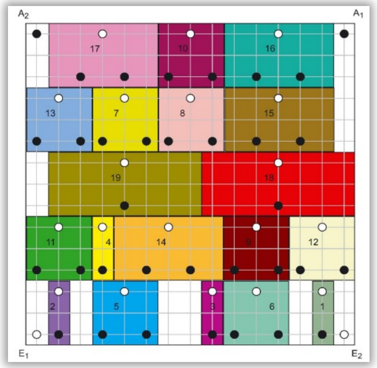
Macro placement



Netlist



Chip canvas To-place macros



Placement result

Performance:

Power

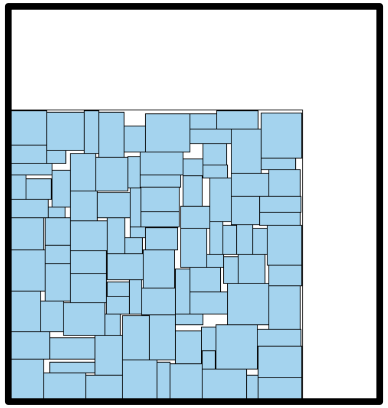
Timing

Congestion

Human experts spend weeks for macro placement, owing to the very large number of macros to be placed and their complex connections

Design efficient algorithms producing better placement than human experts

Previous methods



Classical EAs

1980s-2000s

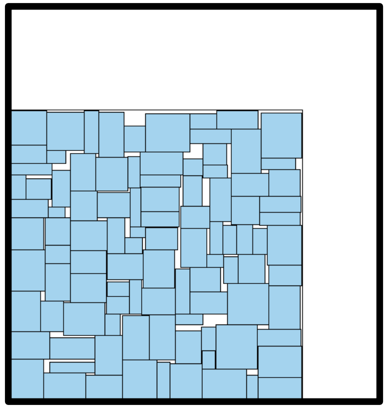
- Packing-based solution representation
- Optimize by Evolutionary Algorithms (EAs)

m: number of macros

- $O(m^2)$ complexity for mapping [Murata et al., TCAD'96]
- Final performance heavily relies on the initial placement
- Low-efficiency and low-scalability

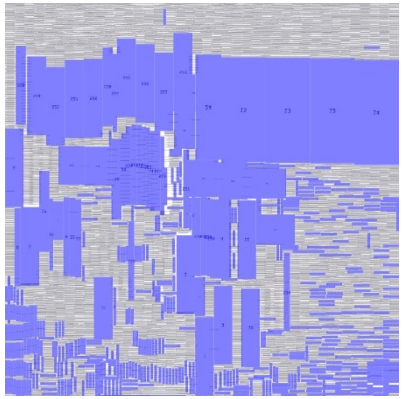
"… it is very slow and difficult to parallelize, thereby failing to scale to the increasingly large and complex circuits of the 1990s and beyond." [Google, Nature'21]

Previous methods



Classical EAs

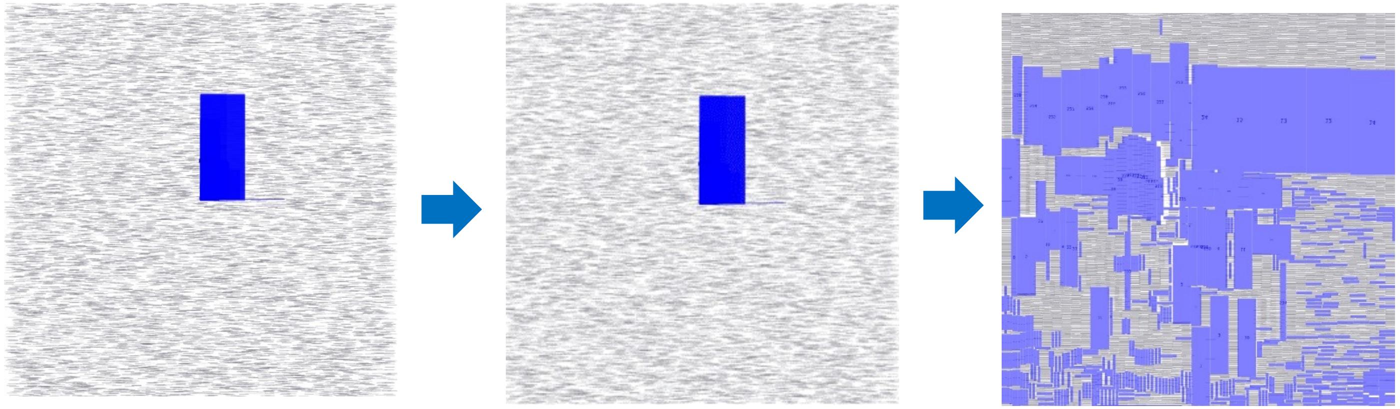
1980s-2000s



Analytical Placers

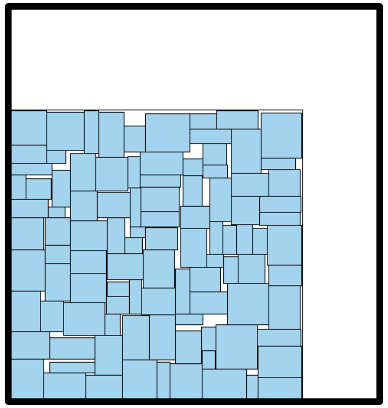
2000s-Now

- Smooth the HPWL formulation
- Optimize by gradient descent
- Overlapping
- Get stuck in local optima



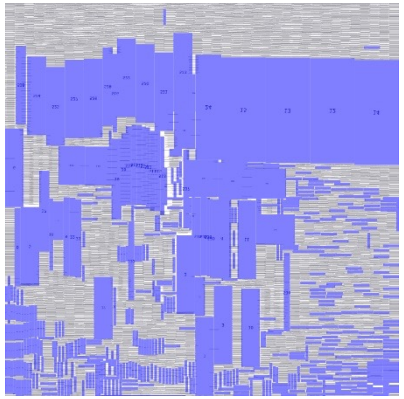
Some figures are from [Yibo Lin, DREAMPlace]

Previous methods



Classical EAs

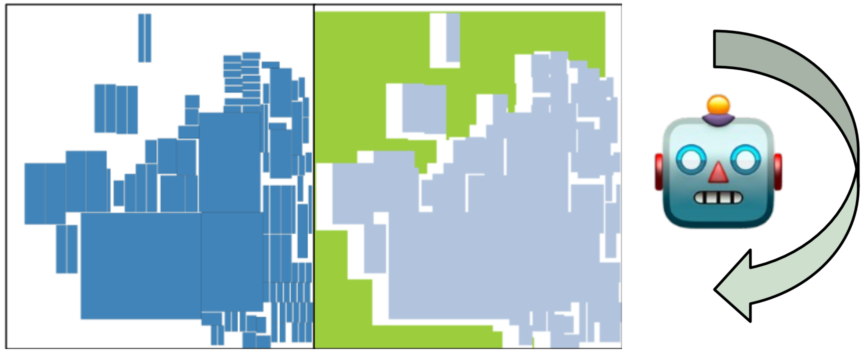
1980s-2000s



Analytical Placers

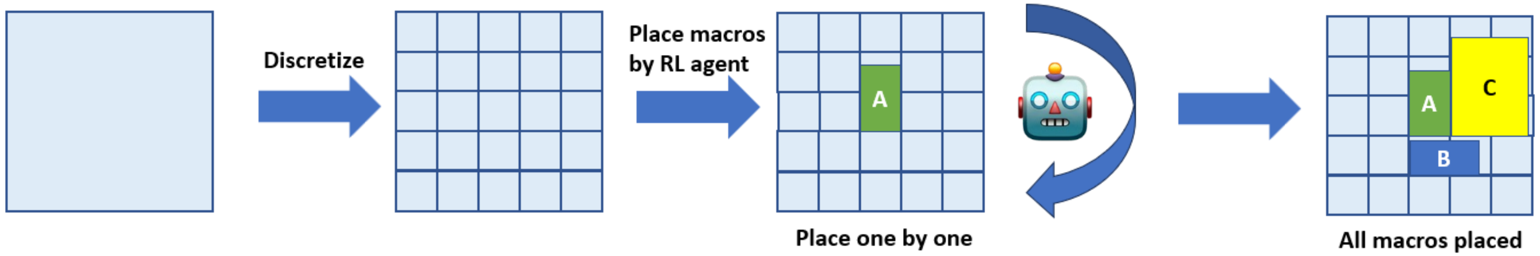
2000s-Now

[Google, Nature'21]



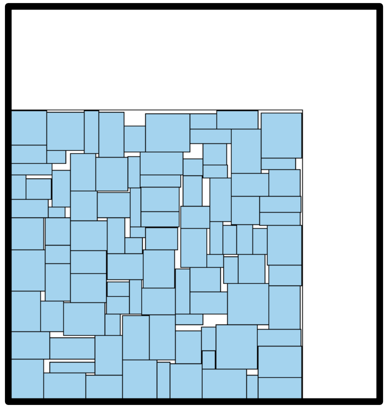
Reinforcement Learning Methods

2021-Now



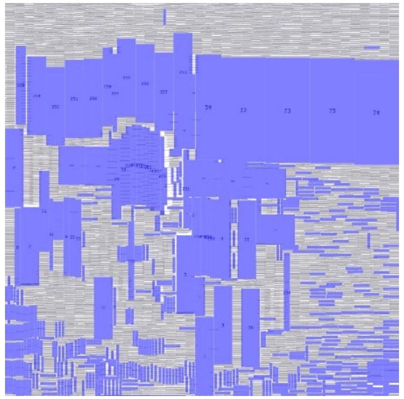
- Markov Decision Process
- Long training time
- Poor exploration

Previous methods



Classical EAs

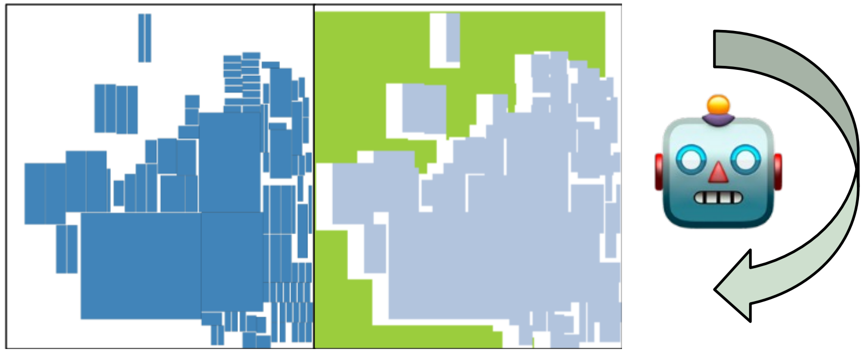
1980s-2000s



Analytical Placers

2000s-Now

[Google, Nature'21]



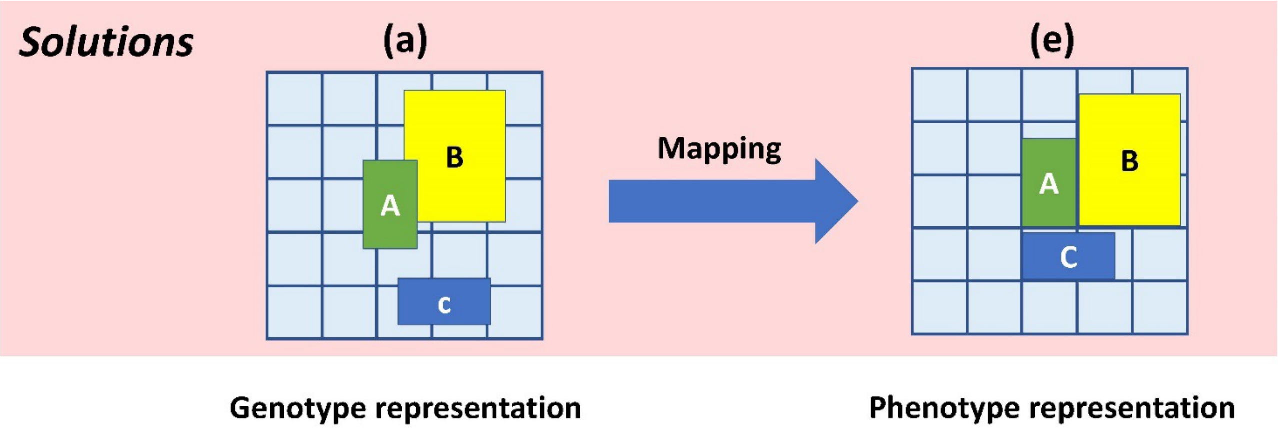
Reinforcement Learning Methods

2021-Now

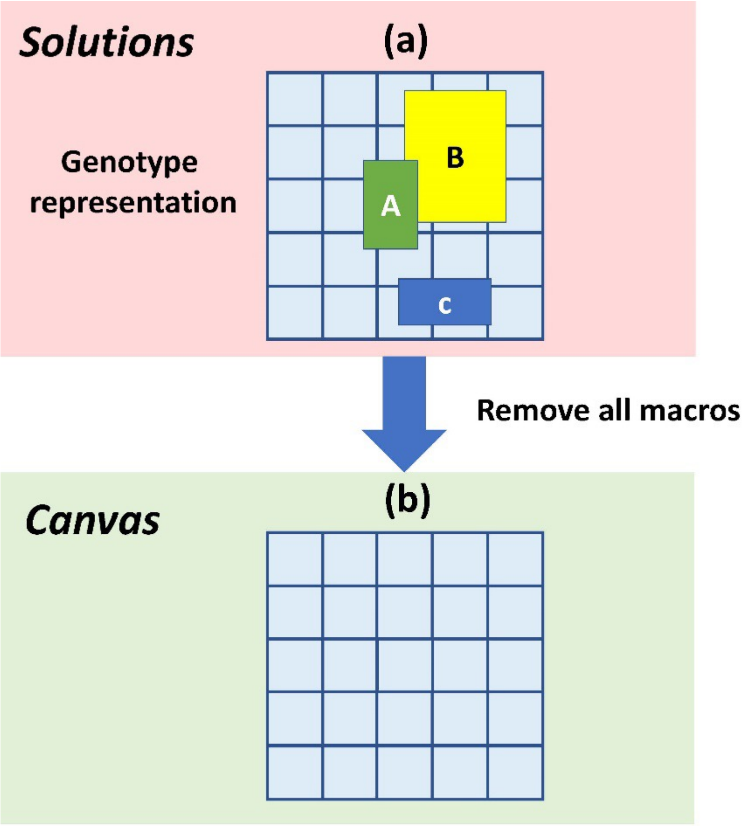
We propose a genotype-phenotype mapping inspired by RL formulation

Bring EAs back to the state-of-the-art!

Proposed genotype-phenotype mapping

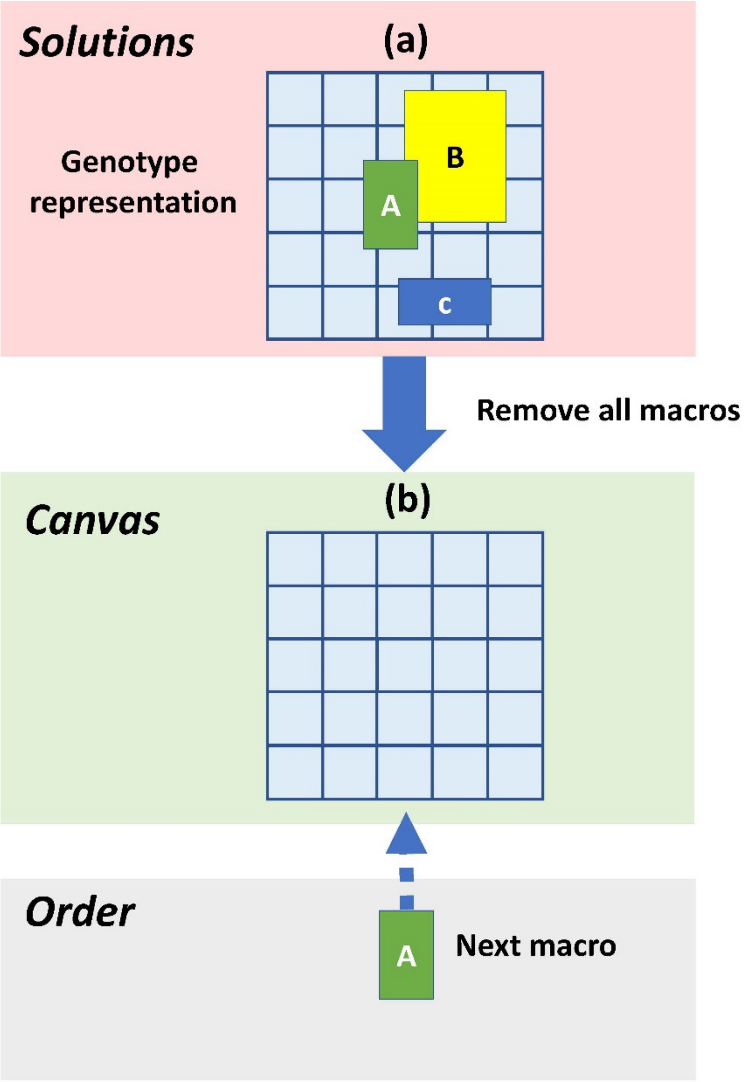


Proposed genotype-phenotype mapping



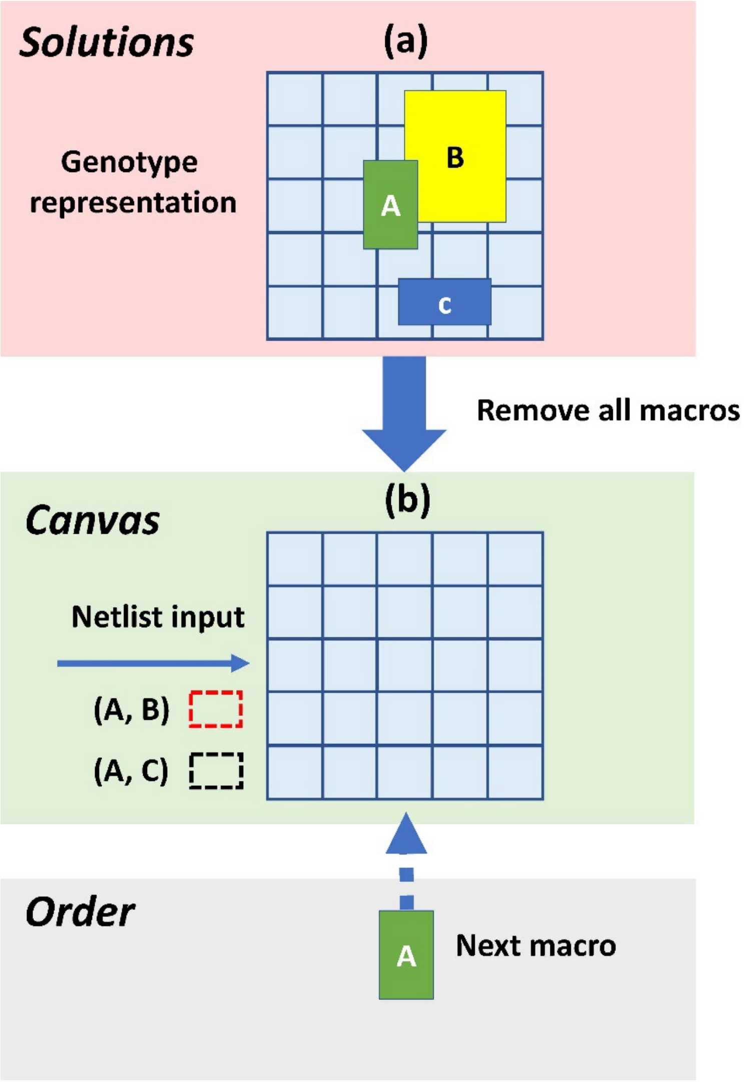
Partition the chip canvas into discrete grids

Proposed genotype-phenotype mapping



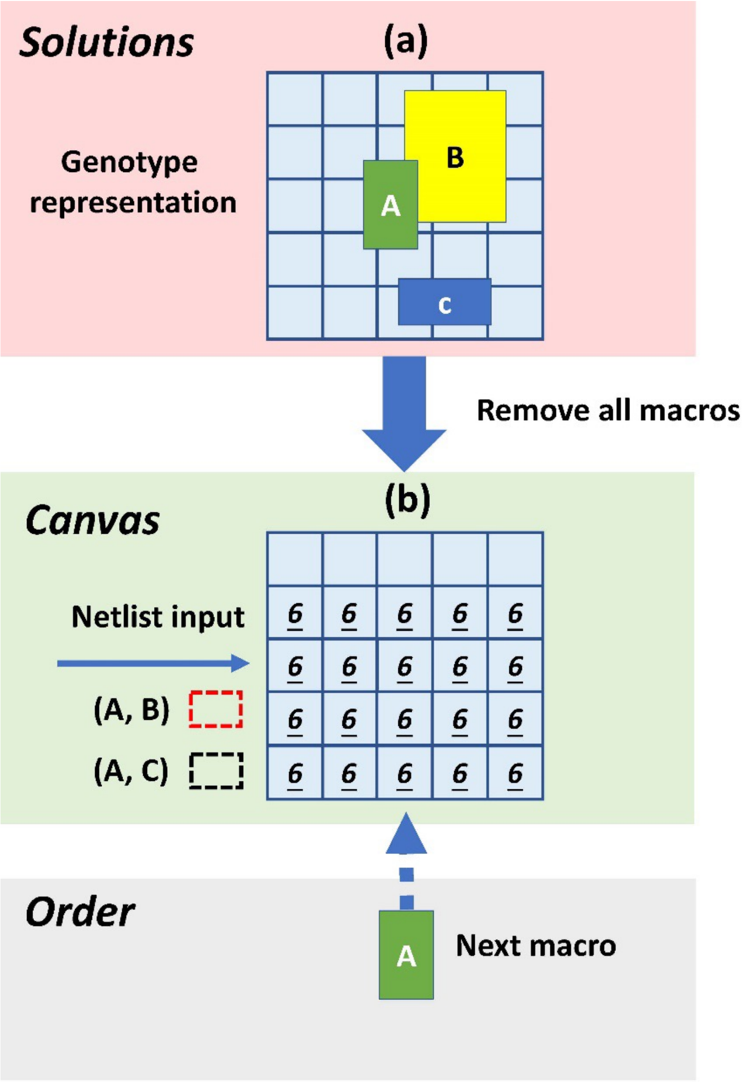
The first macro to place

Proposed genotype-phenotype mapping



The input netlist indicates connection relationships

Proposed genotype-phenotype mapping



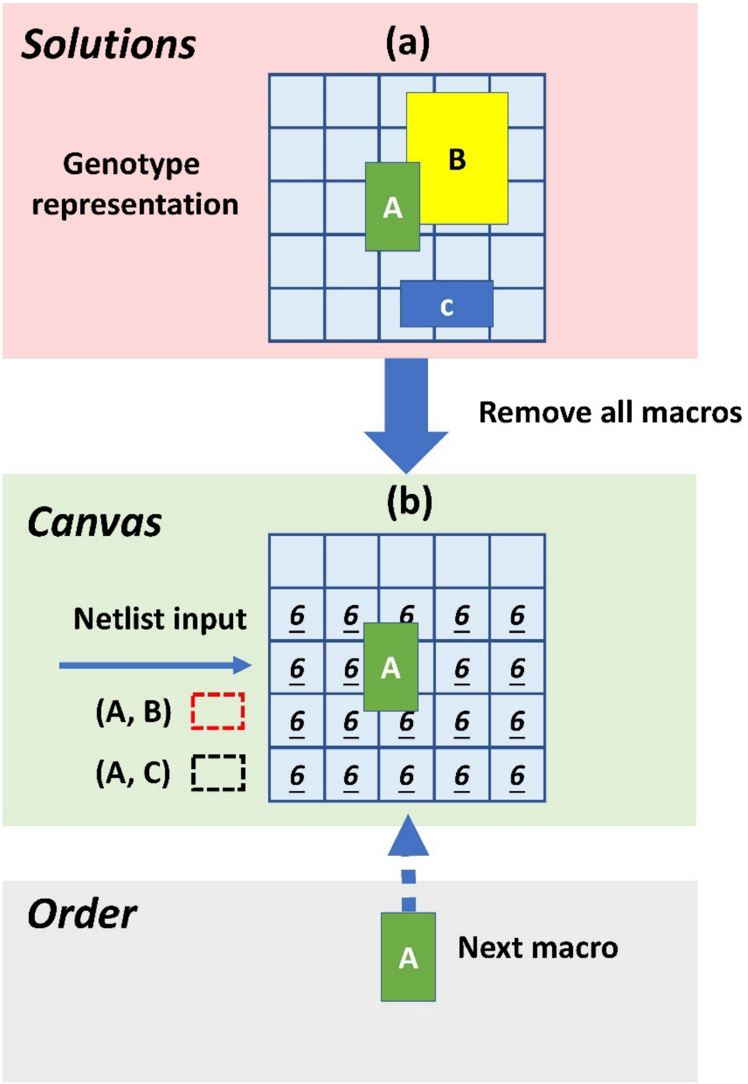
Calculate the wirelength increment after placing the macro:

Invalid location resulting in overlap or exceeding the boundary

6 Wirelength increment if placing at this grid

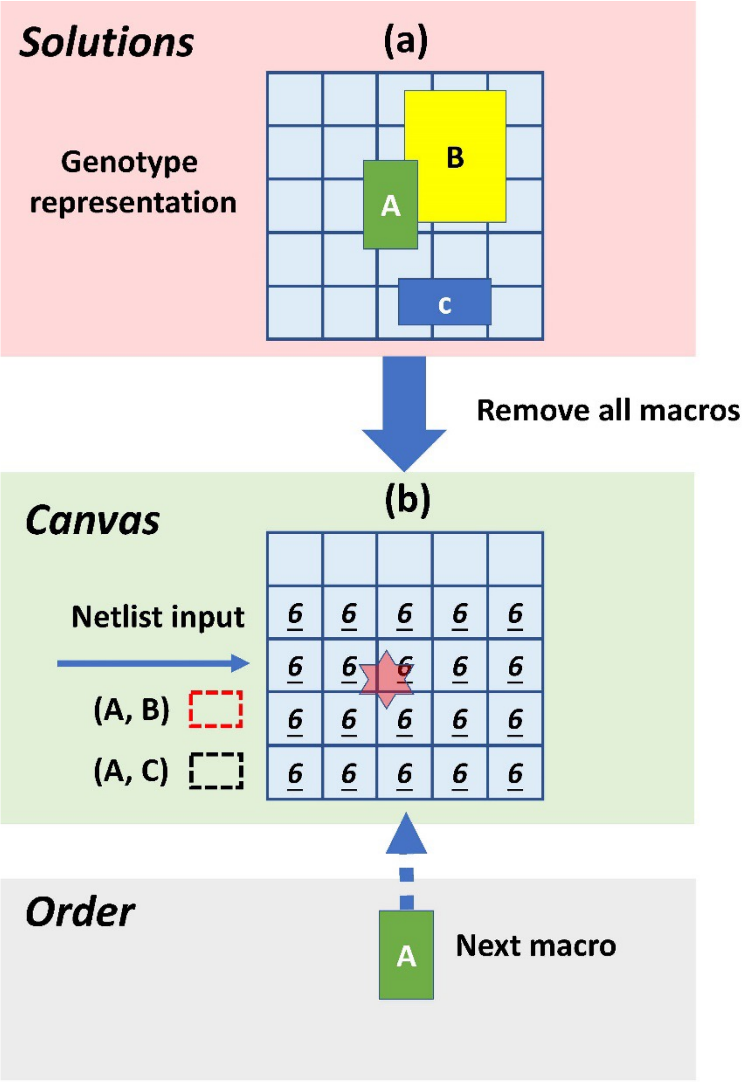
6 Underlined number means the minimum increment

Proposed genotype-phenotype mapping



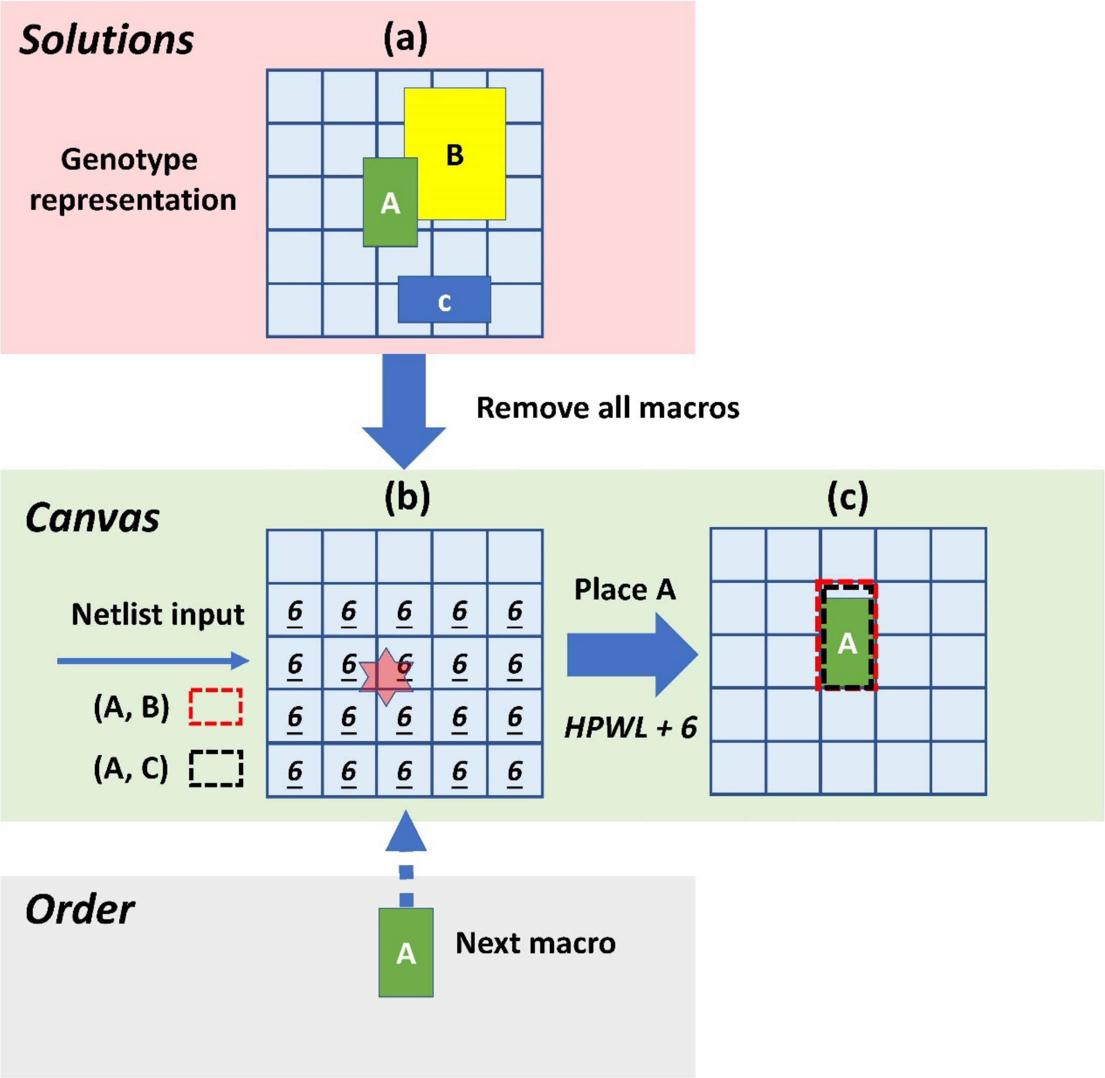
Refer to its original location

Proposed genotype-phenotype mapping



Refer to its original location

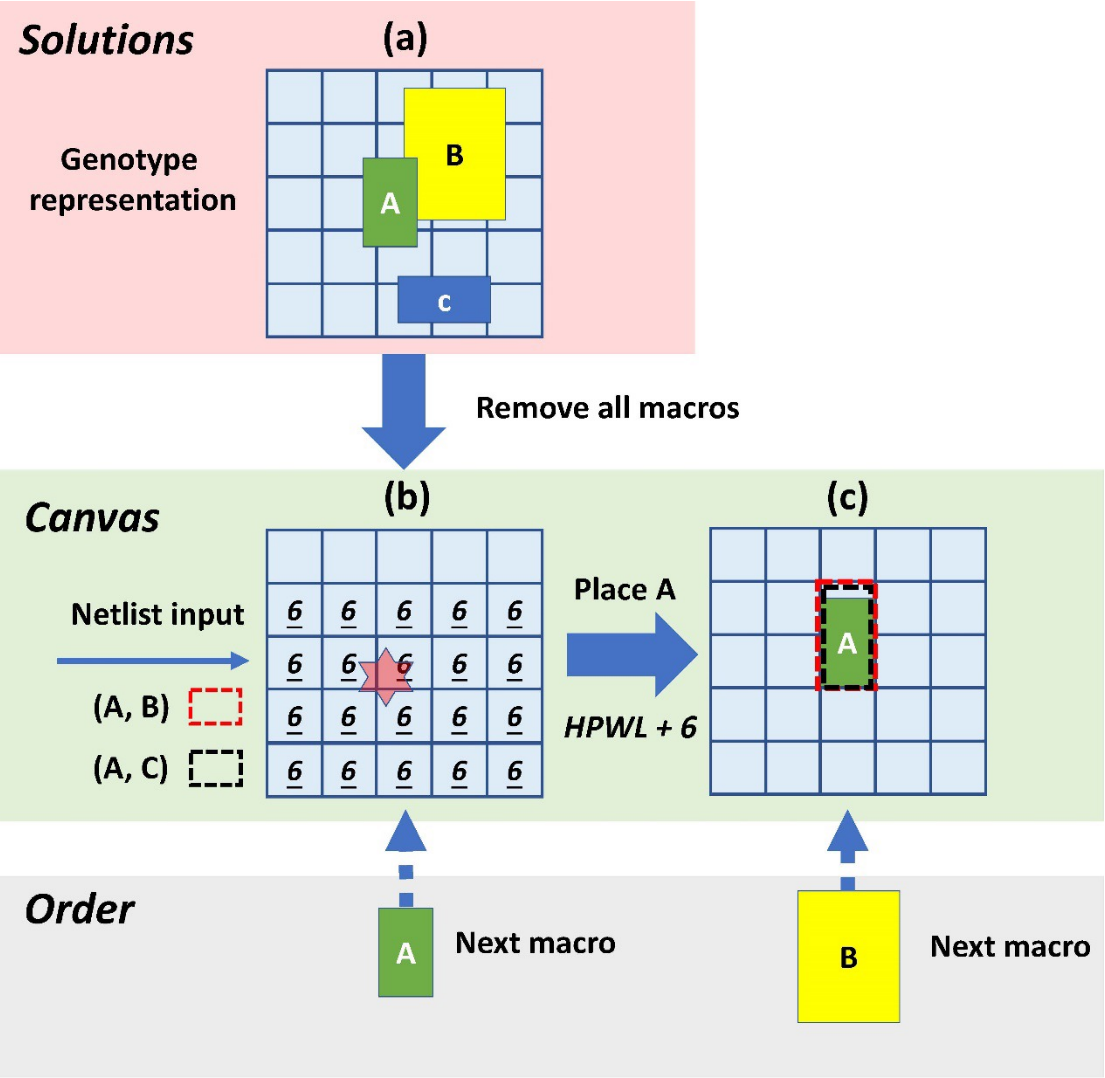
Proposed genotype-phenotype mapping



Choose the nearest best grid

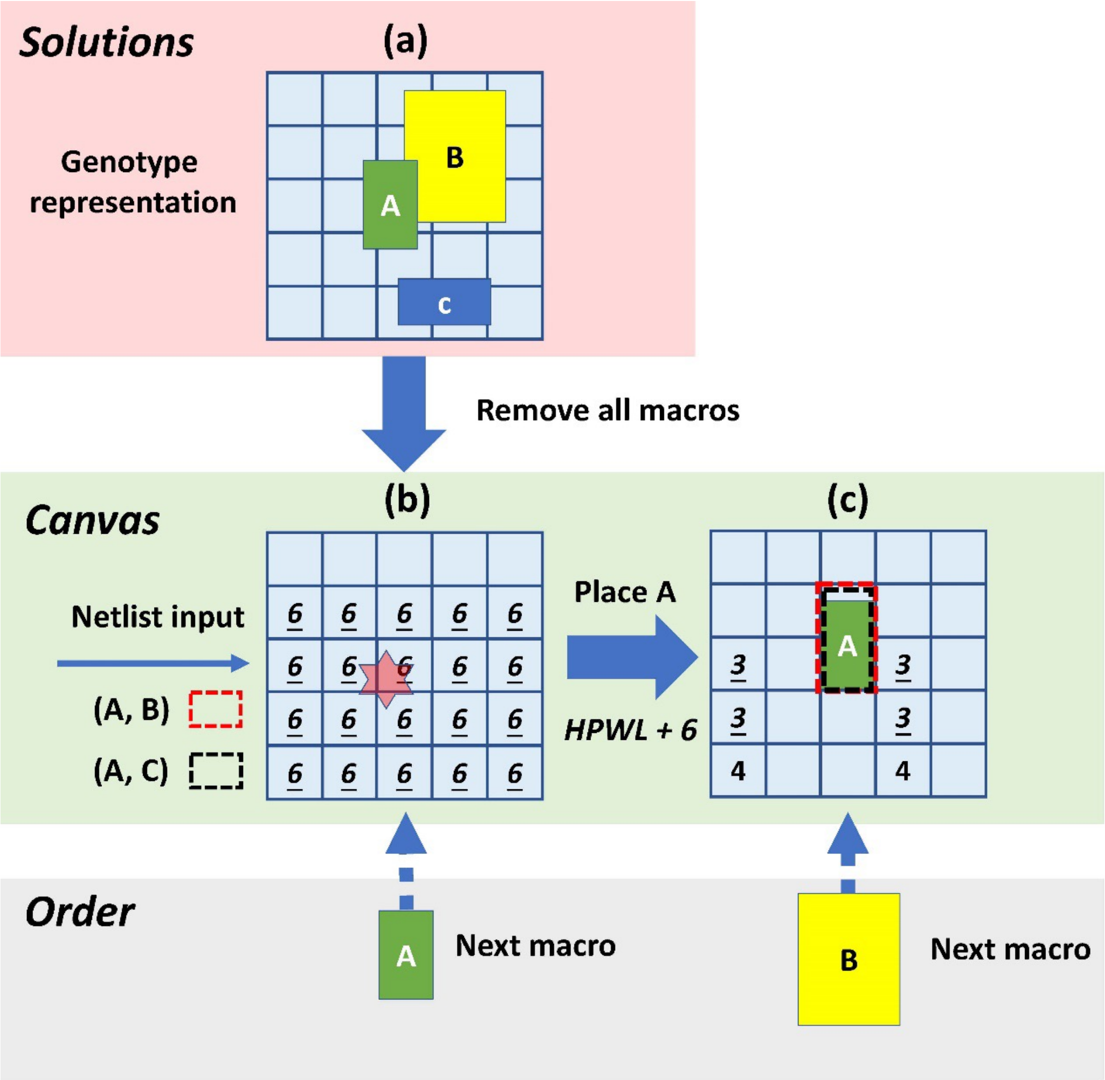
Place the macro

Proposed genotype-phenotype mapping



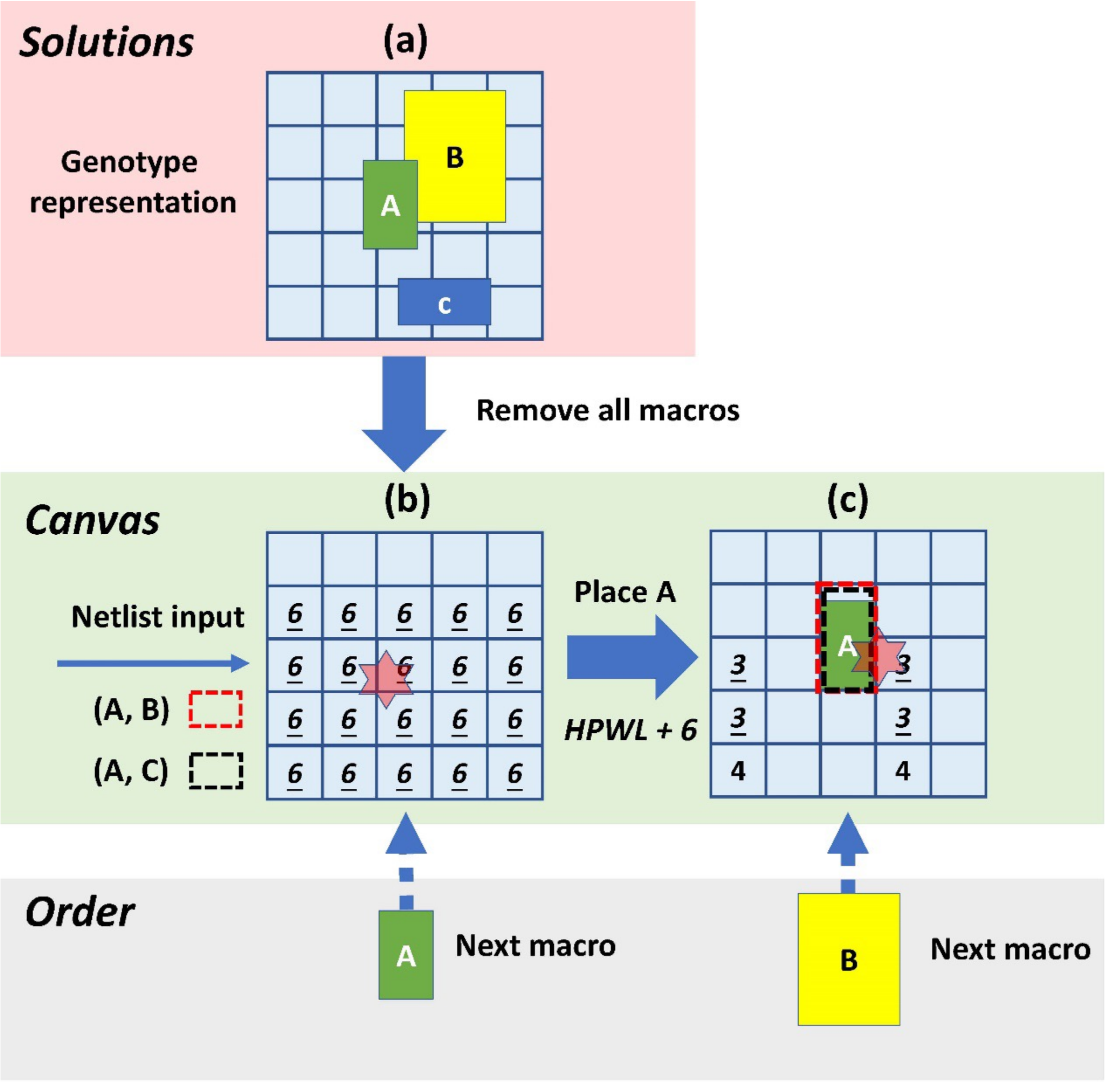
The second macro to place

Proposed genotype-phenotype mapping



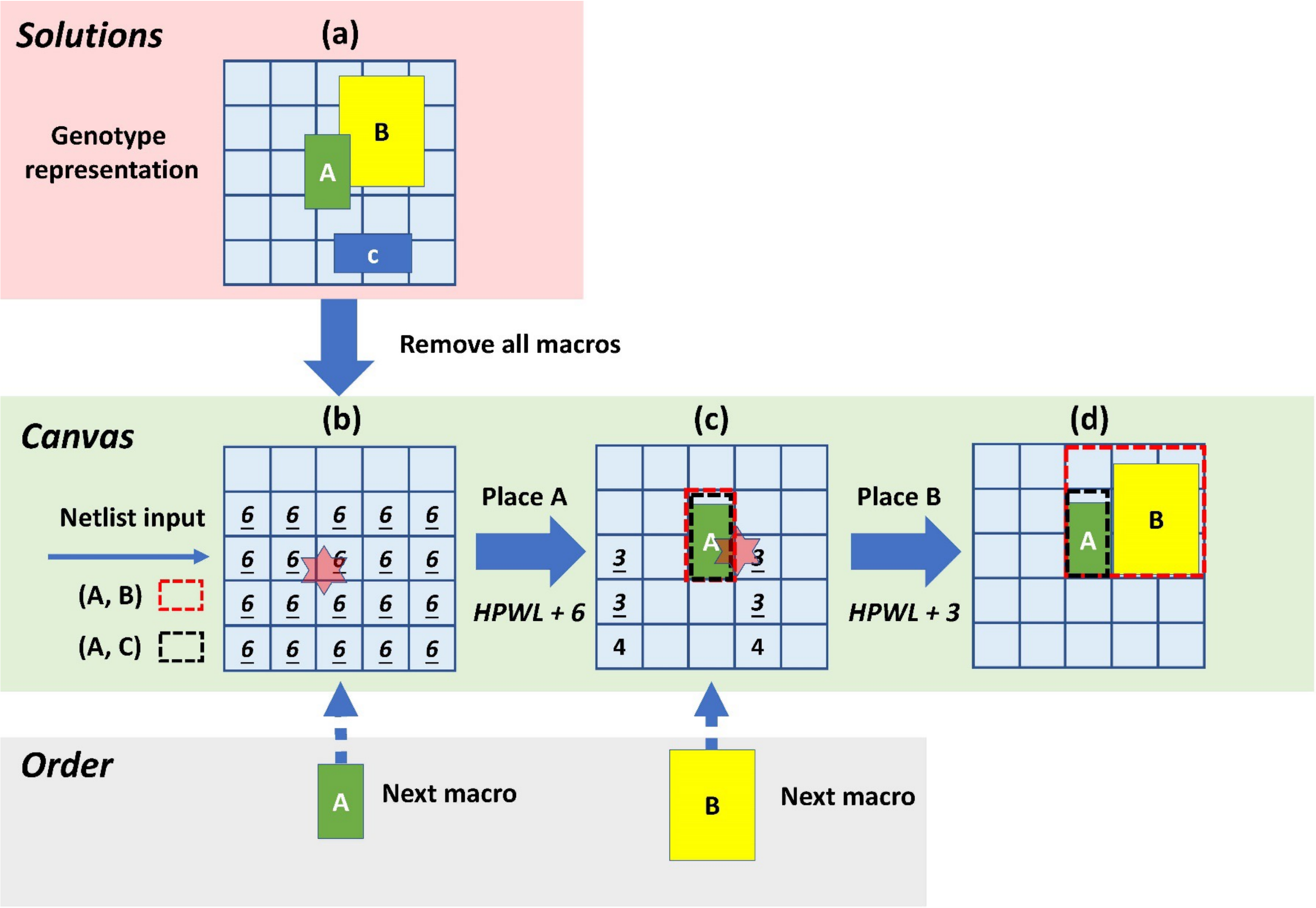
Calculate the wirelength increment after placing the macro at each grid

Proposed genotype-phenotype mapping



Refer to its original location

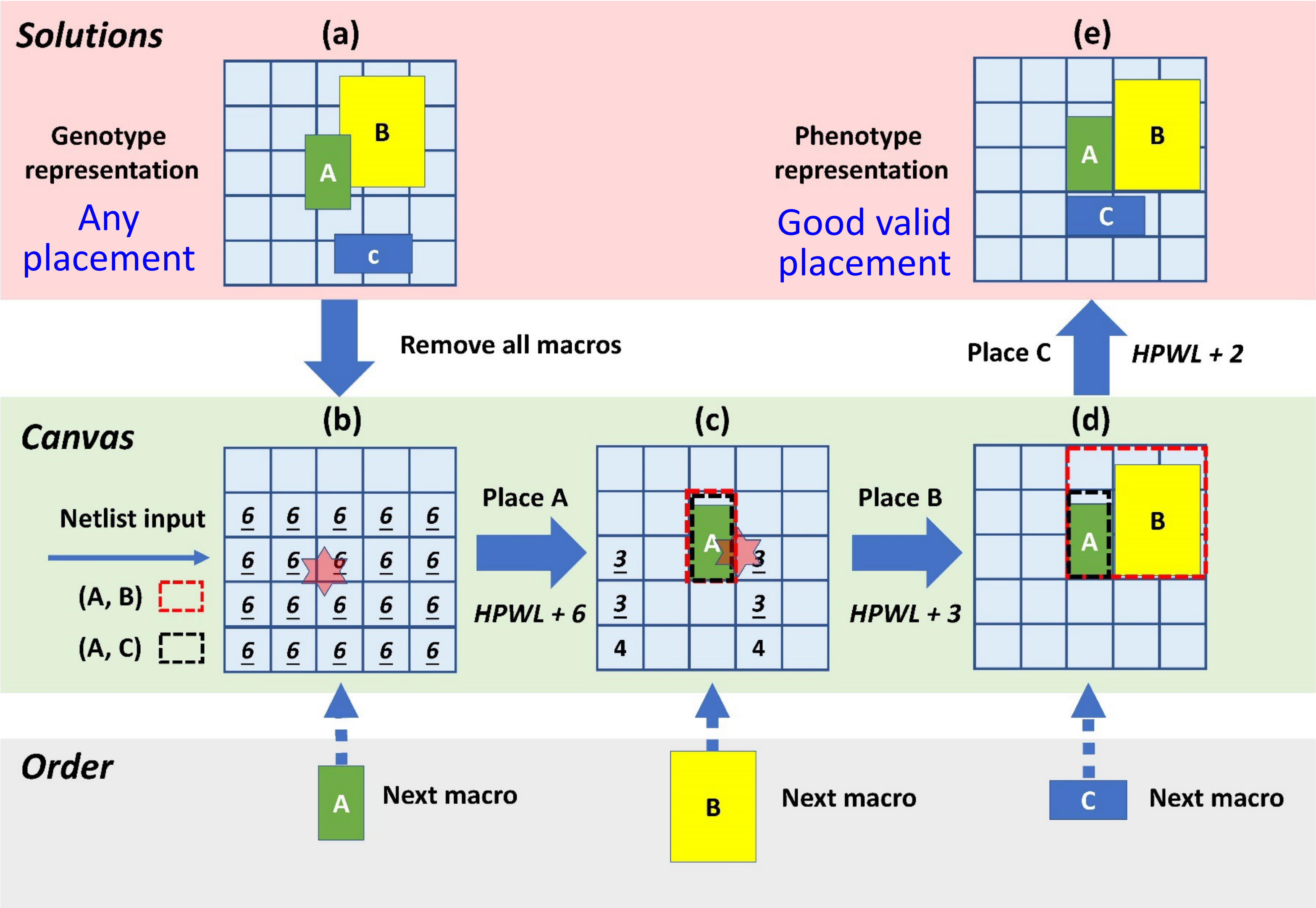
Proposed genotype-phenotype mapping



Choose the nearest best grid

Place the macro

Proposed genotype-phenotype mapping



Place macro C similarly

Obtain the valid phenotype placement result

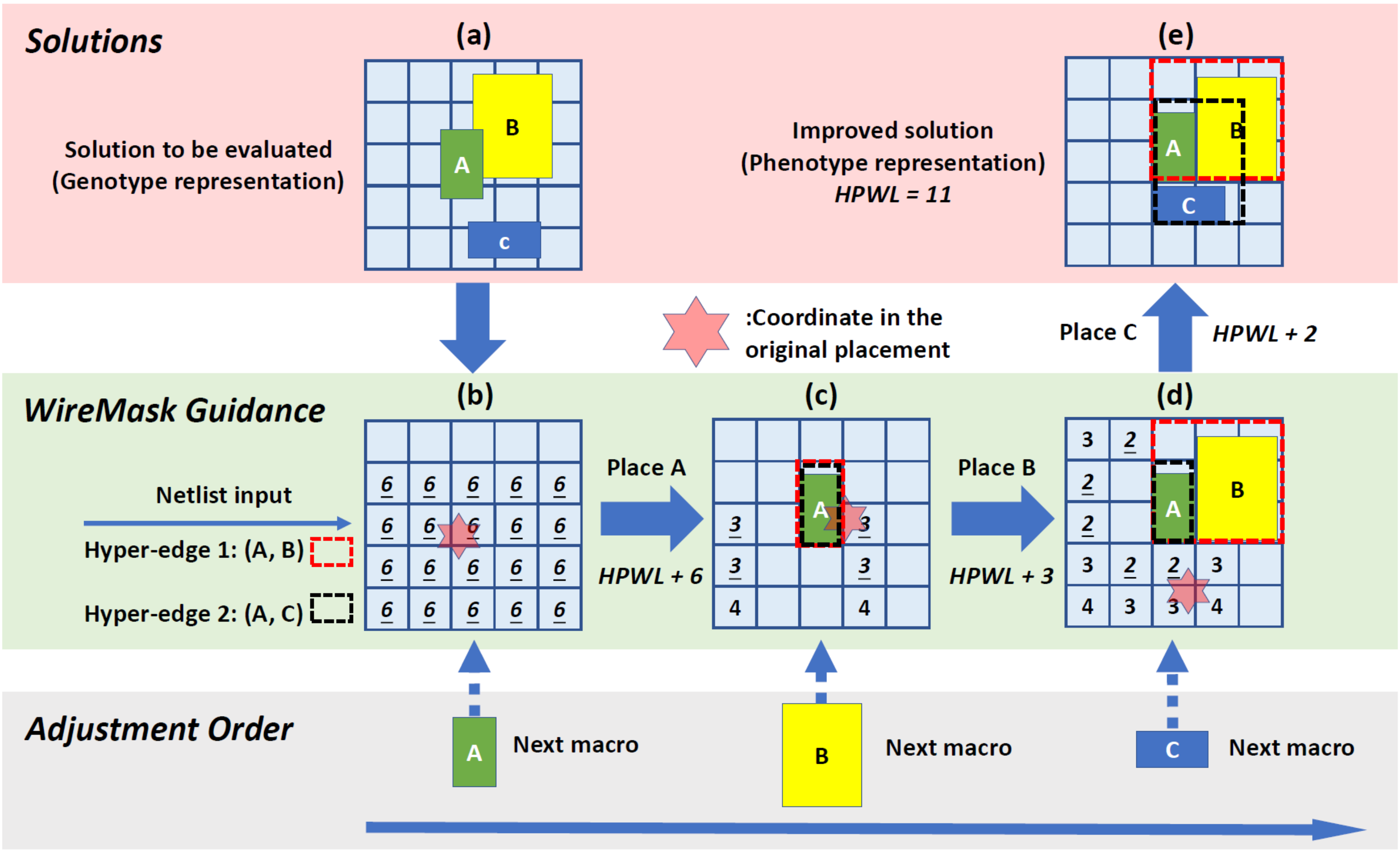
A greedy mapping based on the increment of wirelength

⇩

Improve the efficiency

Optimization based on proposed mapping

Genotype → Phenotype



➕ **Evolutionary Algorithm**

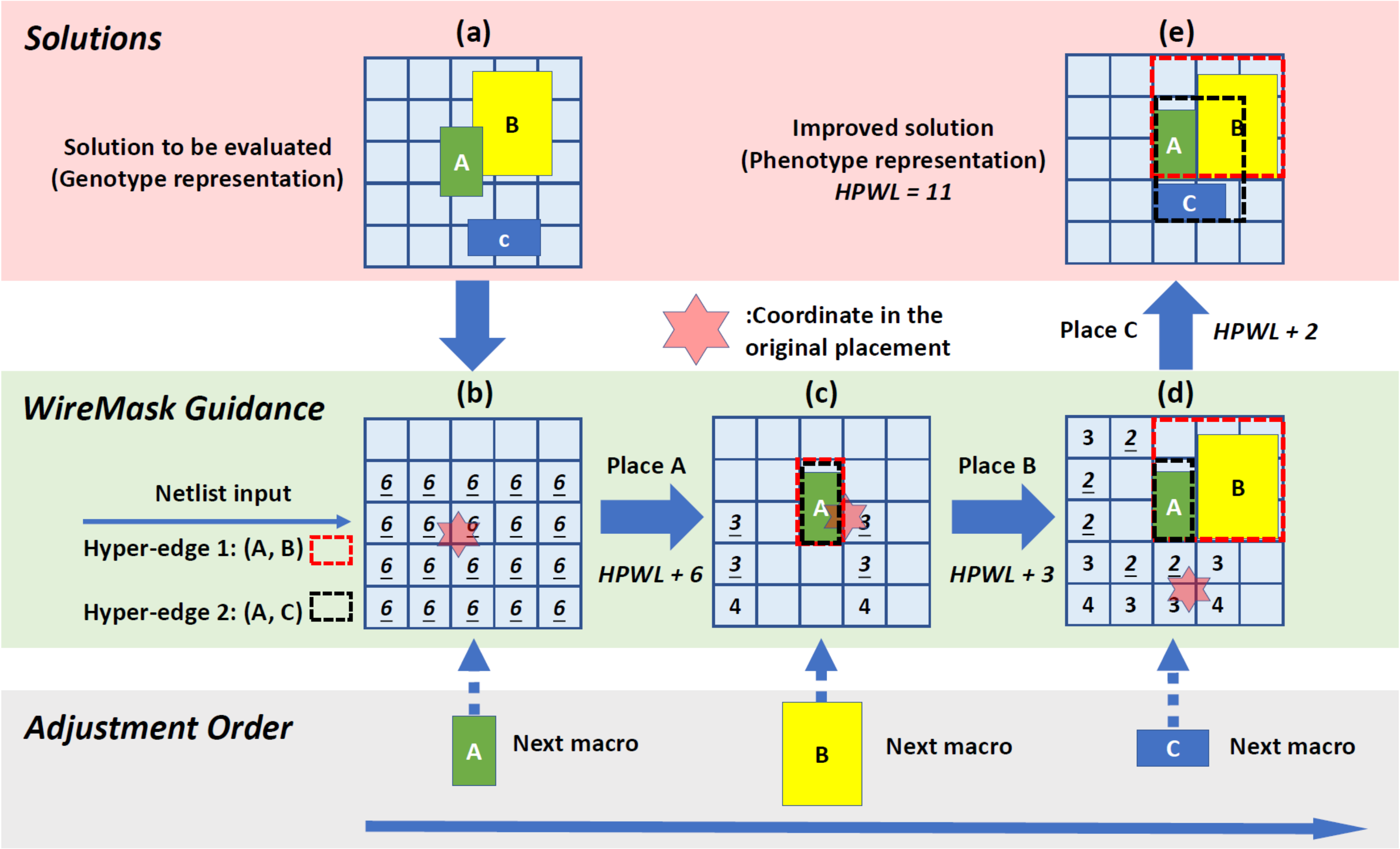
(1+1)-EA

- **Initialization:** Randomly generate 100 genotype solutions and pick the best
- **Mutation:** Randomly exchange two macros' locations

Already performs well!

Optimization based on proposed mapping

Genotype → Phenotype



➕

Any black-box optimization algorithm

(1+1)-EA

Bayesian optimization

Random search

Comparison with previous SOTA methods

Table 1: Wirelength ($\times 10^5$) obtained by ten compared methods on seven popular ISPD contest chips.

	Method	Type	adaptec1	adaptec2	adaptec3	adaptec4	bigblue1	bigblue3	bigblue4 ($\times 10^7$)	+/ − / ≈	Avg. Rank
Classical EA method	SP-SA [33]	Packing	18.84 ± 4.62	117.36 ± 8.73	115.48 ± 7.56	120.03 ± 4.25	5.12 ± 1.43	164.70 ± 19.55	25.49 ± 2.73	0/7/0	6.86
Analytical	NTUPlace3 [12]	Analytical	26.62	321.17	328.44	462.93	22.85	455.53	48.38	0/7/0	9.00
	RePlace [13]	Analytical	16.19 ± 2.10	153.26 ± 29.01	111.21 ± 11.69	37.64 ± 1.05	2.45 ± 0.06	119.84 ± 34.43	11.80 ± 0.73	1/6/0	5.28
	DREAMPlace [28]	Analytical	15.81 ± 1.64	140.79 ± 26.73	121.94 ± 25.05	**37.41 ± 0.87**	2.44 ± 0.06	107.19 ± 29.91	12.29 ± 1.64	1/6/0	4.86
RL	Graph [32]	RL	30.10 ± 2.98	351.71 ± 38.20	358.18 ± 13.95	151.42 ± 9.72	10.58 ± 1.29	357.48 ± 47.83	53.35 ± 4.06	0/7/0	9.00
	DeepPR [15]	RL	19.91 ± 2.13	203.51 ± 6.27	347.16 ± 4.32	311.86 ± 56.74	23.33 ± 3.65	430.48 ± 12.18	68.30 ± 4.44	0/7/0	8.86
	MaskPlace [26]	RL	6.38 ± 0.35	73.75 ± 6.35	84.44 ± 3.60	79.21 ± 0.65	2.39 ± 0.05	91.11 ± 7.83	11.07 ± 0.90	0/7/0	4.28
Our methods	WireMask-RS	Ours	6.13 ± 0.05	59.28 ± 1.48	60.60 ± 0.45	62.06 ± 0.22	2.19 ± 0.01	62.58 ± 2.07	**8.20 ± 0.17**	0/5/2	2.57
	WireMask-BO	Ours	6.07 ± 0.14	59.17 ± 3.94	61.00 ± 2.08	63.86 ± 1.01	2.14 ± 0.03	67.48 ± 6.49	8.62 ± 0.18	0/3/4	2.86
	WireMask-EA	Ours	**5.91 ± 0.07**	**52.63 ± 2.23**	**57.75 ± 1.16**	58.79 ± 1.02	**2.12 ± 0.01**	**59.87 ± 3.40**	8.28 ± 0.25	–	**1.43**

[Google, Nature'21]

Some important baselines

- **Graph** [Nature'21]: RL method proposed by **Google**
- **DREAMPlace** [DAC'19, TCAD'21 Best Paper] : One of the most popular analytical methods
- **DeepPR** [NeurIPS'21] and **MaskPlace** [NeurIPS'22] : Two recent advanced RL methods

WireMask-EA (our proposed framework equipped with EA) achieves **the best average rank**, and **reduces wirelength by 80%** compared to [Google, Nature'21]

Comparison with the latest method ChiPFormer [Lai et al., ICML'23]

Table 2: Wirelength ($\times 10^5$) Compared with ChiPFormer on ten popular ISPD and ICCAD contest chips.

Benchmark	ChiPFormer (1)	WireMask-EA (1)	ChiPFormer (0.3k)	WireMask-EA (0.3k)	ChiPFormer (2k)	WireMask-EA (2k)
adaptec1	8.87 ± 0.98	**7.20 ± 0.34**	7.02 ± 0.11	**6.29 ± 0.07**	6.62 ± 0.05	**5.96 ± 0.08**
adaptec2	122.37 ± 22.61	**111.04 ± 20.09**	70.42 ± 2.67	**61.25 ± 4.10**	67.10 ± 5.46	**53.88 ± 2.53**
adaptec3	107.11 ± 8.84	**75.37 ± 2.93**	78.32 ± 2.03	**64.49 ± 1.69**	76.70 ± 1.15	**59.26 ± 1.30**
adaptec4	85.63 ± 7.52	**75.63 ± 1.30**	69.42 ± 0.54	**64.52 ± 1.81**	68.80 ± 1.59	**59.52 ± 1.71**
bigblue1	3.11 ± 0.03	**2.31 ± 0.06**	2.96 ± 0.04	**2.18 ± 0.01**	2.95 ± 0.04	**2.14 ± 0.01**
bigblue3	131.78 ± 17.36	**99.20 ± 24.69**	81.48 ± 4.83	**64.51 ± 4.15**	72.92 ± 2.56	**56.65 ± 2.81**
ibm01	4.57 ± 0.27	**3.76 ± 0.36**	3.61 ± 0.08	**2.92 ± 0.07**	3.05 ± 0.11	**2.39 ± 0.07**
ibm02	6.01 ± 0.41	**5.13 ± 0.16**	4.84 ± 0.17	**3.86 ± 0.03**	4.24 ± 0.25	**3.56 ± 0.05**
ibm03	**2.15 ± 0.17**	3.10 ± 0.12	**1.75 ± 0.07**	2.20 ± 0.11	**1.64 ± 0.06**	1.69 ± 0.11
ibm04	5.00 ± 0.14	**3.60 ± 0.17**	4.19 ± 0.11	**2.93 ± 0.11**	4.06 ± 0.13	**2.62 ± 0.04**

WireMask-EA outperforms ChiPFormer [Lai et al., ICML'23] on **9 out of 10** chips, using the same number of evaluations

Comparison with expert-crafted results

Learning And Mining from DatA
http://www.lamda.nju.edu.cn

circuit	method	Timing		NVP[1]	Congestion	
		WNS/ps	TNS/ns		H/%	V/%
C1	Human	204	57.1	2569	0.06	0.38
	MaskPlace	161	42.7	1964	0.07	**0.07**
	ChiPFormer	**142**	**19.4**	**1636**	**0.04**	**0.07**
C2	Human	403	492.2	11360	0.63	2.05
	MaskPlace	242	259.1	9710	0.57	1.67
	ChiPFormer	**177**	**224.9**	**8110**	**0.53**	**1.27**
C3	Human	**102**	91.9	5614	**1.02**	0.85
	MaskPlace	116	92.8	5559	1.05	0.87
	ChiPFormer	108	**91.2**	**5452**	**1.02**	**0.82**
C4	Human	399	438.0	13925	0.97	**0.34**
	MaskPlace	389	324.2	12582	0.68	**0.34**
	ChiPFormer	**248**	**266.0**	**12398**	**0.62**	**0.34**
C5	Human	89	10.8	2675	**0.02**	0.07
	MaskPlace	122	32.2	2975	**0.02**	0.22
	ChiPFormer	**80**	**4.9**	**1706**	**0.02**	**0.04**
C6	Human	154	137.4	6833	0.70	0.22
	MaskPlace	81	49.6	7040	0.77	0.26
	ChiPFormer	**78**	**38.1**	**6412**	**0.63**	**0.22**

[Lai et al., ICML'23]

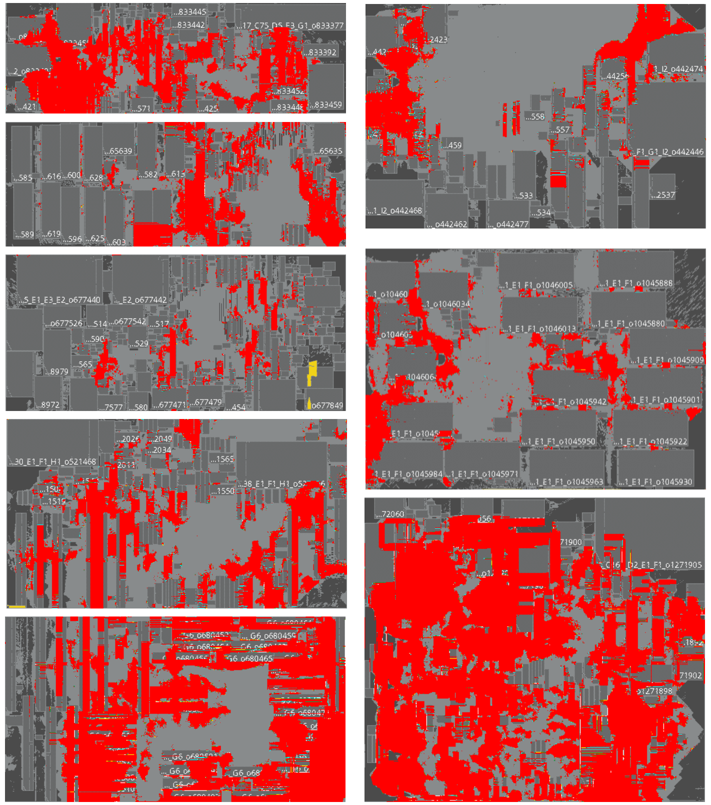
Experiments on **private industry chip cases:**

ChiPFormer **outperforms** human experts (who place the macros with the help of commercial tool *Candence Innovus*)

WireMask-EA > ChiPFormer > Human expert

Visualization by commercial tool *Cadence Innovus*

Learning And Mining from DatA
http://www.lamda.nju.edu.cn

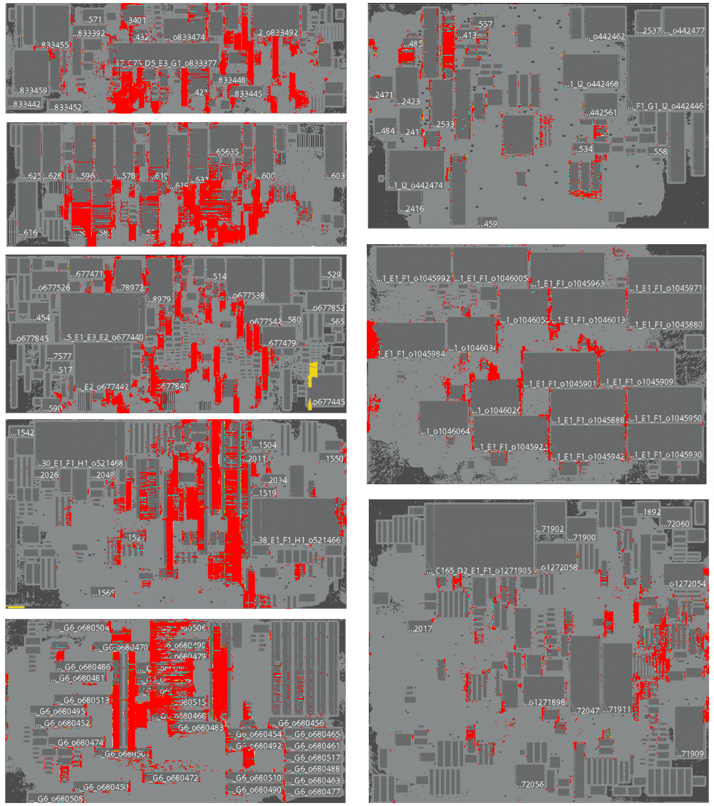


DREAMPlace

Most popular analytical placer

DAC’19 **Best Paper**

TCAD’21 **Best Paper**



WireMask-EA

Our proposed framework

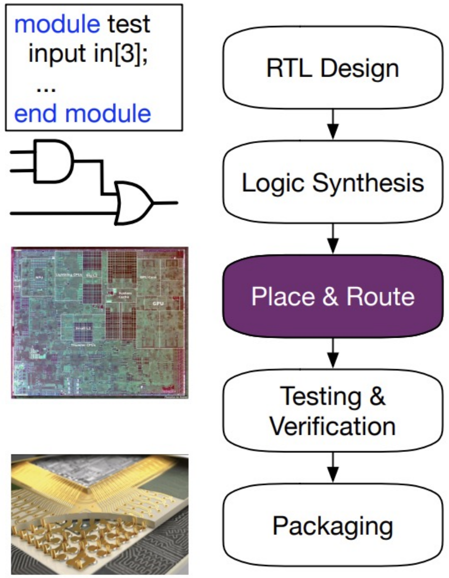
equipped with **EA**

Red points in the figure represent congestion points that can’t be routed

Our results have **much fewer red points**, showing **better routability and performance**

Why we are the best?

The problem we solve



Macro Placement

A vital stage in chip design

The baselines we beat

Google nature

ICML International Conference On Machine Learning

NEURAL INFORMATION PROCESSING SYSTEMS

TCAD, DAC best paper

human experts

Significant improvement

(e.g., 80% wirelength improvement over [Google, Nature'21])

The contributions to community

- **Bring EAs back to the state-of-the-art for macro placement**
- **Reaffirm the potential of EAs for chip design**

We have inspired a recent RL method

LAMDA
Learning And Mining from DatA
http://www.lamda.nju.edu.cn

Reinforcement Learning within Tree Search for Fast Macro Placement

[Geng et al., ICML'24]

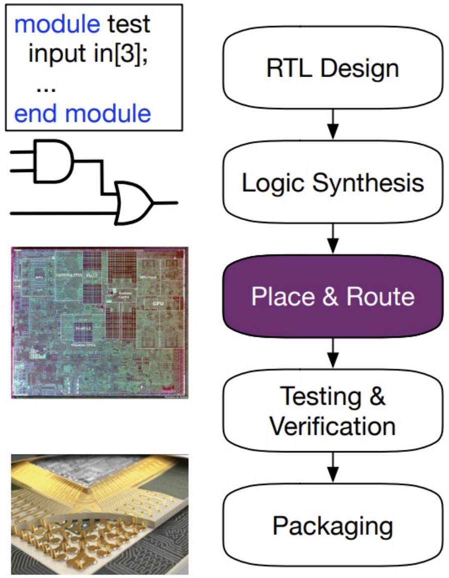
Shi, Y., Xue, K., Song, L., and Qian, C. Macro placement by wire-mask-guided black-box optimization. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Wiremask for Reducing Search Space The concept of wiremask was introduced by Lai et al. (2022) as the visual inputs to the neural networks. Shi et al. (2023) then employed wiremask to devise a greedy policy to guide the BBO algorithms. Similar to them, we restrict actions to the grid areas with the minimal HPWL increment, which narrows down the search space, thereby significantly enhancing the training efficiency and the placement quality.

Inspired by our work, **EfficientPlace** chooses the action **from the feasible grids with minimum wirelength increment**, originated from our genotype-phenotype mapping, enhancing RL searching

Why we are the best?

The problem we solve



Macro Placement

A vital stage in chip design

The baselines we beat

Google nature

ICML International Conference On Machine Learning

NEURAL INFORMATION PROCESSING SYSTEMS

TCAD, DAC best paper

human experts

Significant improvement

(e.g., 80% wirelength improvement over [Google, Nature'21])

The contributions to community

- **Bring EAs back to the state-of-the-art for macro placement**
- **Reaffirm the potential of EAs for chip design**

We are working on

- **Benchmark chip design cases as real-world problems for EAs**

Thanks for listening!

https://github.com/lamda-bbo/WireMask-BBO

http://www.lamda.nju.edu.cn/qianc/