DATE 26
DESIGN, AUTOMATION AND TEST IN EUROPE
THE EUROPEAN EVENT FOR ELECTRONIC SYSTEM DESIGN & TEST
20 – 22 APRIL 2026
VERONA, ITALY
PALAZZO DELLA GRAN GUARDIA

AutoShrink: Adaptive Search Space Shrinkage for Large-Scale Pareto Optimization of HLS Designs

Yingxin Zeng[1], Binghao Cheng[1], Jianwang Zhai[2], Kang Zhao[2], Zhe Lin[1]*

[1]Sun Yat-sen University, China
[2]Beijing University of Posts and Telecommunications, China

中山大學 SUN YAT-SEN UNIVERSITY

北京郵電大學 Beijing University of Posts and Telecommunications

Large-scale HLS DSE

- High-Level Synthesis (HLS)
  - Hardware design using C/C++
  - Optimization directives
  - Different directive configurations
    → Different Quality of Results(QoR)
    → **Vast design space**
- Manual directive tuning for a satisfactory QoR tradeoff relies on
  - Expert knowledge
  - Time-consuming repeated trial-and-error
- **Solution: Design Space Exploration (DSE)**
  - Automatically finding high-quality directive configurations

Existing Surrogate-Based HLS DSE Faces a Dilemma

- Surrogate model: predicts QoR of unexplored designs
- Online-calibrated surrogate models:
  - e.g., Prospector[1],Sherlock [2]
  - Build and update a design-specific surrogate online
- Pre-trained surrogate models:
  - e.g., Ironman-pro[3], GNN-DSE [4]
  - Use a pre-trained predictive model as a well-defined surrogate model

Slow	**Fast**
Need many HLS evaluations before becoming reliable	Avoiding time-consuming online calibration
Adaptive	**Less Generalizable**
Calibrated to each target design	May not match unseen design domains

- In large-scale design spaces, the limitations of both approaches become more severe.

Fast & Adaptive ?

Existing Surrogate-Based HLS DSE Faces a Dilemma

- Surrogate model: predicts QoR of unexplored designs
- Online-calibrated surrogate models:
  - e.g., Prospector[1],Sherlock [2]
  - Build and update a design-specific surrogate online
- Pre-trained surrogate models:
  - e.g., Ironman-pro[3], GNN-DSE [4]
  - Use a pre-trained predictive model as a well-defined surrogate model

Slow	→	**Fast**
Need many HLS evaluations before becoming reliable		Avoiding time-consuming online calibration
Adaptive	←	**Less Generalizable**
Calibrated to each target design		May not match unseen design domains

- In large-scale design spaces, the limitations of both approaches become more severe.

Fast & Adaptive ?

Existing Surrogate-Based HLS DSE Faces a Dilemma

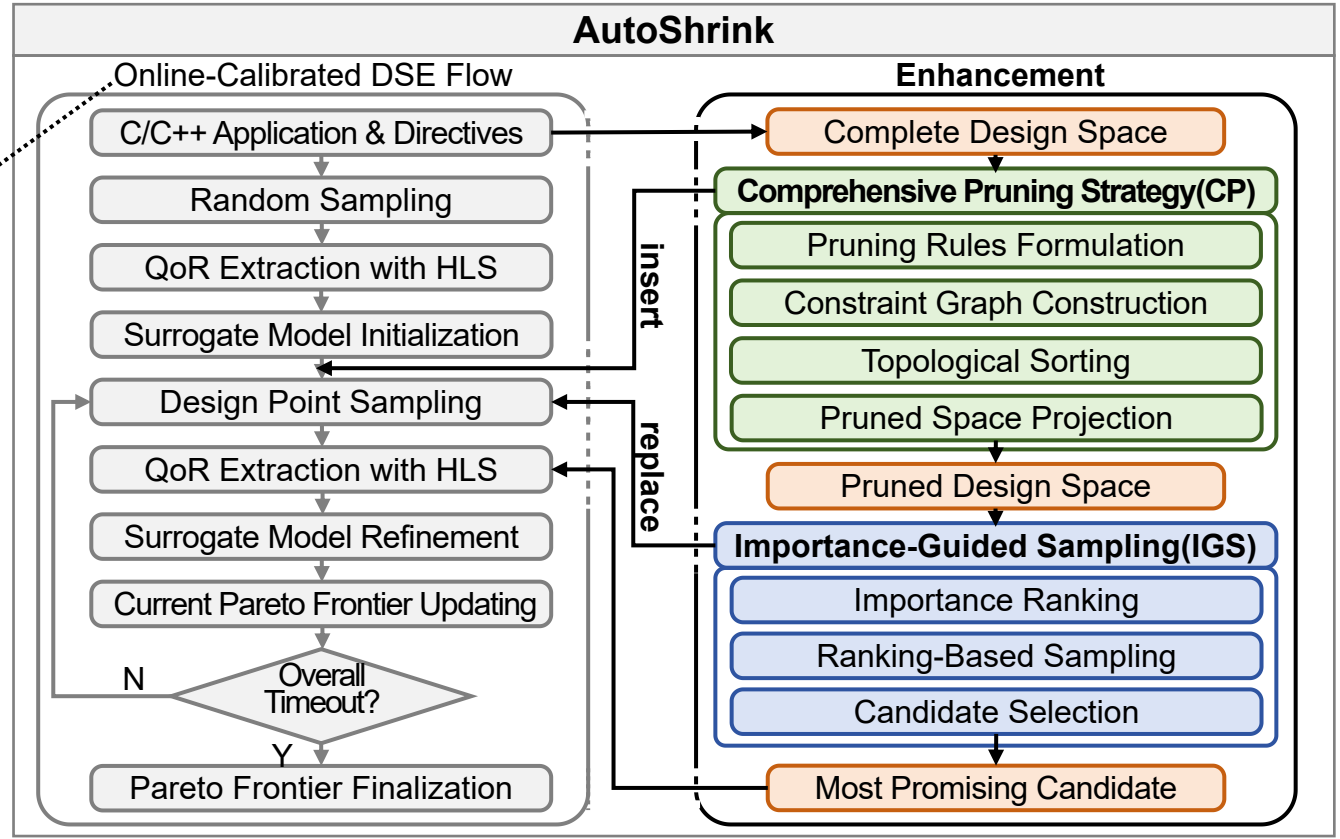
- Surrogate model: predicts QoR of unexplored designs
- Online-calibrated surrogate models:
  - e.g., Prospector[1],Sherlock [2]
  - Build and update a design-specific surrogate online
- Pre-trained surrogate models:
  - e.g., Ironman-pro[3], GNN-DSE [4]
  - Use a pre-trained predictive model as a well-defined surrogate model

Slow	→	**Fast**
Need many HLS evaluations before becoming reliable		Avoiding time-consuming online calibration
Adaptive	←	**Less Generalizable**
Calibrated to each target design		May not match unseen design domains

- In large-scale design spaces, the limitations of both approaches become more severe.

Fast & Adaptive ?

AutoShrink Overview



Naturally Adaptive

- Prune redundant and inefficient configurations at scale
- Track directive importance to guide the search

Comprehensive Pruning Strategy

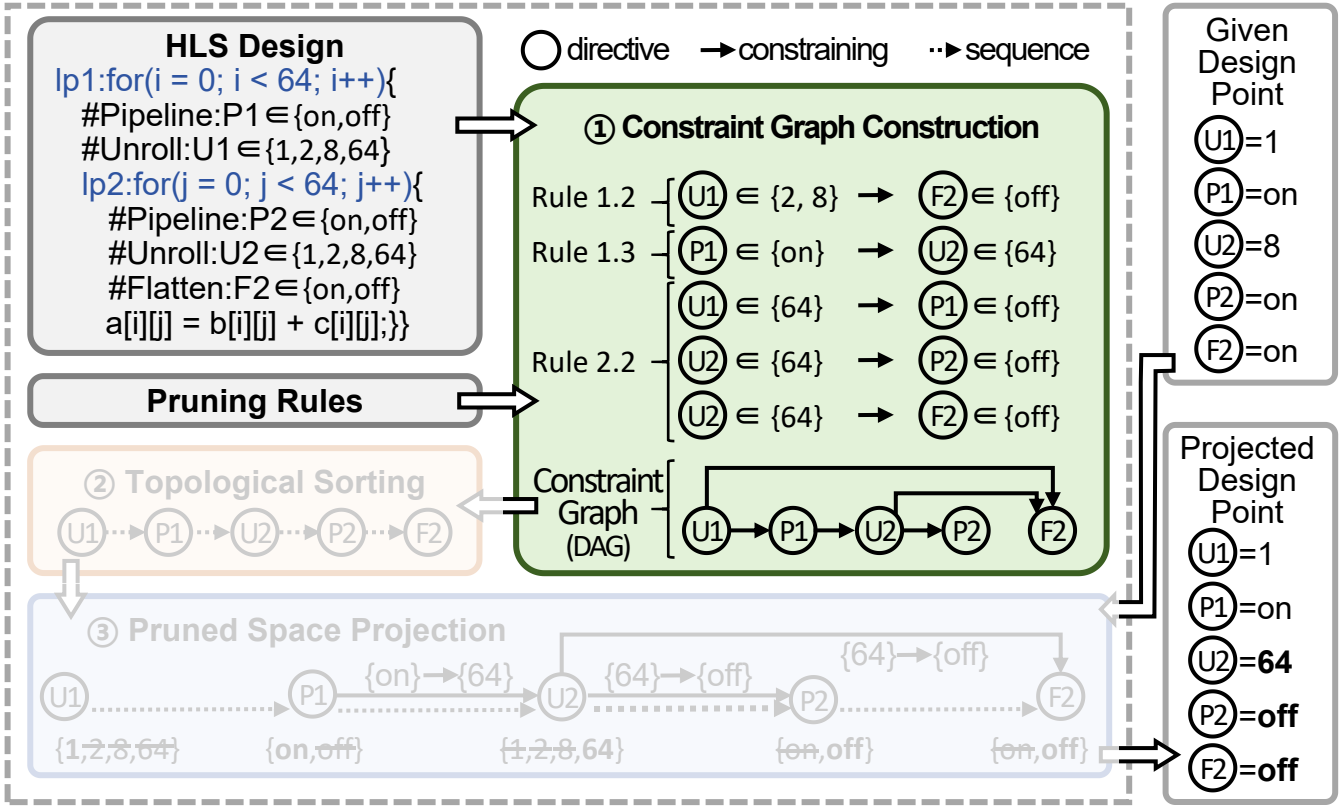
Observation 1:

- **Directive interactions** cause a large number of **redundancy and under-optimization**

Pruning Rules Formulation

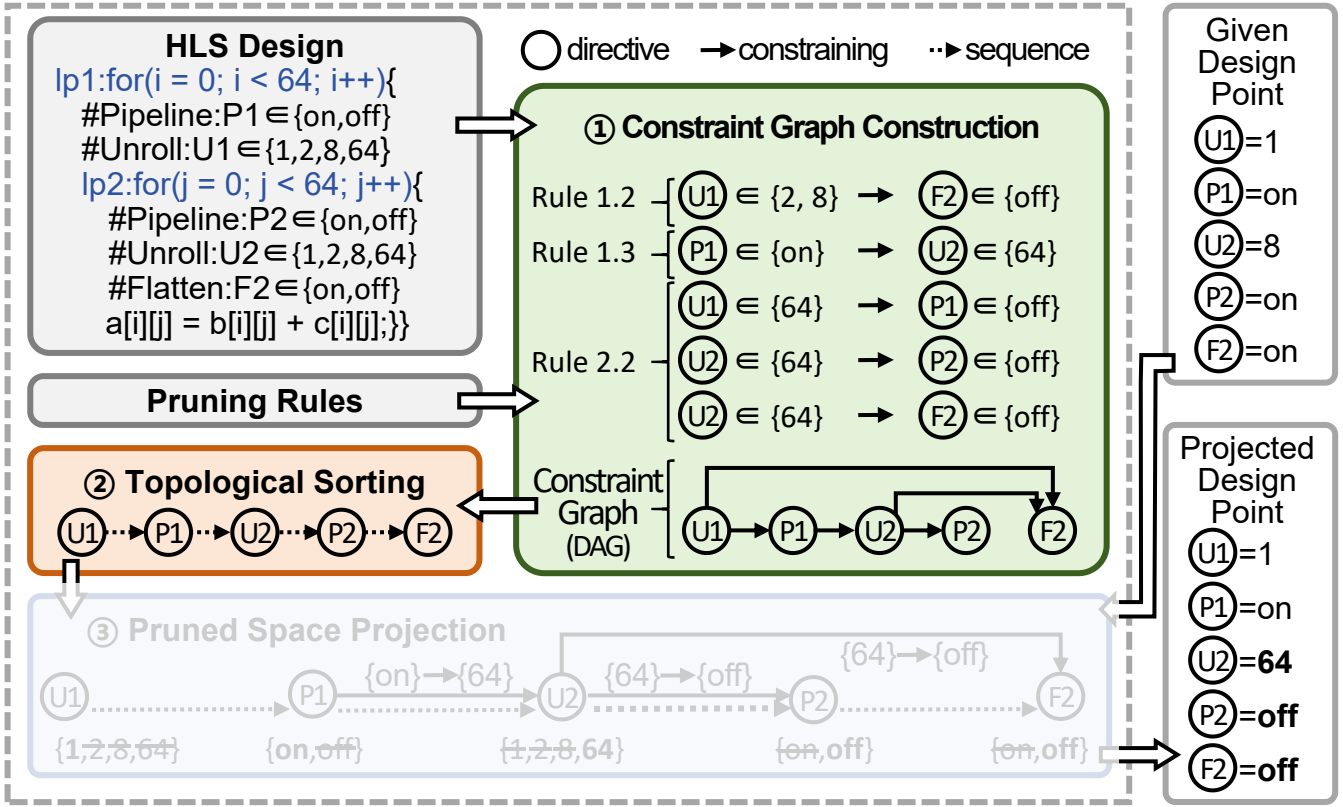
Rule	Constraining Directive	Configuration	Constrained Directive	Configuration	
1.1	Function Dataflow	On	Inner Functions' Inline	Off	*Exclusivity*
1.2	Loop Unroll	Partially Unroll	Inner Loops' Flatten	Off	
1.3	Loop Pipeline	On	Inner Loops' Unroll	Fully Unroll	
2.1	Function Inline	On	Same Function's Dataflow	Off	*Elimination*
2.2	Loop Unroll	Fully Unroll	Same Loop's Flatten	Off	
			Same Loop's Pipeline	Off	
3.1	Loop Unroll (Access Same Array)	List of Unroll Factors	Accessed Array's Partition	Max of Unroll Factors	*Coordination*
3.2	Loop Merge	On	Inner Functions' Inline	On	

Comprehensive Pruning Strategy



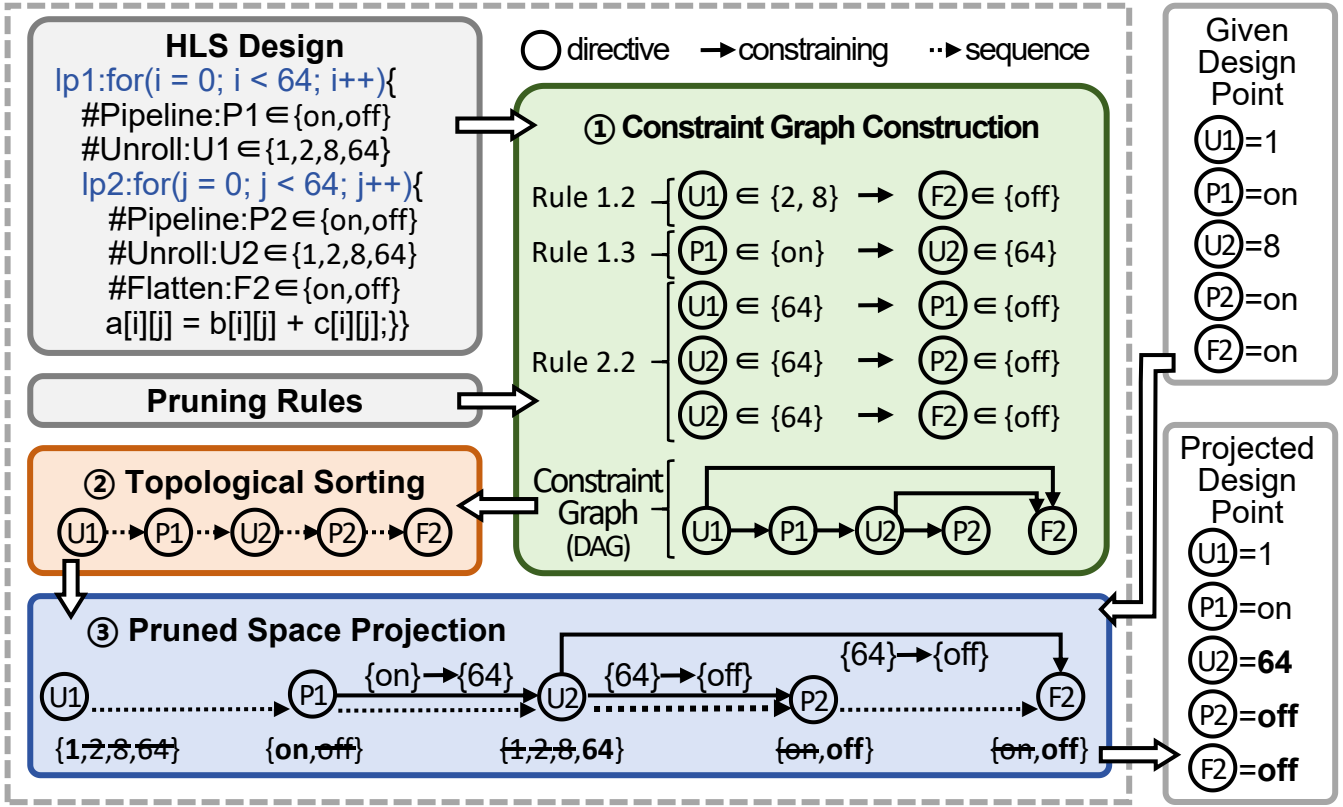
- Avoid rule conflicts
- Filters out invalid configurations
- No pruned-space enumeration

Comprehensive Pruning Strategy



- Avoid rule conflicts
- Filters out invalid configurations
- No pruned-space enumeration

Comprehensive Pruning Strategy



- Avoid rule conflicts
- Filters out invalid configurations
- No pruned-space enumeration

Importance-Guided Sampling

Observation 2: Not all directives matter equally for QoR, and only a few dominate performance and resource use.

→ Uniform sampling wastes effort
→ Keep important directives more stable
→ Vary less important directives more
→ Faster convergence

Importance-Guided Sampling

① Using fANOVA (Functional Analysis of Variance) to Measure Directive Importance

QoR (Latency / Resource Usage) — *The i-th Directive*

$$f(\mathrm{x}) = f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_{i<j} f_{ij}(x_i, x_j) + \ldots + f_{1,2,\ldots,n}(x_1, x_2, \ldots, x_n)$$

$$I_i = \frac{\mathrm{Var}\left[f_i(x_i)\right]}{\mathrm{Var}\left[f(\mathrm{x})\right]}$$

Variation Caused by the i-th Directive / *Total QoR Variation*

Importance I_i **:** the fraction of total performance variation caused solely by the i-th directive

② Ranking Directives by Importance

③ Ranking-Based Sampling

Importance-Guided Sampling

① Using fANOVA (Functional Analysis of Variance) to Measure Directive Importance

QoR (Latency / Resource Usage) — *The i-th Directive*

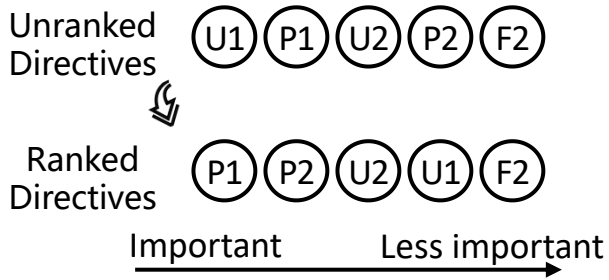
$$f(\mathrm{x}) = f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_{i<j} f_{ij}(x_i, x_j) + \ldots + f_{1,2,\ldots,n}(x_1, x_2, \ldots, x_n)$$

$$I_i = \frac{\mathrm{Var}[f_i(x_i)]}{\mathrm{Var}[f(\mathrm{x})]}$$

Variation Caused by the i-th Directive / *Total QoR Variation*

Importance I_i **:** the fraction of total performance variation caused solely by the i-th directive

② Ranking Directives by Importance



③ Ranking-Based Sampling

Importance-Guided Sampling

① Using fANOVA (Functional Analysis of Variance) to Measure Directive Importance

QoR (Latency / Resource Usage) — *The i-th Directive*

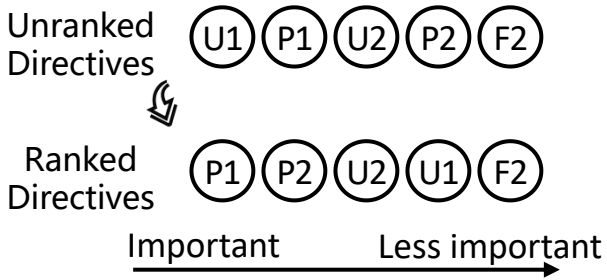
$$f(\mathrm{x}) = f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_{i<j} f_{ij}(x_i, x_j) + \ldots + f_{1,2,\ldots,n}(x_1, x_2, \ldots, x_n)$$

$$I_i = \frac{\mathrm{Var}\left[f_i(x_i)\right]}{\mathrm{Var}\left[f(\mathrm{x})\right]}$$

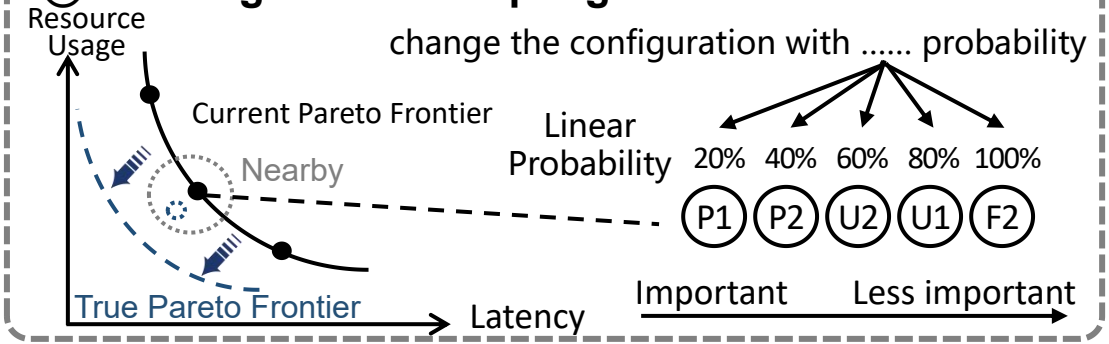
Variation Caused by the i-th Directive / *Total QoR Variation*

Importance I_i **:** the fraction of total performance variation caused solely by the i-th directive

② Ranking Directives by Importance



③ Ranking-Based Sampling



Experimental Results

Fair comparation:

- same time budget
- same initialization
- ……

AutoShrink:

- **5.73× / 4.47× improvement in ADRS over generic / SOTA methods, respectively**
- CP removes **2-6 orders** of magnitude of redundant and inefficient configurations
- Two components effective individually; synergistic when combined

Benchmark	Design Space Size		ADRS (%, Arithmetic Mean ± Standard Deviation over 3 Runs)								
	Original	Pruned (ours)	GA	RL	SA	Sherlock	ScaleHLS	Variants for Ablation			AutoShrink (ours)
								w/o CP\&IGS	w/o CP	w/o IGS	
aes	1.24×10^{12}	2.01×10^{7}	93.12±37.59	110.06±47.67	34.34±13.38	91.51±19.19	47.54±10.84	30.40±6.08	40.86±3.85	23.35±7.20	**18.38±6.89**
atax	5.31×10^{6}	7.40×10^{3}	41.69±19.74	40.52±12.91	17.45±7.13	42.02±19.96	22.06±6.53	15.60±7.23	12.78±7.69	8.68±2.90	**5.28±1.00**
backprop	2.41×10^{14}	9.75×10^{7}	51.52±18.32	55.33±21.23	23.52±15.11	44.51±19.60	31.31±12.80	17.10±7.48	22.91±6.90	6.05±1.20	**4.05±1.95**
bicg	6.37×10^{7}	1.11×10^{4}	52.68±4.43	40.95±17.58	24.78±10.66	53.07±3.25	30.93±9.57	30.84±7.56	26.48±7.76	10.81±1.82	**5.32±3.22**
fft	1.08×10^{16}	1.37×10^{10}	74.30±23.63	71.55±16.74	31.31±14.38	58.07±18.12	34.47±4.74	31.67±20.00	31.20±8.69	71.54±27.44	**12.36±2.99**
gemm	1.42×10^{7}	3.01×10^{4}	30.77±5.89	33.76±6.73	22.13±12.46	55.01±35.43	23.40±3.55	48.82±41.17	25.36±1.16	22.87±12.49	**16.56±12.47**
gesummv	9.95×10^{5}	1.20×10^{3}	29.28±13.09	20.57±9.62	10.55±2.56	24.97±5.54	12.96±6.27	14.33±5.97	13.44±8.90	3.42±1.43	**2.95±1.64**
md_knn	2.10×10^{5}	9.00×10^{1}	12.02±2.90	8.54±5.06	9.71±0.85	12.23±2.45	7.08±5.16	1.52±1.01	3.90±4.37	1.19±0.62	**0.13±0.08**
mvt	1.91×10^{8}	1.11×10^{4}	47.37±4.96	52.37±25.59	29.24±3.62	59.65±16.32	39.27±11.93	25.52±2.84	27.89±2.97	31.79±1.06	**19.62±7.06**
spam_filter	4.29×10^{9}	1.62×10^{6}	142.89±45.96	137.28±42.08	80.13±28.58	116.30±60.65	96.51±10.08	18.83±12.26	22.16±11.58	11.79±6.56	**1.63±1.35**
spmv	7.78×10^{3}	3.00×10^{1}	17.19±2.12	10.92±0.57	10.13±0.69	12.52±1.57	10.37±0.97	1.55±0.79	0.73±0.35	1.97±0.54	**0.25±0.22**
stencil3d	1.22×10^{10}	7.13×10^{6}	79.90±11.69	75.03±19.19	47.10±26.06	72.53±11.04	41.10±37.01	28.32±4.08	25.68±16.00	22.51±14.25	**10.64±3.62**
Geo. Mean	/	/	45.5	41.39	23.45	44.21	26.53	15.4	15.2	10.32	**3.94**
Arith. Mean	/	/	56.06	54.74	28.37	53.53	33.08	22.04	21.11	18	**8.1**

Importance-Guided Sampling (IGS)
Comprehensive Pruning (CP)

Experimental Results

Fair comparation:
- same time budget
- same initialization
- ……

AutoShrink:
- **5.73× / 4.47× improvement in ADRS over generic / SOTA methods, respectively**
- CP removes **2-6 orders** of magnitude of redundant and inefficient configurations
- Two components effective individually; synergistic when combined

Benchmark	Design Space Size		ADRS (%, Arithmetic Mean ± Standard Deviation over 3 Runs)								
	Original	Pruned (ours)	GA	RL	SA	Sherlock	ScaleHLS	Variants for Ablation			AutoShrink (ours)
								w/o CP\&IGS	w/o CP	w/o IGS	
aes	1.24×10^{12}	2.01×10^{7}	93.12±37.59	110.06±47.67	34.34±13.38	91.51±19.19	47.54±10.84	30.40±6.08	40.86±3.85	23.35±7.20	**18.38±6.89**
atax	5.31×10^{6}	7.40×10^{3}	41.69±19.74	40.52±12.91	17.45±7.13	42.02±19.96	22.06±6.53	15.60±7.23	12.78±7.69	8.68±2.90	**5.28±1.00**
backprop	2.41×10^{14}	9.75×10^{7}	51.52±18.32	55.33±21.23	23.52±15.11	44.51±19.60	31.31±12.80	17.10±7.48	22.91±6.90	6.05±1.20	**4.05±1.95**
bicg	6.37×10^{7}	1.11×10^{4}	52.68±4.43	40.95±17.58	24.78±10.66	53.07±3.25	30.93±9.57	30.84±7.56	26.48±7.76	10.81±1.82	**5.32±3.22**
fft	1.08×10^{16}	1.37×10^{10}	74.30±23.63	71.55±16.74	31.31±14.38	58.07±18.12	34.47±4.74	31.67±20.00	31.20±8.69	71.54±27.44	**12.36±2.99**
gemm	1.42×10^{7}	3.01×10^{4}	30.77±5.89	33.76±6.73	22.13±12.46	55.01±35.43	23.40±3.55	48.82±41.17	25.36±1.16	22.87±12.49	**16.56±12.47**
gesummv	9.95×10^{5}	1.20×10^{3}	29.28±13.09	20.57±9.62	10.55±2.56	24.97±5.54	12.96±6.27	14.33±5.97	13.44±8.90	3.42±1.43	**2.95±1.64**
md_knn	2.10×10^{5}	9.00×10^{1}	12.02±2.90	8.54±5.06	9.71±0.85	12.23±2.45	7.08±5.16	1.52±1.01	3.90±4.37	1.19±0.62	**0.13±0.08**
mvt	1.91×10^{8}	1.11×10^{4}	47.37±4.96	52.37±25.59	29.24±3.62	59.65±16.32	39.27±11.93	25.52±2.84	27.89±2.97	31.79±1.06	**19.62±7.06**
spam_filter	4.29×10^{9}	1.62×10^{6}	142.89±45.96	137.28±42.08	80.13±28.58	116.30±60.65	96.51±10.08	18.83±12.26	22.16±11.58	11.79±6.56	**1.63±1.35**
spmv	7.78×10^{3}	3.00×10^{1}	17.19±2.12	10.92±0.57	10.13±0.69	12.52±1.57	10.37±0.97	1.55±0.79	0.73±0.35	1.97±0.54	**0.25±0.22**
stencil3d	1.22×10^{10}	7.13×10^{6}	79.90±11.69	75.03±19.19	47.10±26.06	72.53±11.04	41.10±37.01	28.32±4.08	25.68±16.00	22.51±14.25	**10.64±3.62**
Geo. Mean	/	/	45.5	41.39	23.45	44.21	26.53	15.4	15.2	10.32	**3.94**
Arith. Mean	/	/	56.06	54.74	28.37	53.53	33.08	22.04	21.11	18	**8.1**

Importance-Guided Sampling (IGS)
Comprehensive Pruning (CP)

Thank you !

Contact me: zengyx29@mail2.sysu.edu.cn

[1] A. Mehrabi, A. Manocha, B. C. Lee, and D. J. Sorin, "Prospector: Synthesizing efficient accelerators via statistical learning," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2020, pp. 151–156.
[2] Q. Gautier, A. Althoff, C. L. Crutchfield, and R. Kastner, "Sherlock: A multi-objective design space exploration framework," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 27, no. 4, pp. 1–20, 2022.
[3] N. Wu, Y. Xie, and C. Hao, "Ironman-pro: Multiobjective design space exploration in hls via reinforcement learning and graph neural network-based modeling," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 42, no. 3, pp. 900–913,2022.
[4] A. Sohrabizadeh, Y. Bai, Y. Sun, and J. Cong, "Automated accelerator optimization aided by graph neural networks," in *ACM/IEEE Design Automation Conference (DAC)*, 2022, pp. 55–60.