Leveraging Prior Knowledge for Effective Design-Space Exploration in High-Level Synthesis

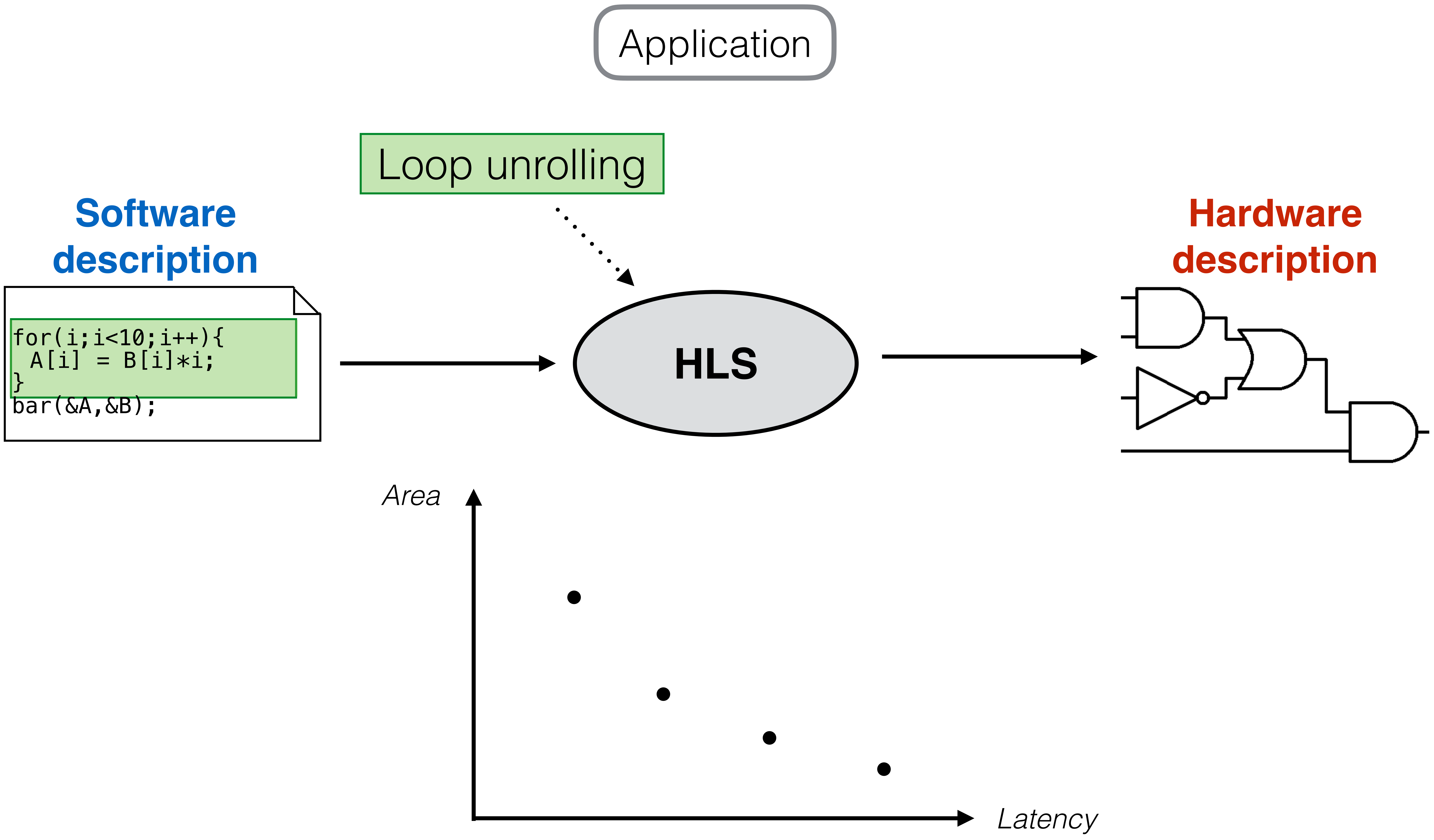
Authors:

Lorenzo Ferretti[1], Jihye Kwon[2], Giovanni Ansaloni[1], Giuseppe Di Guglielmo[2], Luca Carloni[2], Laura Pozzi[1],

[1] Università della Svizzera italiana, Lugano, Switzerland
[2] Columbia University, New York, United States

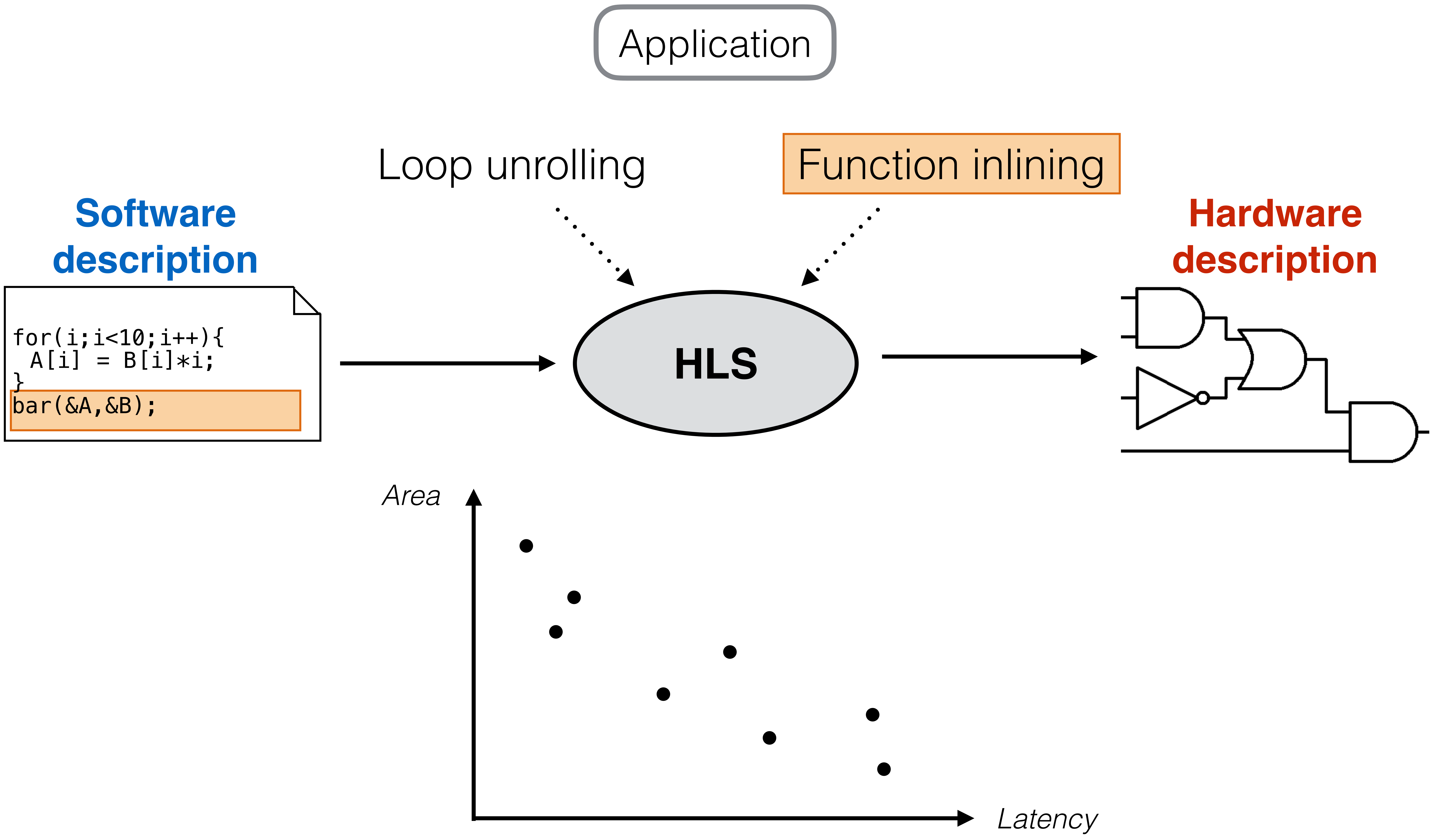
ESWEEK 2020

Motivation



Application
Loop unrolling
Software description
for(i;i<10;i++){
A[i] = B[i]*i;
}
bar(&A,&B);
HLS
Hardware description
Area
Latency

Motivation


Application
Loop unrolling
Function inlining
Software description
for(i;i<10;i++){
A[i] = B[i]*i;
}
bar(&A,&B);
HLS
Hardware description
Area
Latency

Motivation

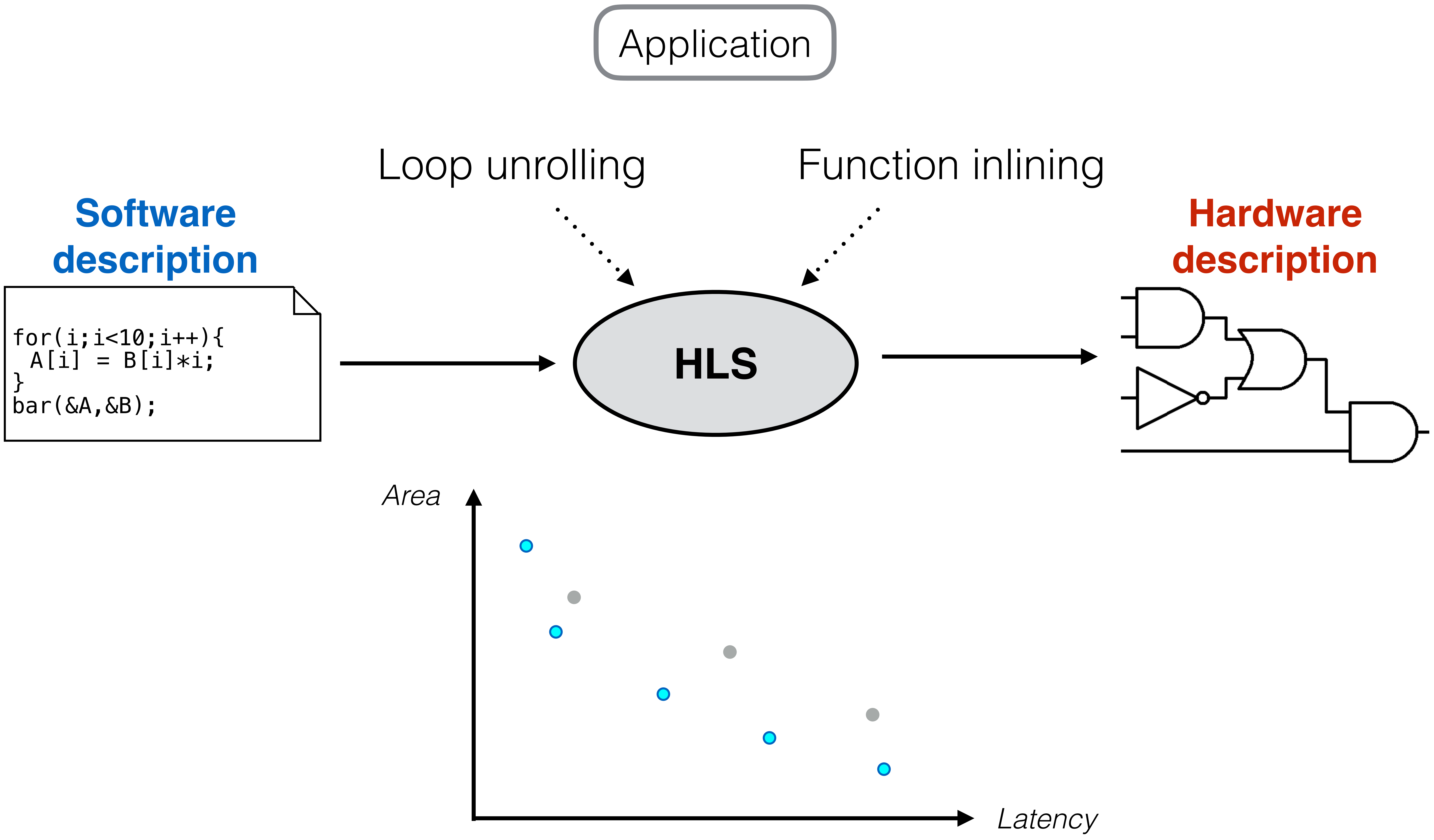
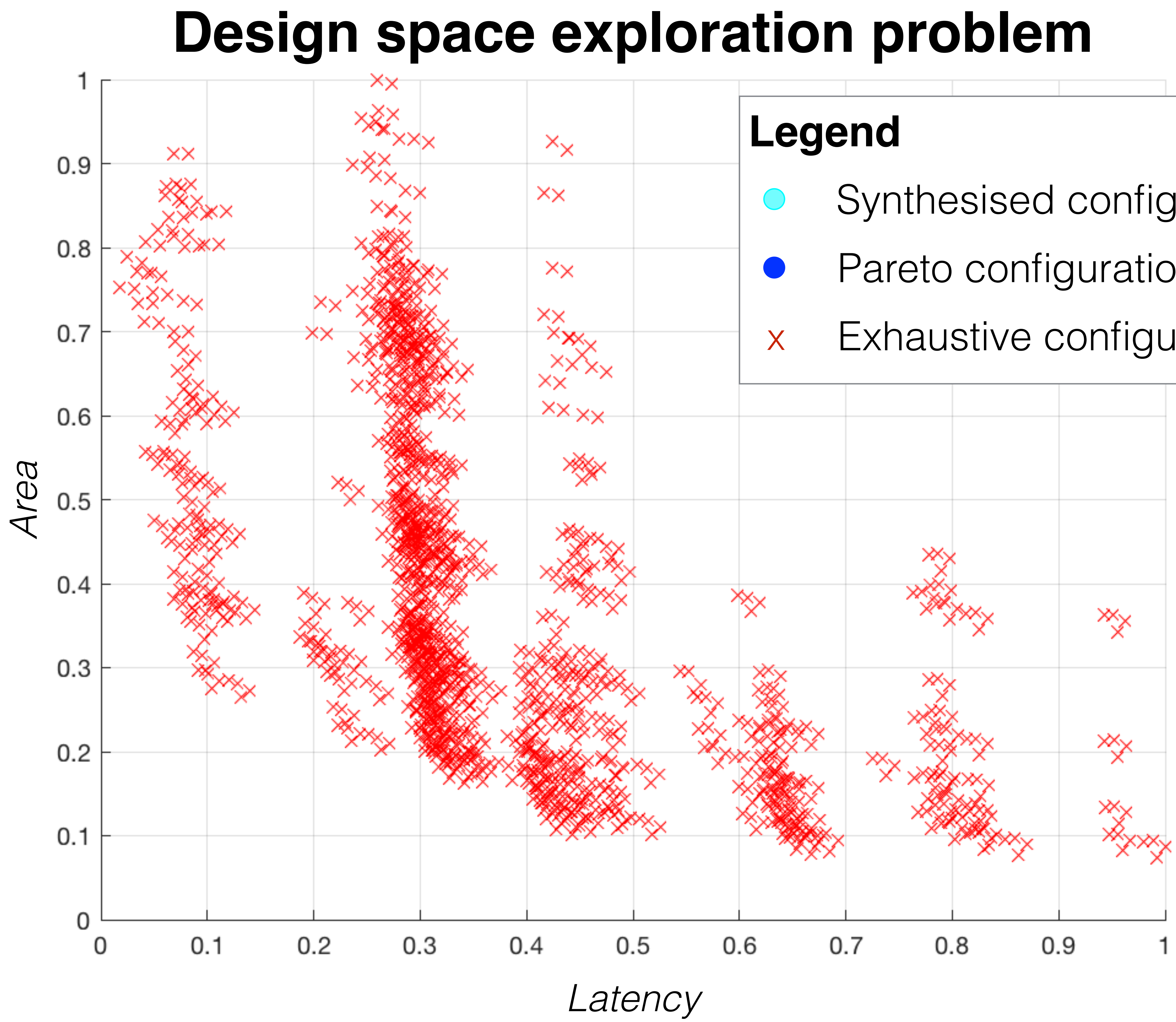

Application
Loop unrolling
Function inlining
Software description
for(i;i<10;i++){
A[i] = B[i]*i;
}
bar(&A,&B);
HLS
Hardware description
Area
Latency

Table of Contents

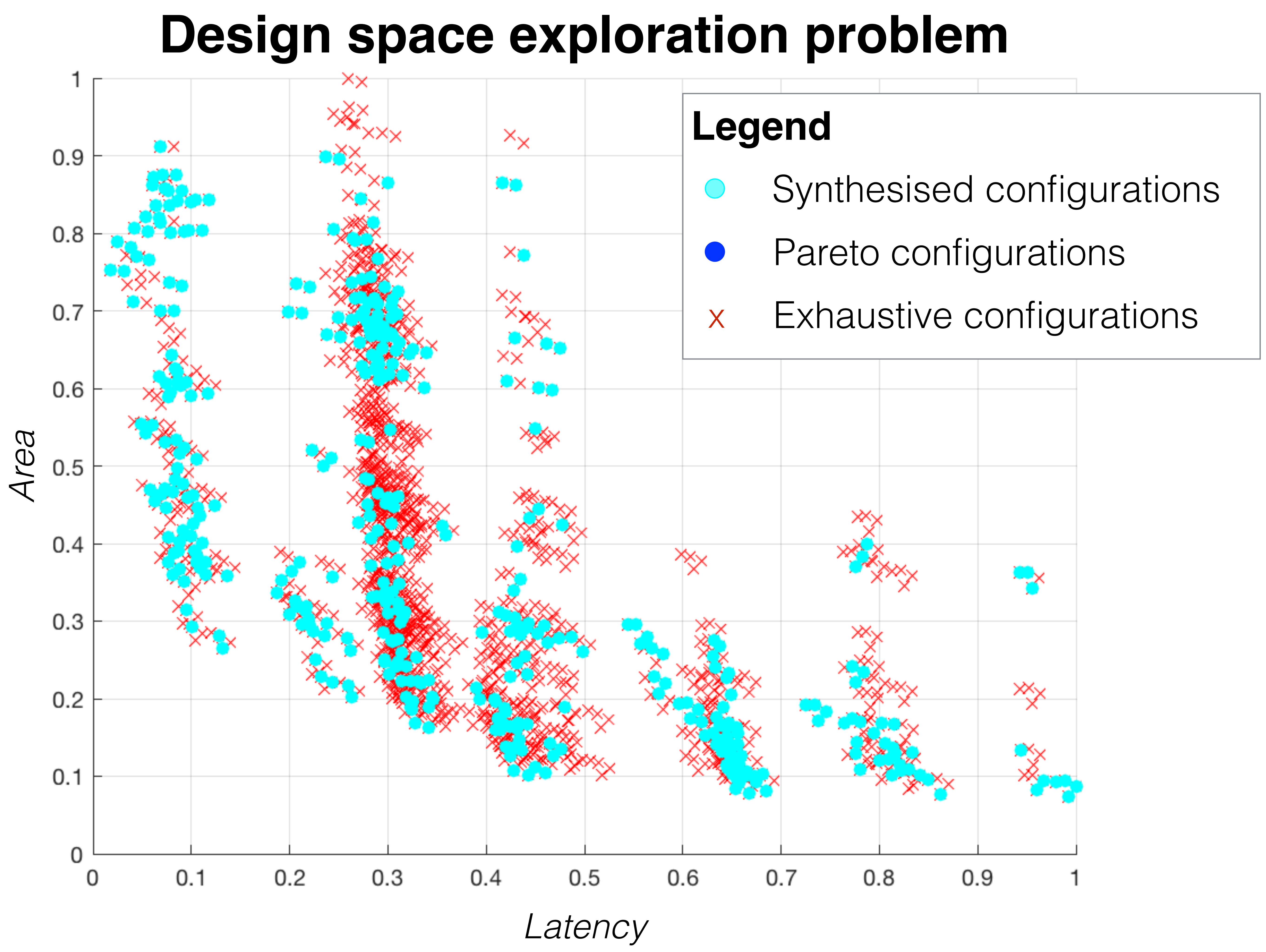
HLS-driven Design Space Exploration (DSE)

Goal: get close to the Pareto solutions while minimising the number of synthesis.



HLS-driven Design Space Exploration (DSE)

Goal: get close to the Pareto solutions while minimising the number of synthesis.



HLS-driven Design Space Exploration (DSE)

Goal: get close to the Pareto solutions while minimising the number of synthesis.

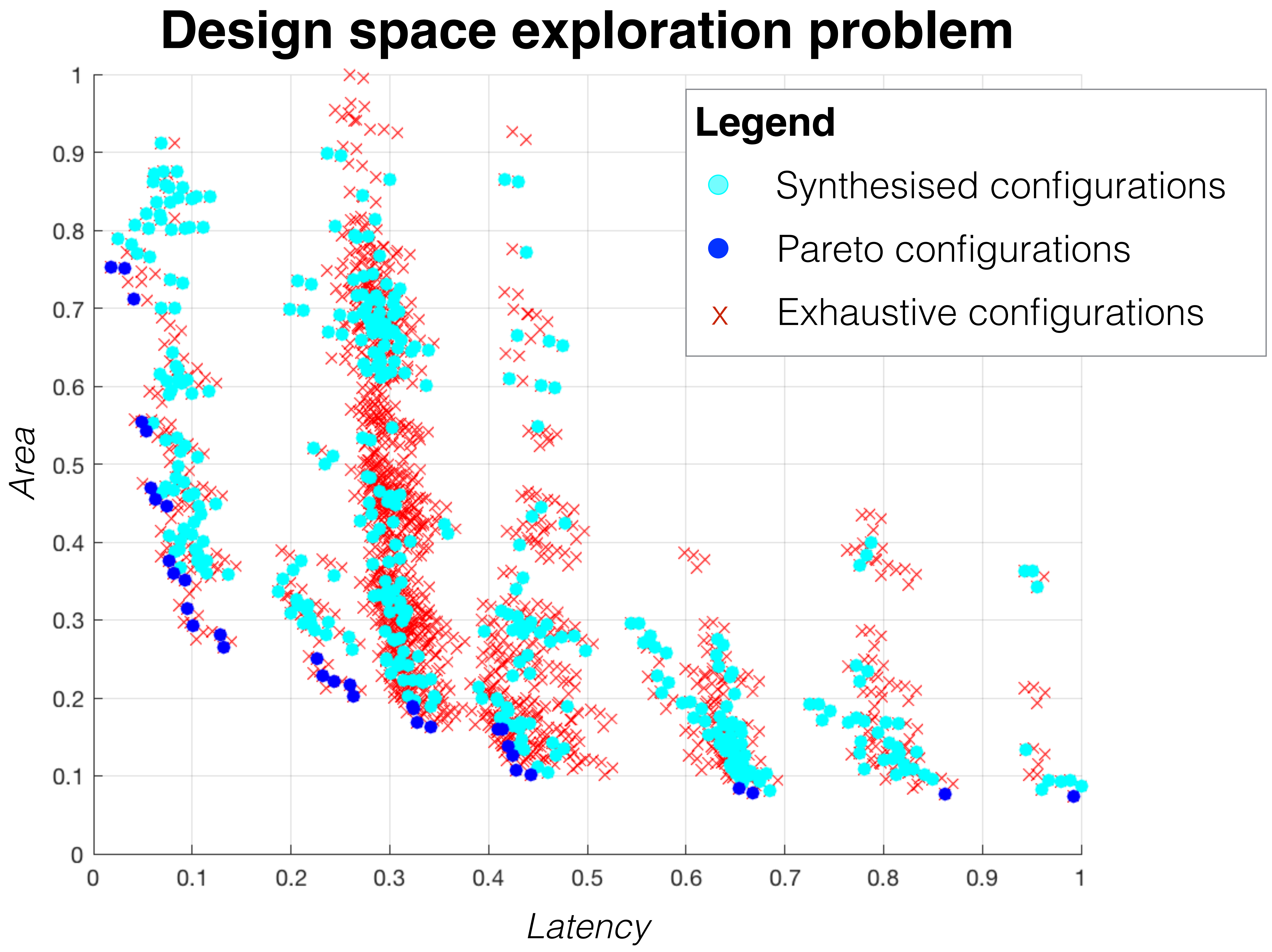
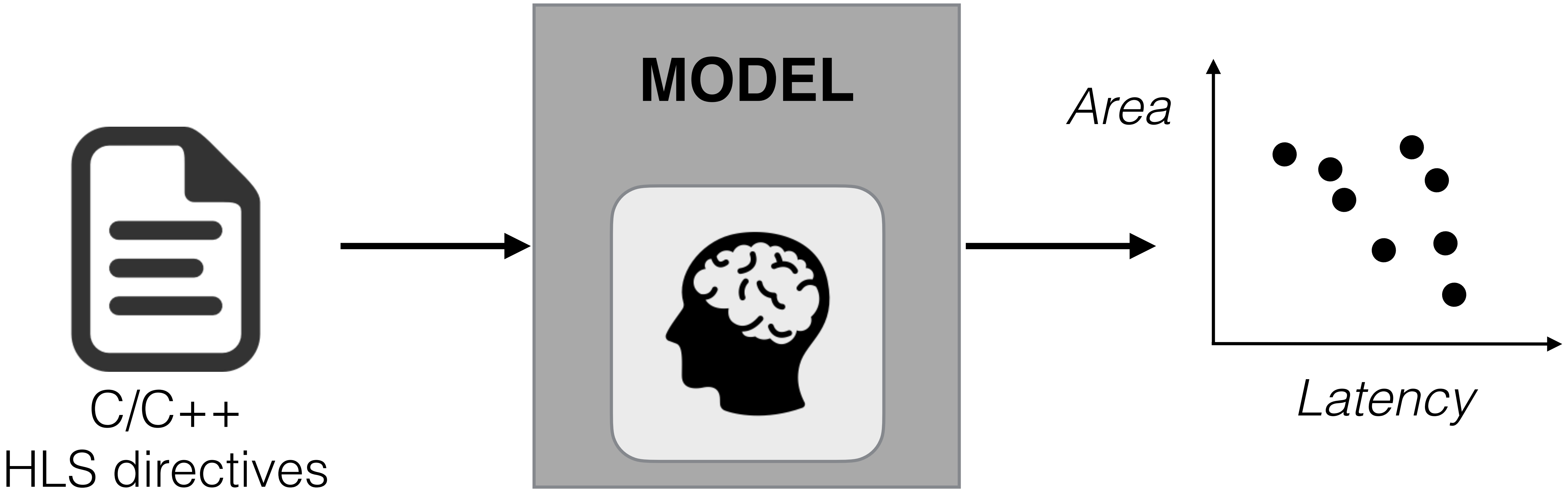


Table of Contents

State of the Art for DSE

Two main approaches:

- Model-based methodologies

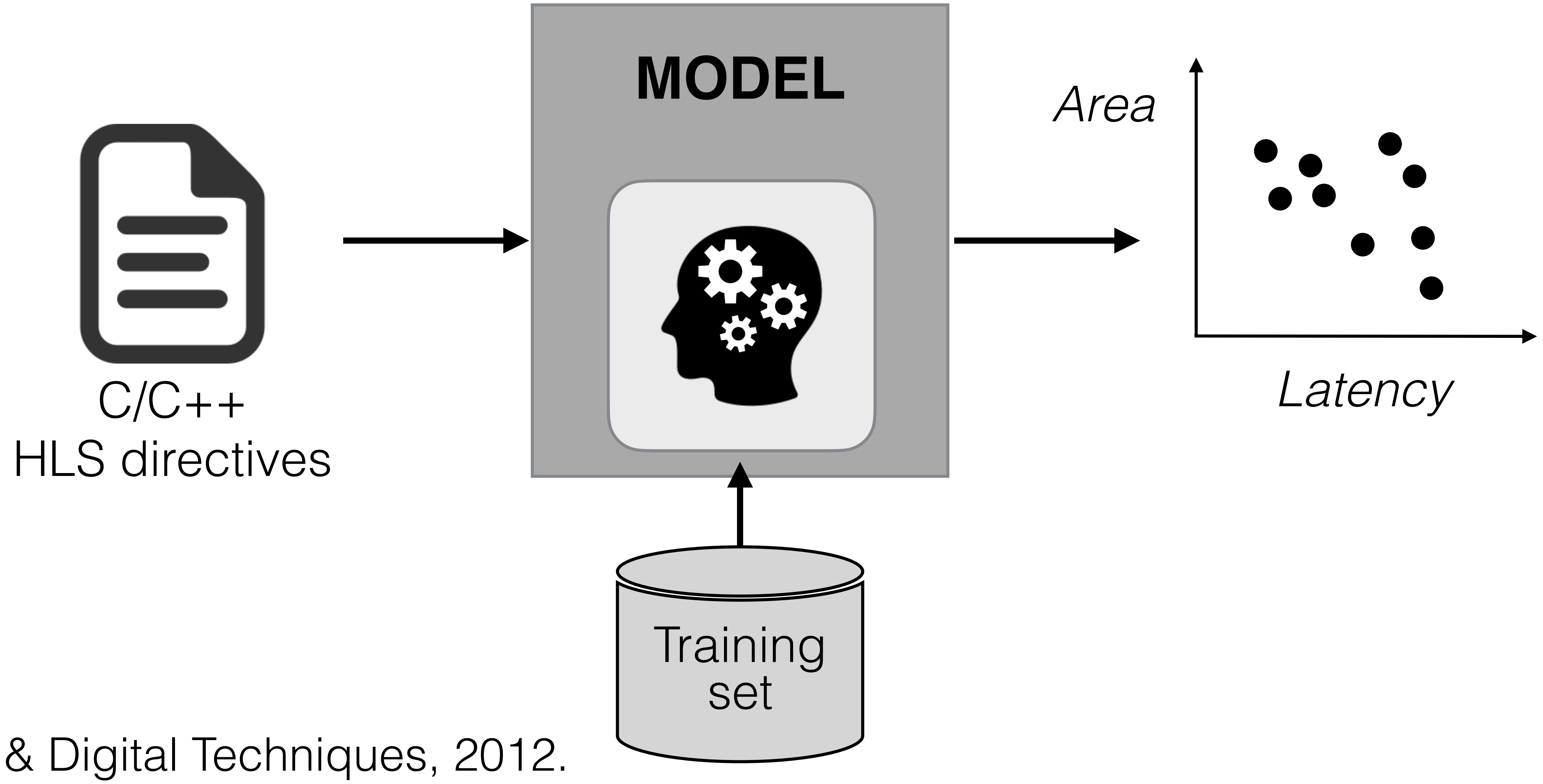


[11] Zhong et al. International Conference on Computer Design, 2014.
[12] N. K. Pham et al. Design, Automation Test in Europe Conference Exhibition, 2015.
[13] Zhong et al. International Conference on Computer Design, 2016.
[14] J. Zhao et al. International Conference on Computer Aided Design, 2017.

State of the Art for DSE

Two main approaches:

- Model-based methodologies
- Black-box-based methodologies
  - Training-based

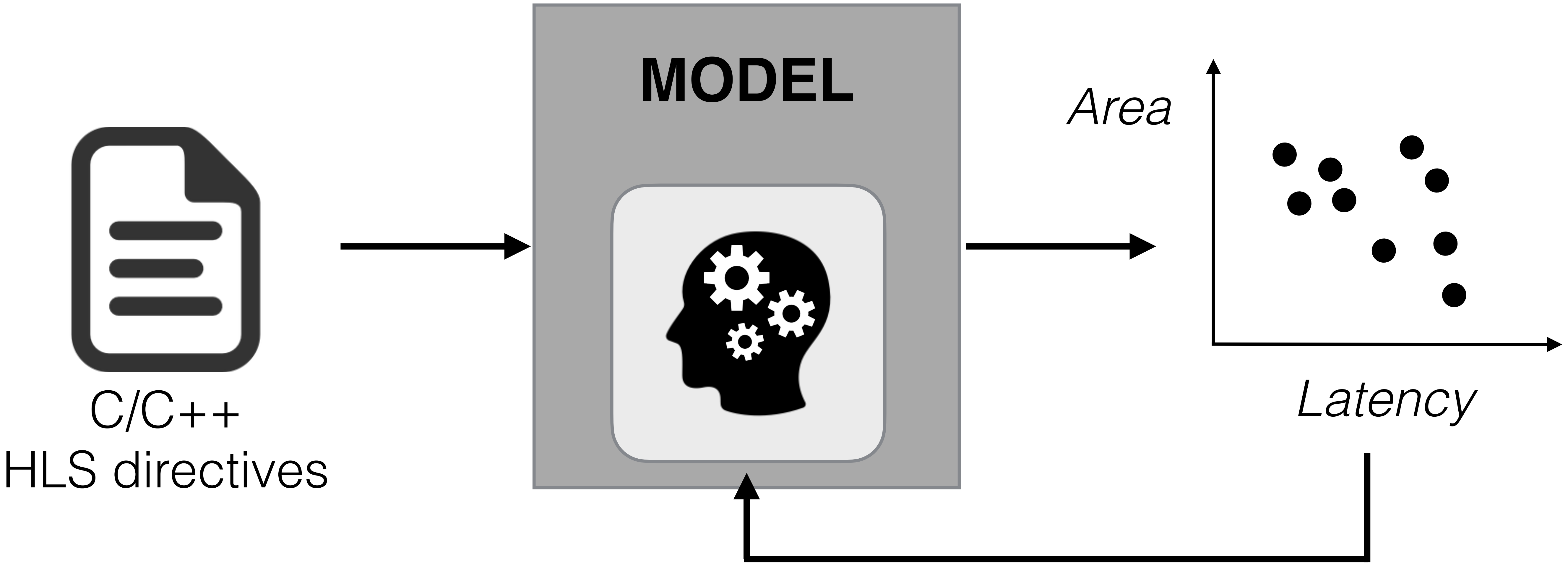


[15] Schafer et al. IET Computers & Digital Techniques, 2012.
[16] A. Mahapatra et al, Electronic System Level Synthesis Conference, 2014

State of the Art for DSE

Two main approaches:

- Model-based methodologies
- Black-box-based methodologies
  - Training-based
  - Refinement-based

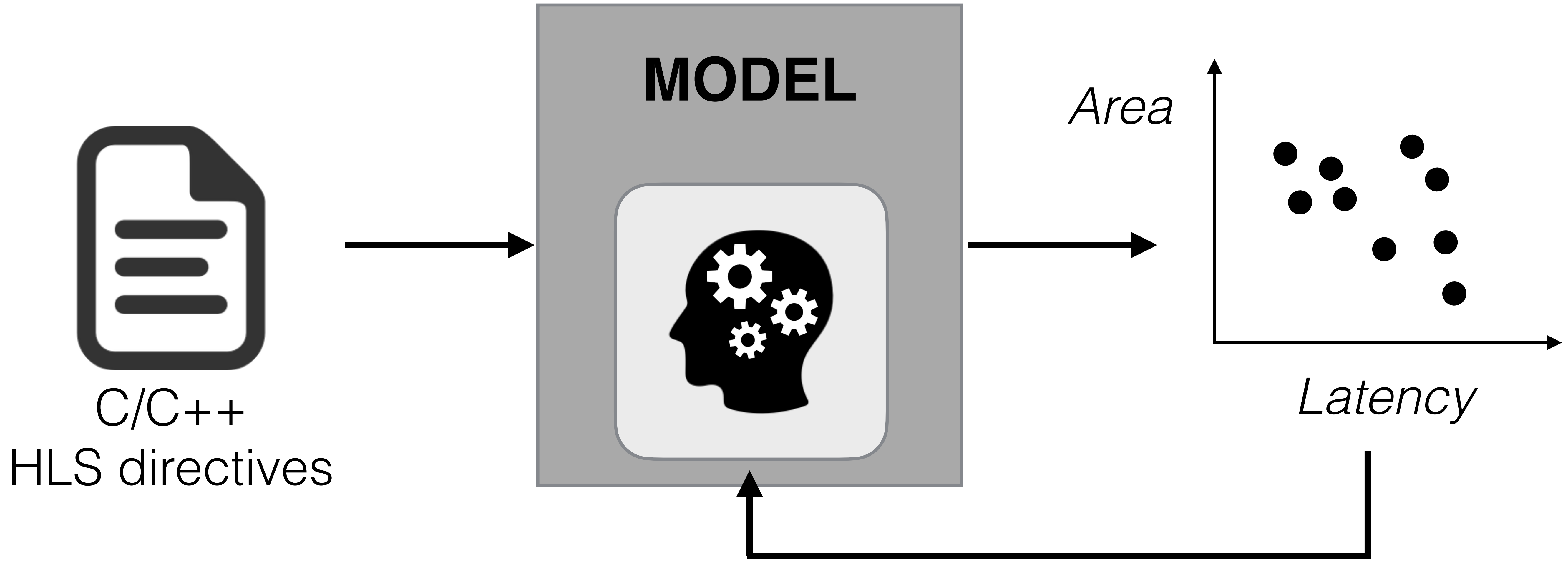


[8] H. Liu et al. Design Automation Conference, 2013.
[9] L. Ferretti et al. IEEE Transactions on Emerging Topics in Computing, 2018.
[10] L. Ferretti et al. International Conference on Computer Design, 2018.
[11] Zhong et al. International Conference on Computer Design, 2014.

State of the Art for DSE

Two main approaches:

- Model-based methodologies
- Black-box-based methodologies
  - Training-based
  - Refinement-based

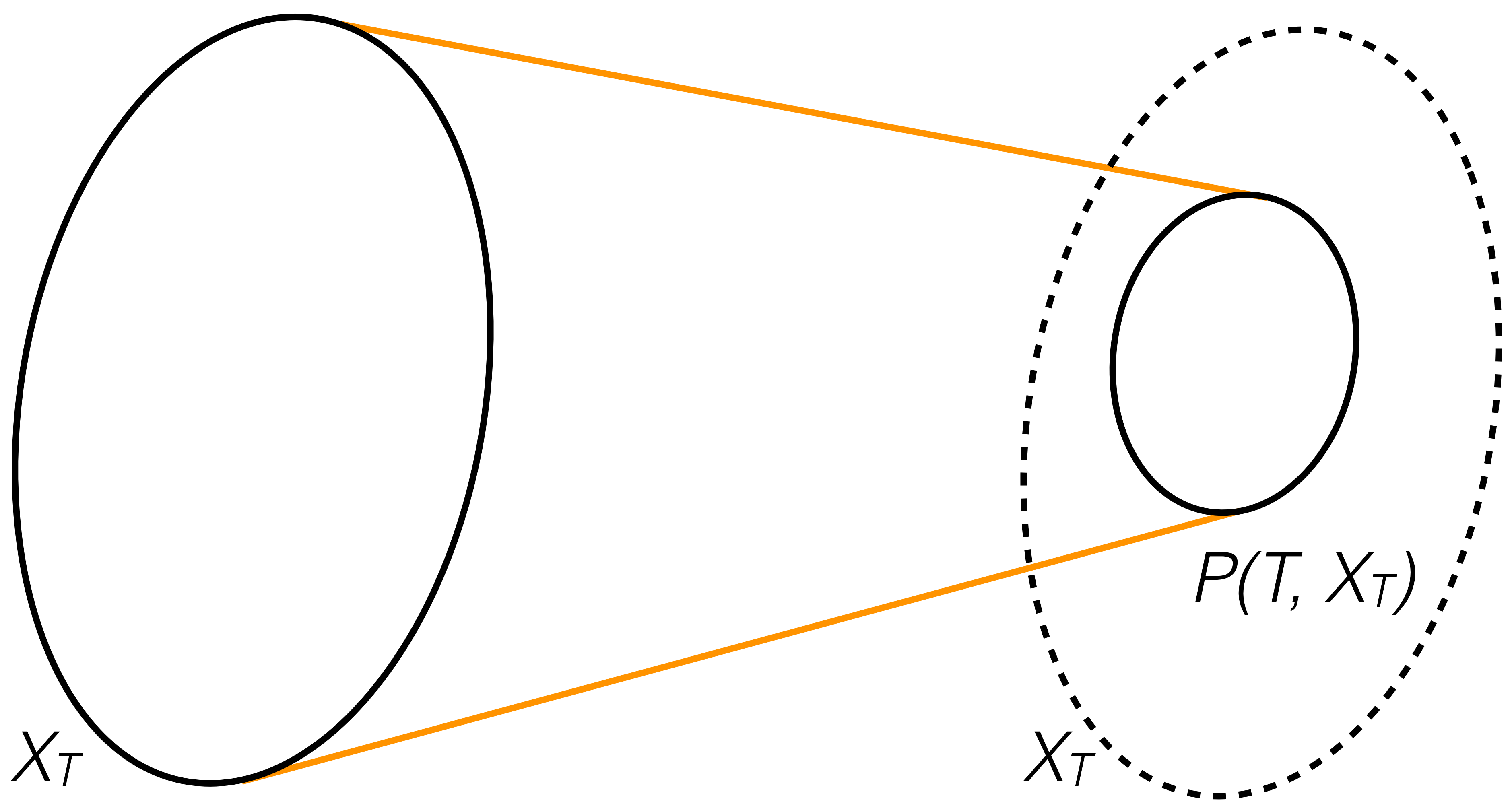


[8] H. Liu et al. Design Automation Conference, 2013.
[9] L. Ferretti et al. IEEE Transactions on Emerging Topics in Computing, 2018.
[10] L. Ferretti et al. International Conference on Computer Design, 2018.
[11] Zhong et al. International Conference on Computer Design, 2014.

Table of Contents

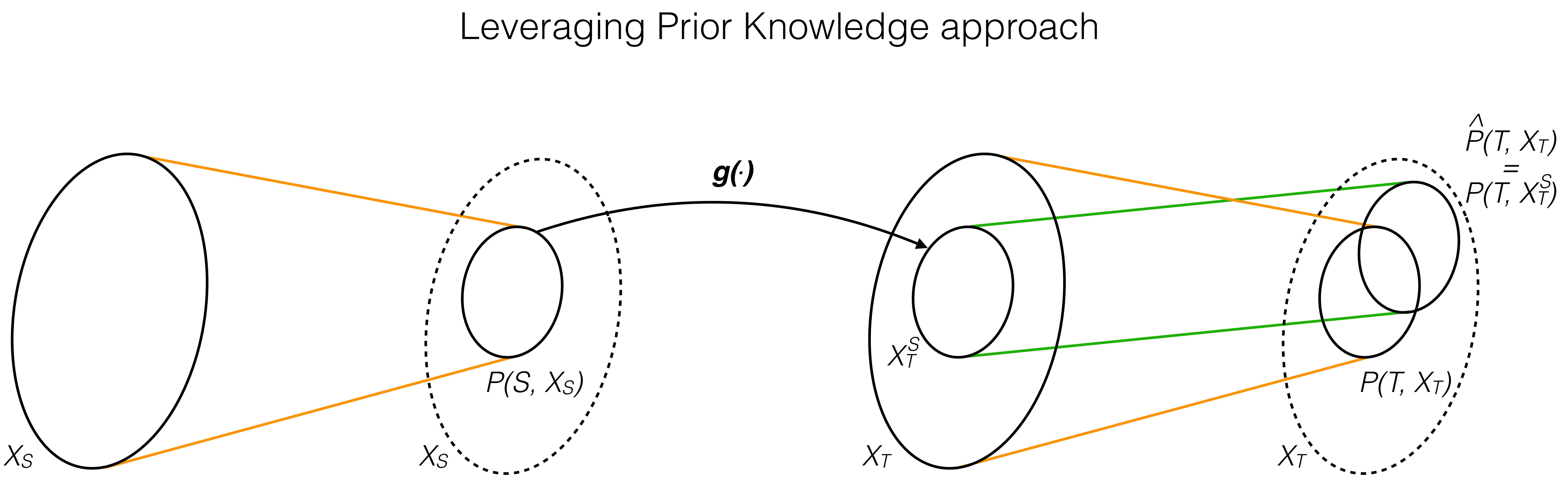
The Idea: Leveraging Prior Knowledge

Standard approach



X_T Set of configurations explored in the DSE of a design T.
$P(T, X_T)$ Set of Pareto-optimal configurations identified with the DSE of T.

The Idea: Leveraging Prior Knowledge



X_S Set of configurations explored in the DSE of a design S.
$P(S, X_S)$ Set of Pareto-optimal configurations identified with the DSE of S.

X_T Set of configurations explored in the DSE of a design T.
$P(T, X_T)$ Set of Pareto-optimal configurations identified with the DSE of T.

Table of Contents

The Methodology

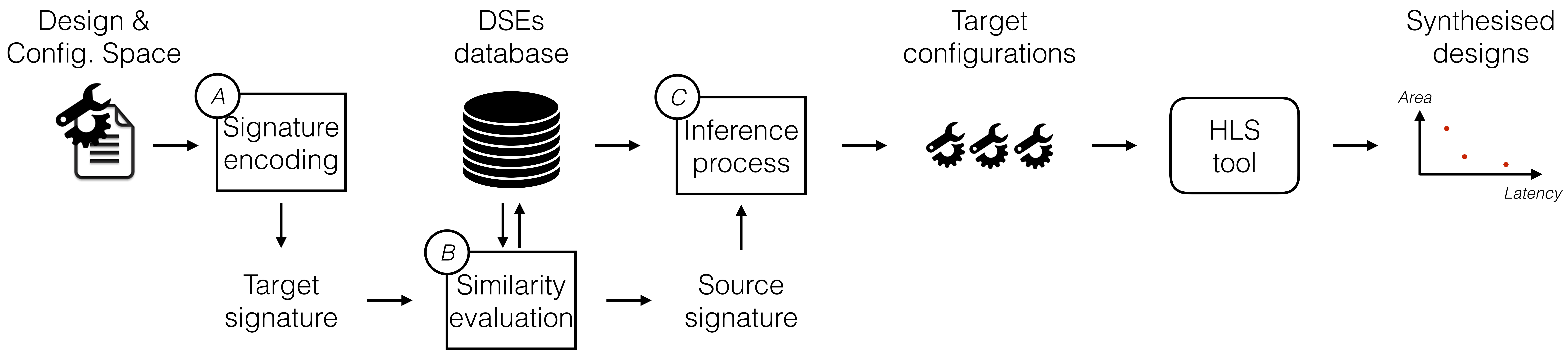


Table of Contents

The Methodology: Signature Encoding

A DSE is characterized with a simplified compact representation that abstracts the specification (code) and the associated configurations (set of applied directives).

Signature encoding : **Specification Encoding** & **Configuration Space Descriptor**

The **Specification Encoding (SE)** is a simplified representation of the original code describing those aspects of an HLS-application that can be targeted by HLS directives.

The **Configuration Space Descriptor (CSD)** describes the user-defined configuration space indicating the set of optimisations type and values considered for the DSE.

The Methodology: Signature Encoding

Specification Encoding (SE): simplified representation of the original code automatically generated trough a compiler pass with LLVM.

Specification Encoding symbols:

- Functions —> **F**
- Function parameter passed by value—> **V**
- Function parameter passed by reference —> **P**
- Arrays definition or declaration—> **A**
- Structs definition or declaration —> **S**
- Loops —> **L**
- Load operations (e.g. a = Array[0]) —> **R**
- Store operations (e.g Array[0] = a) —> **W**
- Function call —> **C#<function_name>#**
- Scope —> **{}**

```
void last_step_scan(int bucket[BUCKETSIZE], int sum[SCAN_RADIX]) {
    int radixID, i, bucket_indx;
    last_1:for (radixID=0; radixID<SCAN_RADIX; radixID++) {
        last_2:for (i=0; i<SCAN_BLOCK; i++) {
            bucket_indx = radixID * SCAN_BLOCK + i;
            bucket[bucket_indx] = bucket[bucket_indx] + sum[radixID];
        }
    }
}
```

↓

LLVM pass

↓

F{PP}L{L{RRW}}

Running Example

The Methodology: Signature Encoding

Configuration Space Descriptor (CSD): a DSL is created to describe the optimisations type and values considered for the DSE. A CSD defines entirely the user-defined configuration space.

Example of CSD:

```
resource;last_step_scan;bucket;{RAM_2P_BRAM}
resource;last_step_scan;sum;{RAM_2P_BRAM}
array_partition;last_step_scan;bucket;1;
   {cyclic,block};{1->512,pow_2}
array_partition;last_step_scan;sum;1;
   {cyclic,block};{1->128,pow_2}
unroll;last_step_scan;last_1;{1->128,pow_2}
unroll;last_step_scan;last_2;{1,2,4,8,16}
clock;{10}
```

```
void last_step_scan(int bucket[BUCKETSIZE], int sum[SCAN_RADIX]) {
   int radixID, i, bucket_indx;
   last_1:for (radixID=0; radixID<SCAN_RADIX; radixID++) {
      last_2:for (i=0; i<SCAN_BLOCK; i++) {
         bucket_indx = radixID * SCAN_BLOCK + i;
         bucket[bucket_indx] = bucket[bucket_indx] + sum[radixID];
      }
   }
}
```

Directive type | **Location** | **Set of directive values** → **Knob**

Running Example

Table of Contents

The Methodology: Similarity Evaluation

Similarity Evaluation: in order to identify the proper source for the inference process the similarity among a target DSE and the available source is calculated.

The similarity function (*Sim*) is given by a linear combination of Signature Encoding similarity (Sim_{SE}) and Configuration Space Descriptor similarity (Sim_{CSD}).

$$Sim = \alpha Sim_{SE} + (1 - \alpha) Sim_{CSD} \quad \alpha \in [0,1]$$

$$Sim_{SE} = LCS(SE_T, SE_S)$$

$$Sim_{CSD} = 1 - \left[\frac{1}{I} \sum_{i=1}^{I} \Delta(K_i, M_{T,S}(K_i)) / D_{MAX} \right]$$

$$\Delta(K_i, K_j) = \sqrt{\sum_{n=1}^{|K_i|} (\min_{m=1}^{|K_j|} |\delta(k_n, k_m)|)^2} \quad k_n \in K_i, k_m \in K_j \qquad \delta(k_n, k_m) = \sqrt{\sum_{z=1}^{Z} |k_{n,z}, k_{m,z}|^2}$$

The Methodology: Similarity Evaluation

CSD similarity: measures the similarity among knobs of target and source CSDs.

A top-down mapping maps knobs of the source Signature Encoding to knobs of the target one.

SE_{Source} F{PPP}L{L{RRW}}

SE_{Target} F{PP}L{L{RRW}}

CSD_{Source}

```
array_partition;get_delta_matrix_weights2;delta_weights2;1;
   {cyclic,block};{1->256,pow_2}
array_partition;get_delta_matrix_weights2;output_difference;
   1;{cyclic,block};{1->64,pow_2}
array_partition;get_delta_matrix_weights2;last_activations;
   1;{cyclic,block};{1->64,pow_2}
unroll;get_delta_matrix_weights2;loop_1;{1->64,pow_2}
unroll;get_delta_matrix_weights2;loop_2;{1->64,pow_2}
clock;{10}
```

CSD_{Target}

```
resource;last_step_scan;bucket;{RAM_2P_BRAM}
resource;last_step_scan;sum;{RAM_2P_BRAM}
array_partition;last_step_scan;bucket;1;
   {cyclic,block};{1->512,pow_2}
array_partition;last_step_scan;sum;1;
   {cyclic,block};{1->128,pow_2}
unroll;last_step_scan;last_1;{1->128,pow_2}
unroll;last_step_scan;last_2;{1,2,4,8,16}
clock;{10}
```

Running Example

The Methodology: Similarity Evaluation

CSD similarity: measures the similarity among knobs of target and source CSDs.

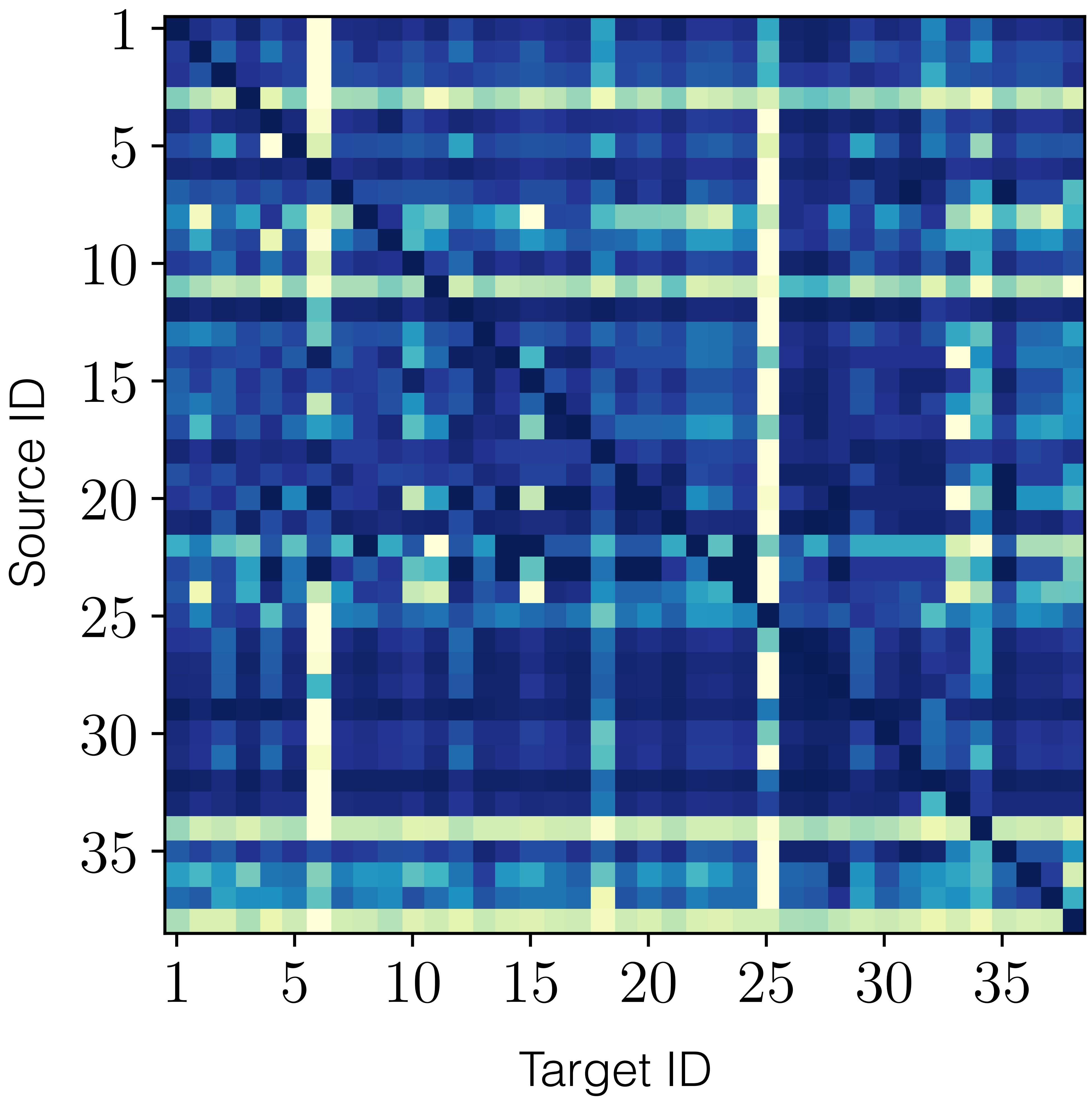
Domain	Mapping							
Target knobs	K_1 Resource	K_2 Resource	K_3 Part. type Part. factor	K_4 Part. type Part. factor		K_5 Unroll	K_6 Unroll	K_7 Clock
Source knobs			K_1 Part. type Part. factor	K_2 Part. type Part. factor	K_3 Part. type Part. factor	K_4 Unroll	K_5 Unroll	K_6 Clock

Domain	Knob values								
Set of values for target knob K_6	1	2	4	6	8	16	32	64	128
Set of values for source knob K_5	1	2	4	6	8	16	32	64	

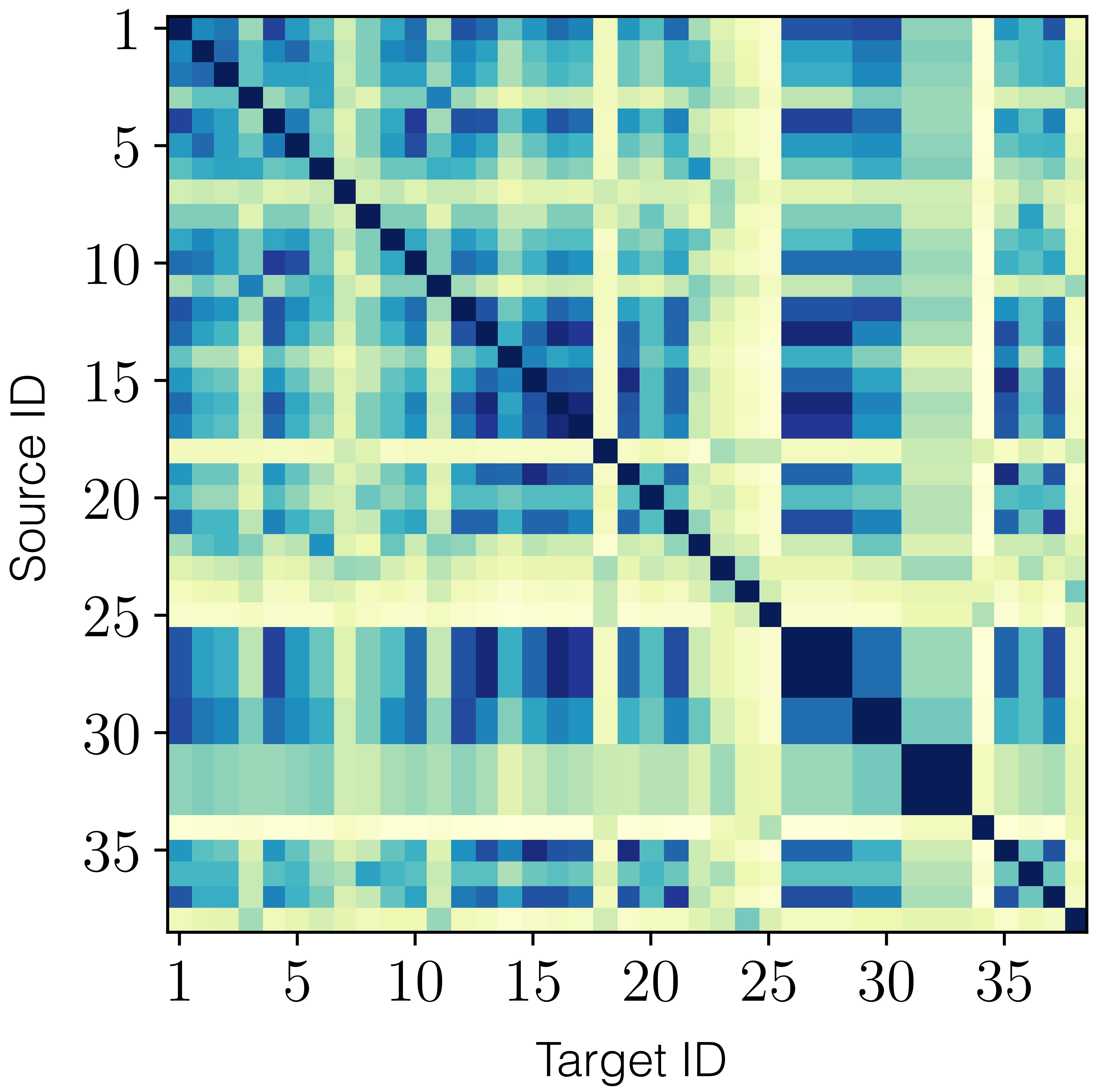
Running Example

The Methodology: Similarity Evaluation

Sim_{SE}



Sim_{CSD}



The Methodology: Similarity Evaluation

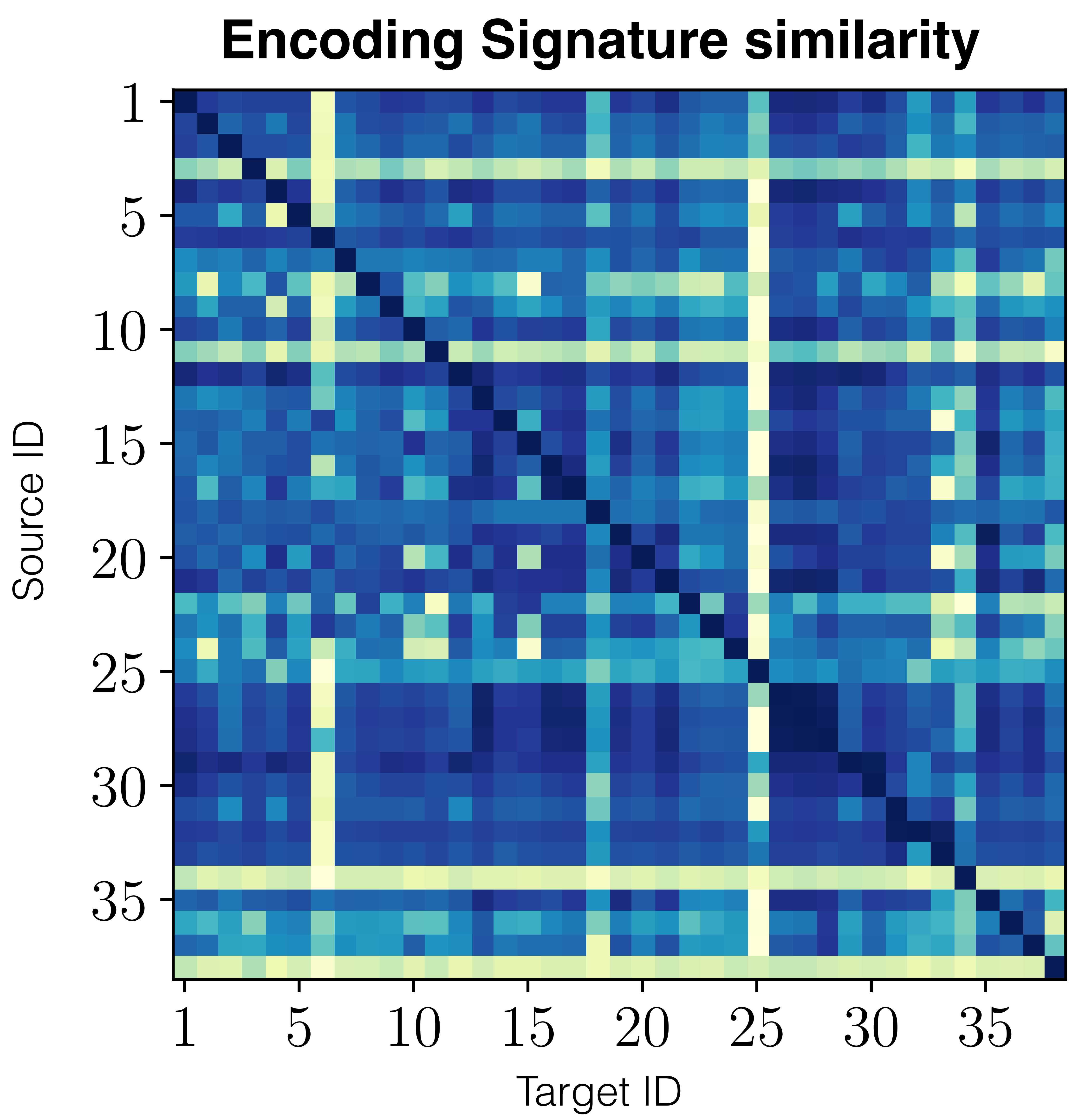
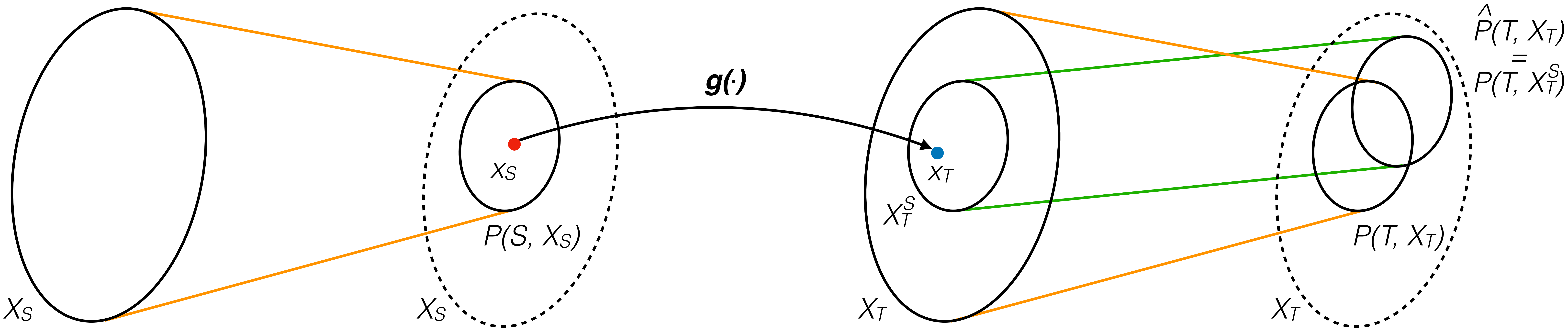


Table of Contents

The methodology: Inference Process

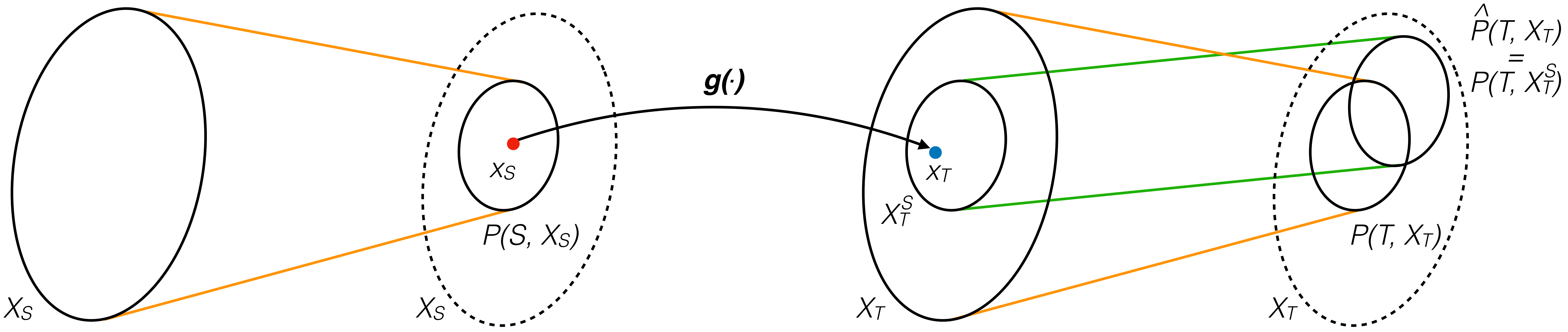


$$x_S = [x_S^1, \ldots, x_S^J] \in X_S \qquad x_T = [x_T^1, \ldots, x_T^I] \in X_T$$

$$x_T^i = \underset{n}{argmin}\{\delta(k_n, x_S^j)\}$$

$$\delta(k_n, k_m) = \sqrt{\sum_{z=1}^{Z} |k_{n,z}, k_{m,z}|^2}$$

The methodology: Inference Process

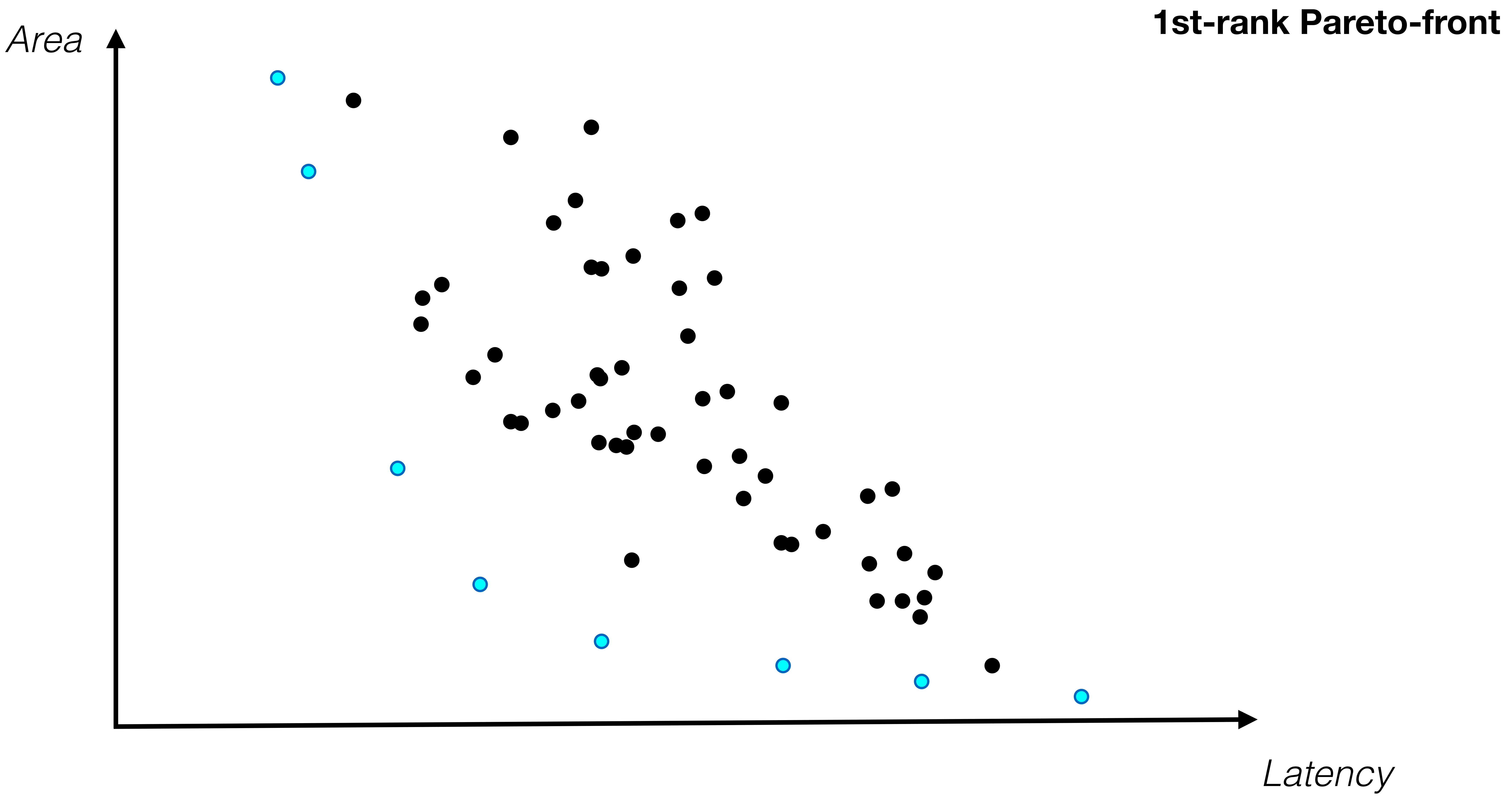


Domain	Inference						
Source knobs			K_1 cyclic 256	K_2 cyclic 8	K_4 32	K_5 64	K_6 10
Target knobs	K_1 2P_BRAM	K_2 2P_BRAM	K_3 cyclic, block 1,…,256,512	K_4 cyclic, block 1,..,8,..,128	K_5 1,..,32,..,128	K_6 1,2,4,8,16	K_7 10

Running Example

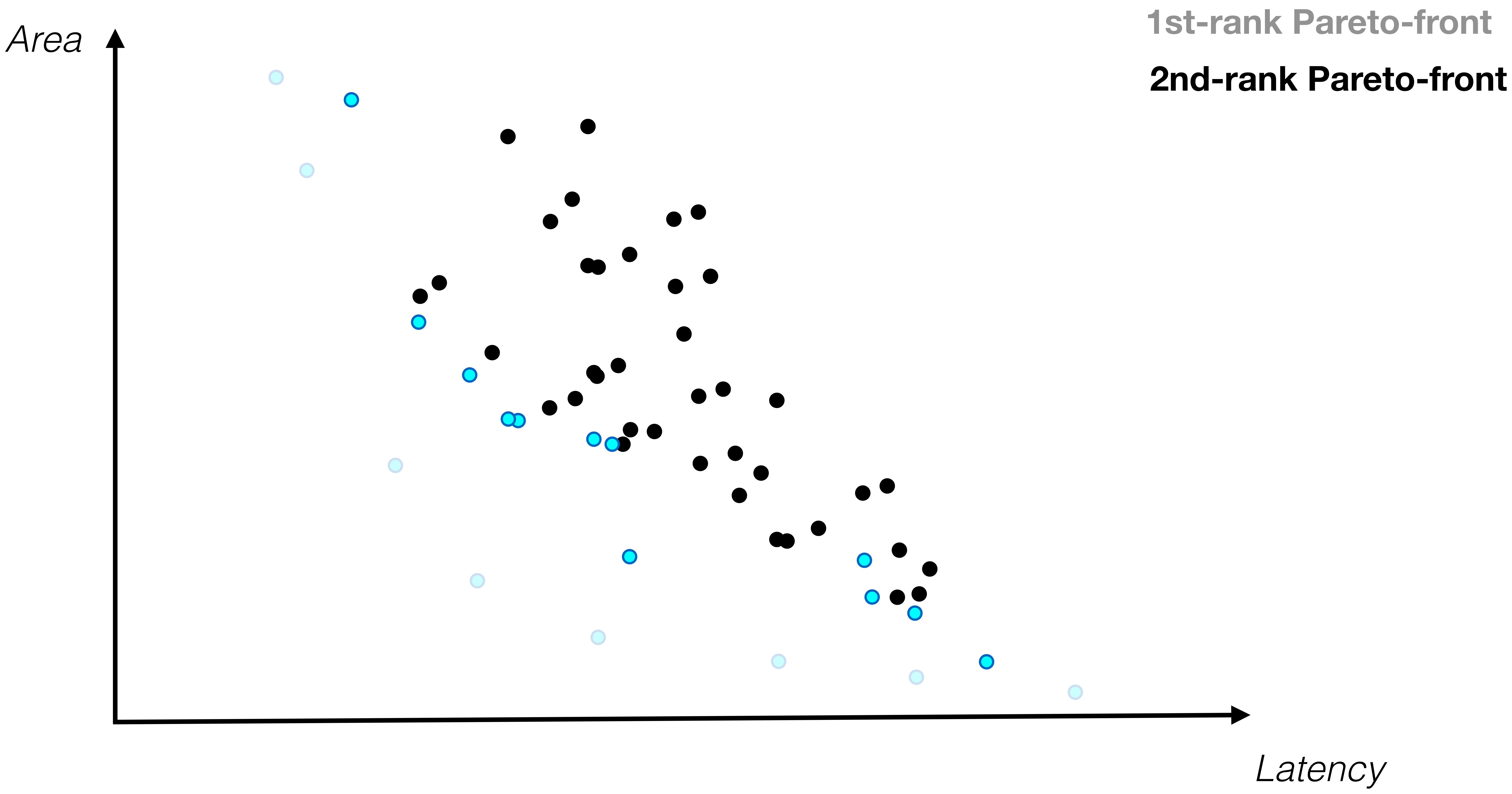
The methodology: Inference Process

The source design space is iteratively peeled and lower-rank Pareto frontiers are used for the inference.



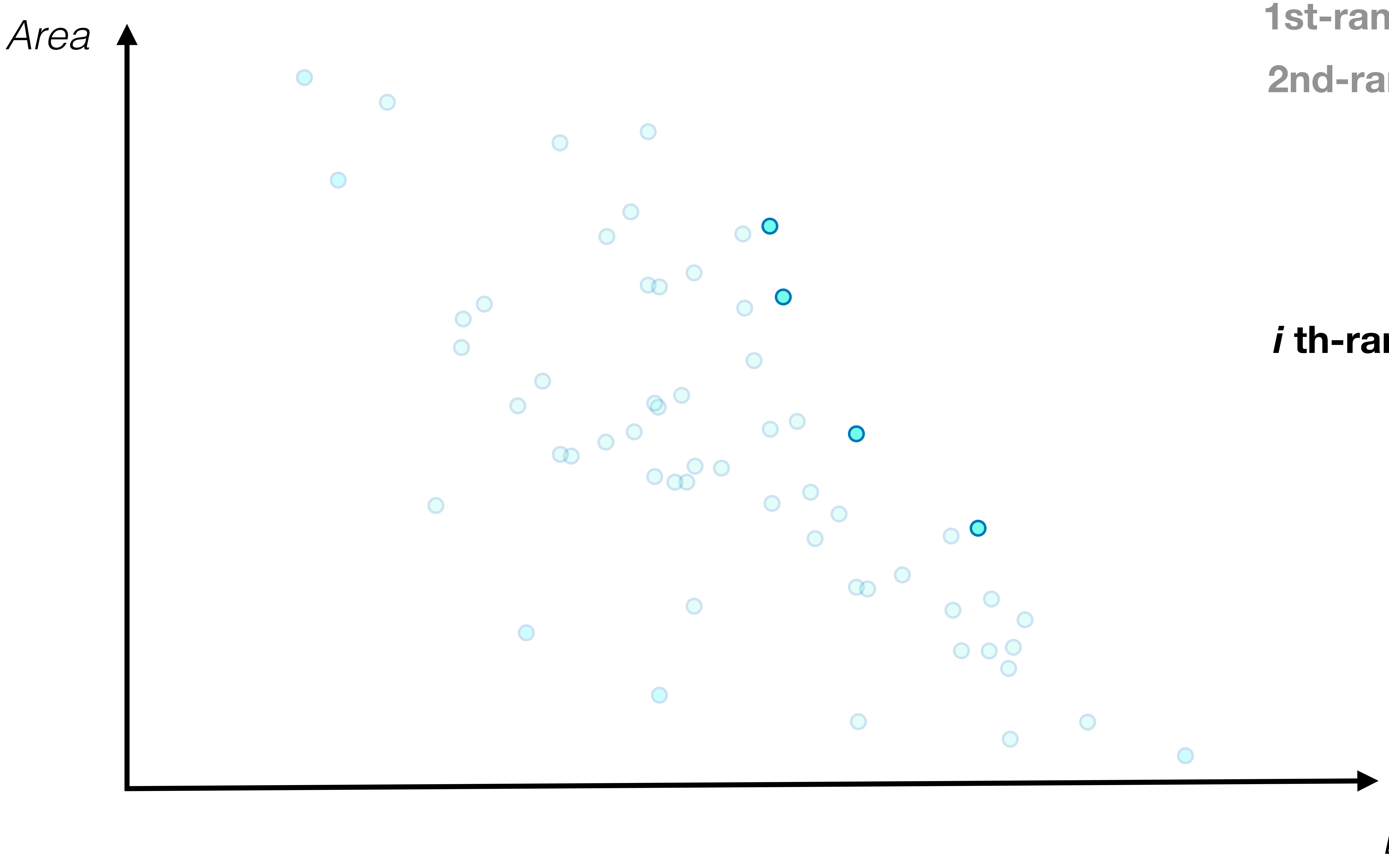
The methodology: Inference Process

The source design space is iteratively peeled and lower-rank Pareto frontiers are used for the inference.



The methodology: Inference Process

The source design space is iteratively peeled and lower-rank Pareto frontiers are used for the inference.



1st-rank Pareto-front

2nd-rank Pareto-front

.
.
.
.

***i* th-rank Pareto-front**

Table of Contents

Results

We have considered 39 out of 50 possible design from Machsuite[4].

For each of them we have performed an exhaustive exploration and used it as a ground-truth to evaluate the quality of the DSE.

We have used Average Distance from Reference Set (ADRS) metric to measure the distance among the retrieved Pareto frontier and the ground-truth.

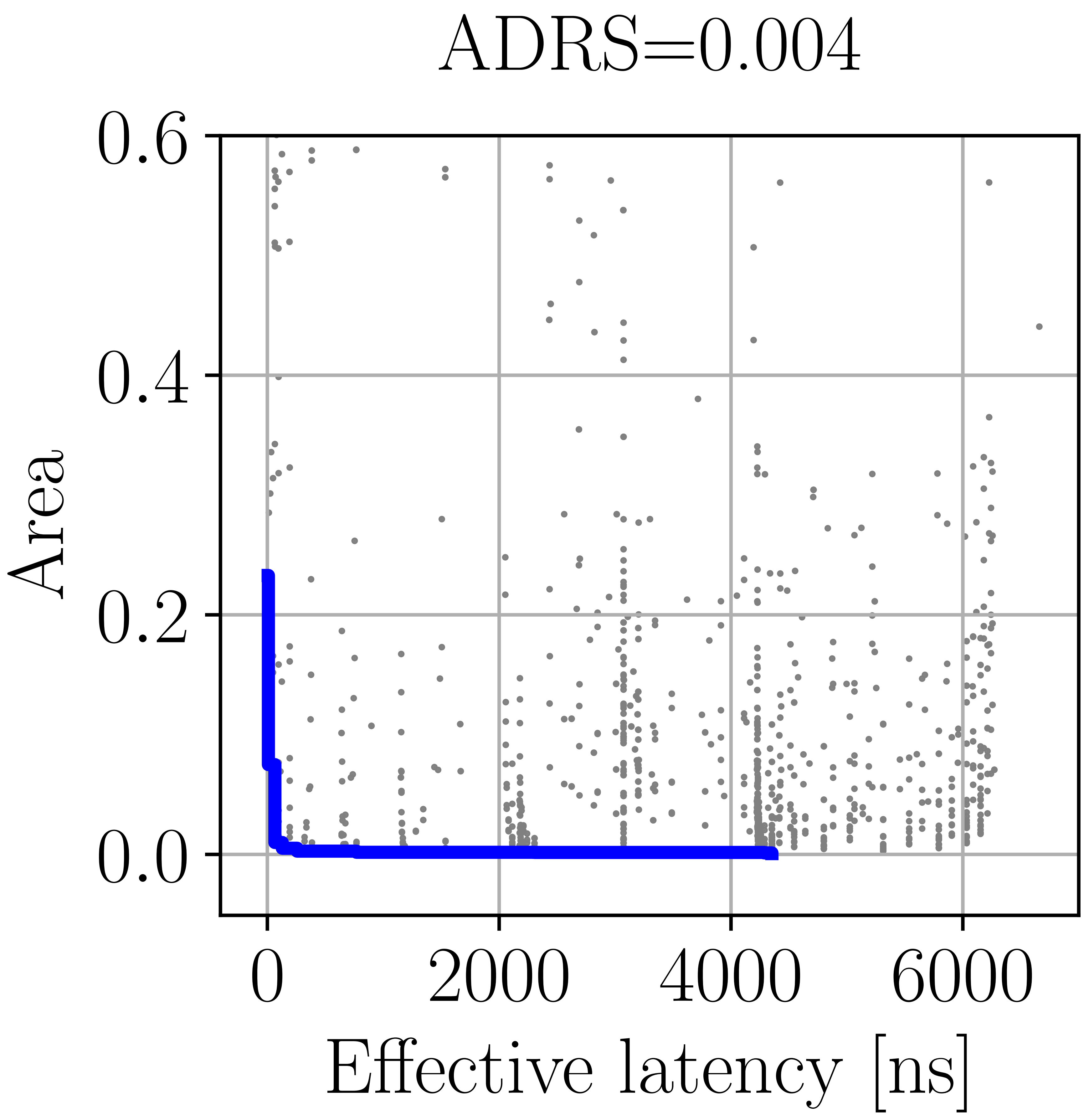
$$ADRS(\bar{P}, P) = \left[\frac{1}{|P|} \sum_{p \in P} \min_{\bar{p} \in \bar{P}} (d(\bar{p}, p)) \right]$$

$$d(\bar{p}, p) = \max\{0, (A_{\bar{p}} - A_p)/A_p, (L_{\bar{p}} - L_p)/L_p\}$$

[4] B. Reagen et al. International Symposium on Workload Characterisation, 2014.

Results

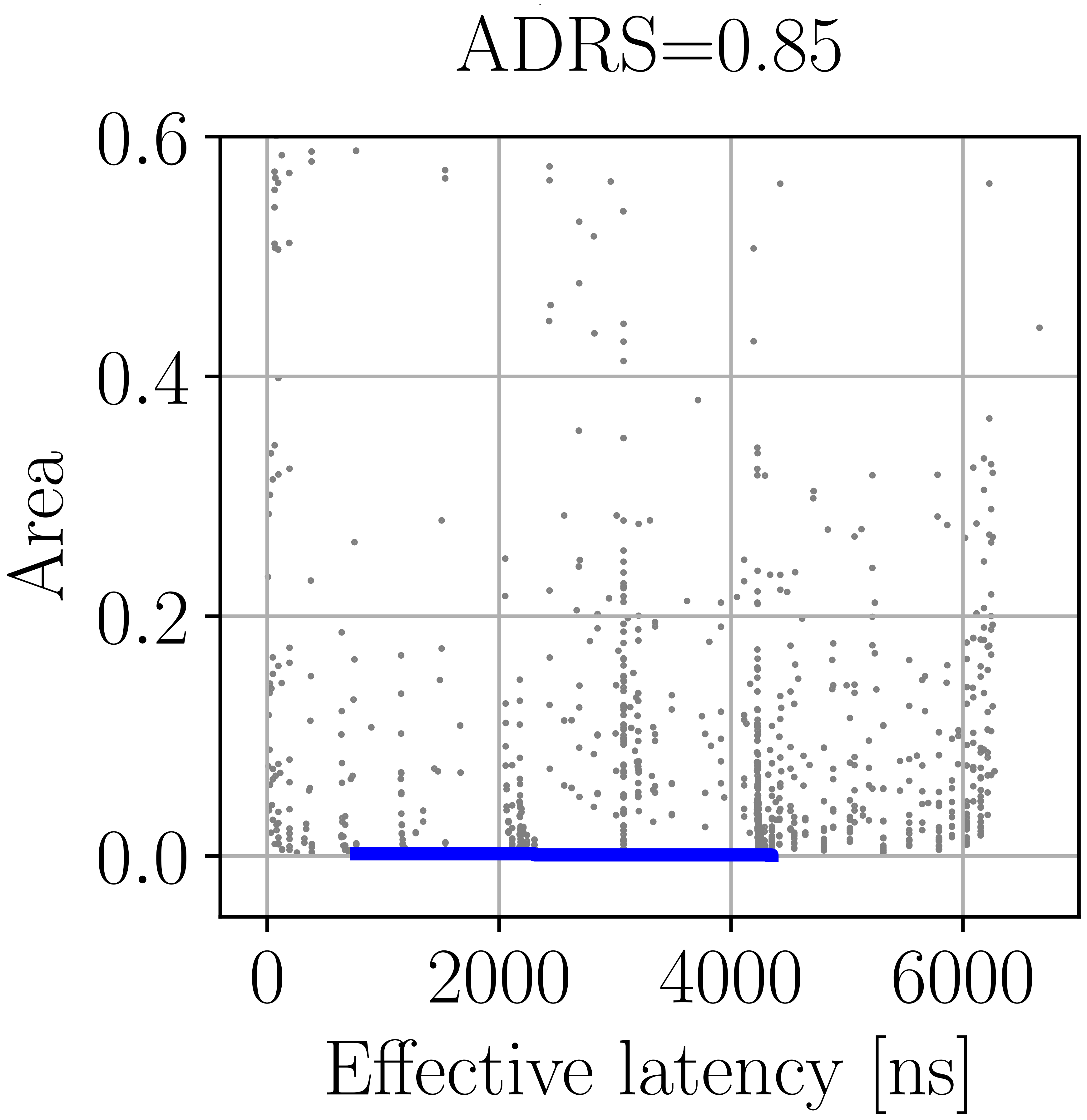
Effectiveness of the similarity metric: 1st-ranked source.



- ✓ High Specification Encoding similarity
- ✓ High Configuration Space Descriptor similarity

Results

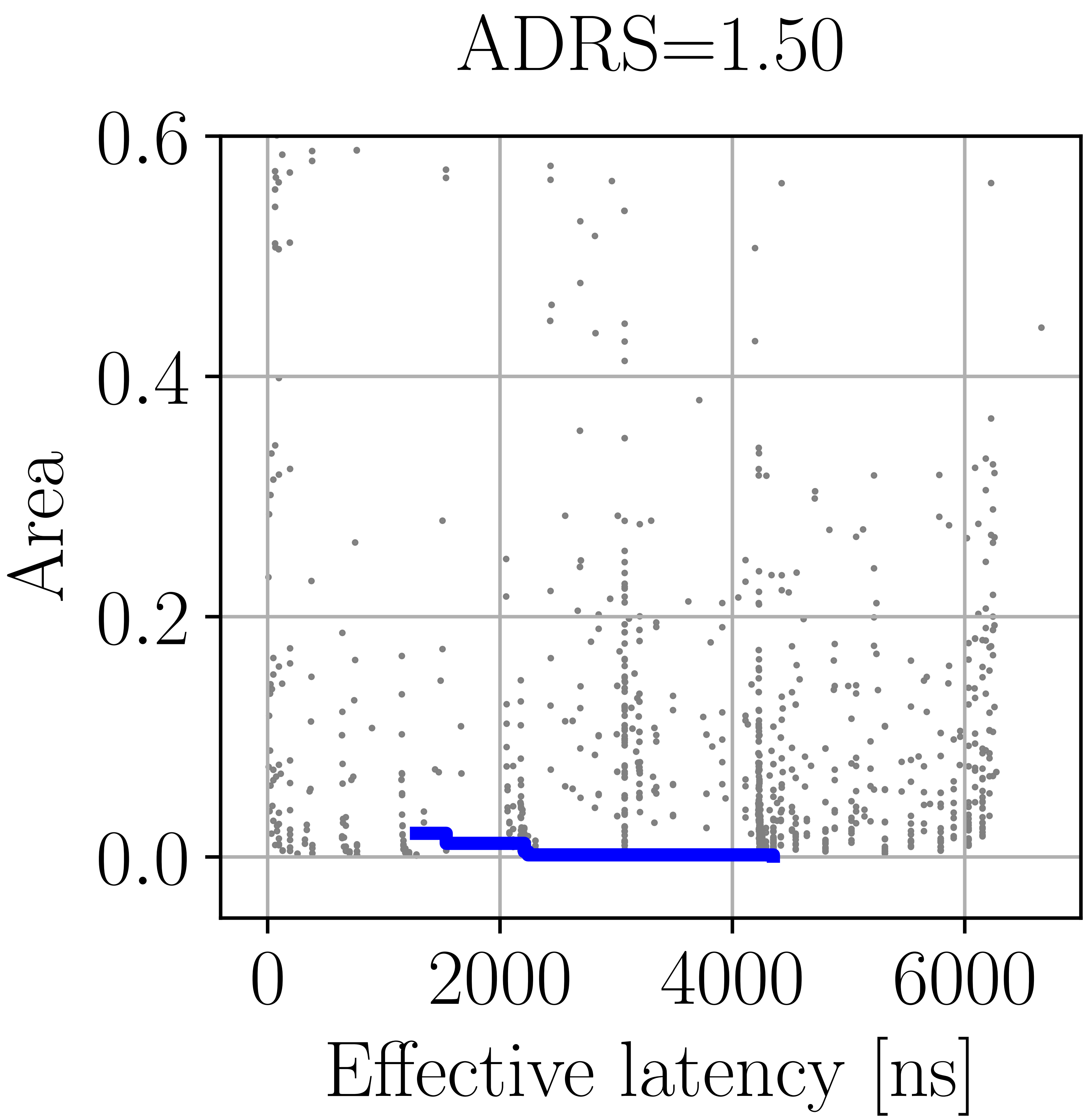
Effectiveness of the similarity metric: 30th-ranked source.



- X Low Specification Encoding similarity
- ✓ High Configuration Space Descriptor similarity

Results

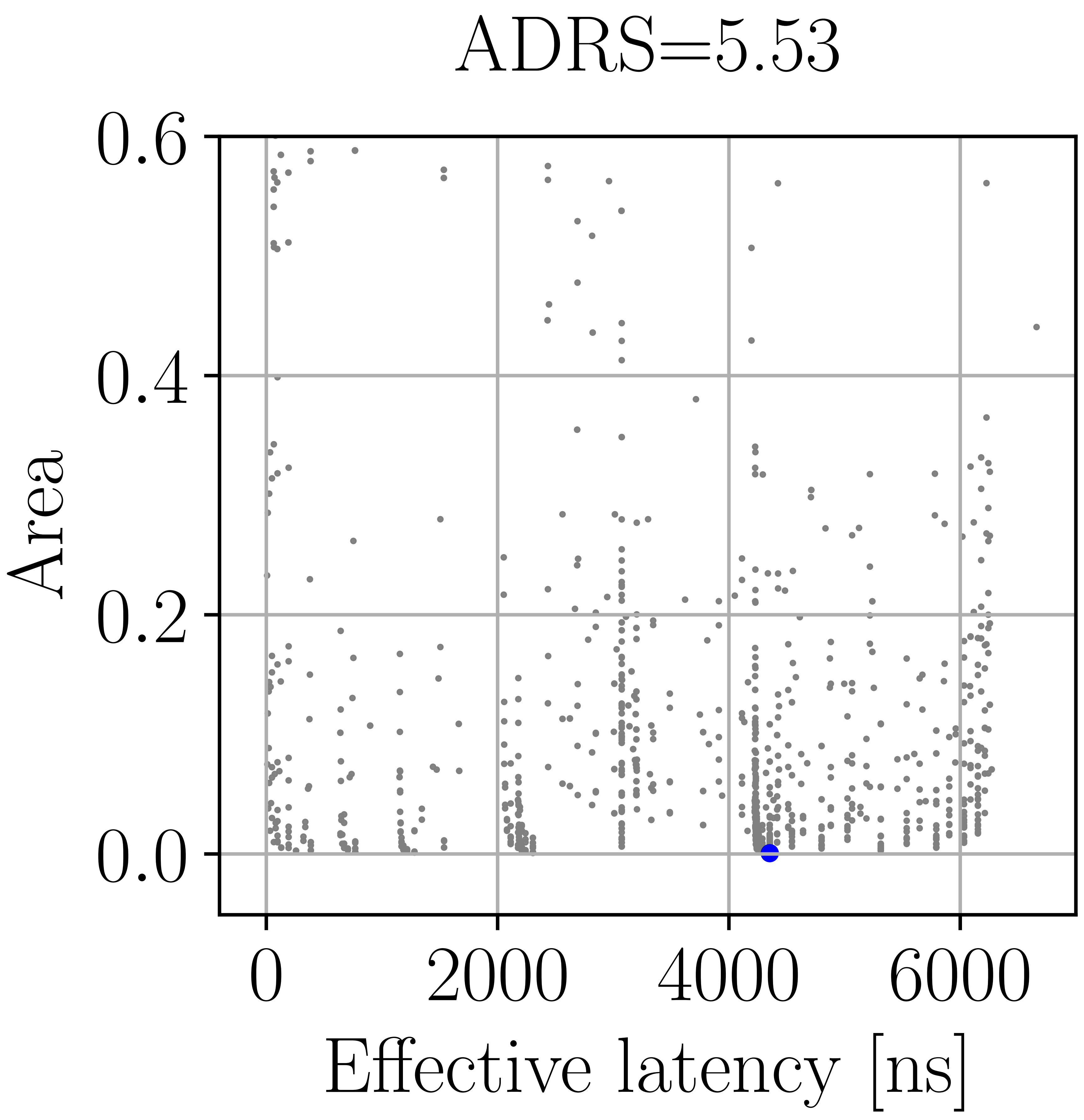
Effectiveness of the similarity metric: 35th-ranked source.



- ✓ High Specification Encoding similarity
- X Low Configuration Space Descriptor similarity

Results

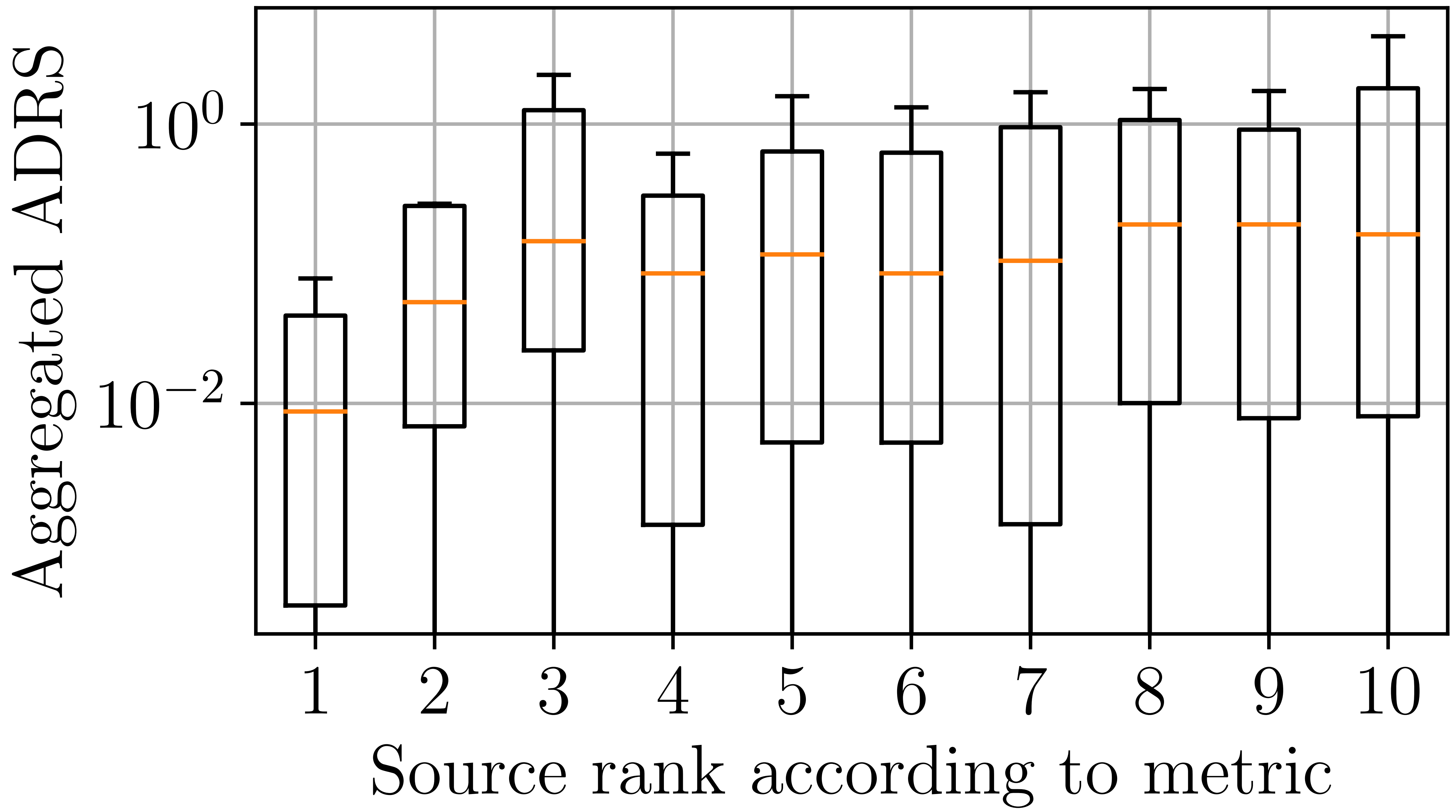
Effectiveness of the similarity metric: 37th-ranked source.



- X Low Specification Encoding similarity
- X Low Configuration Space Descriptor similarity

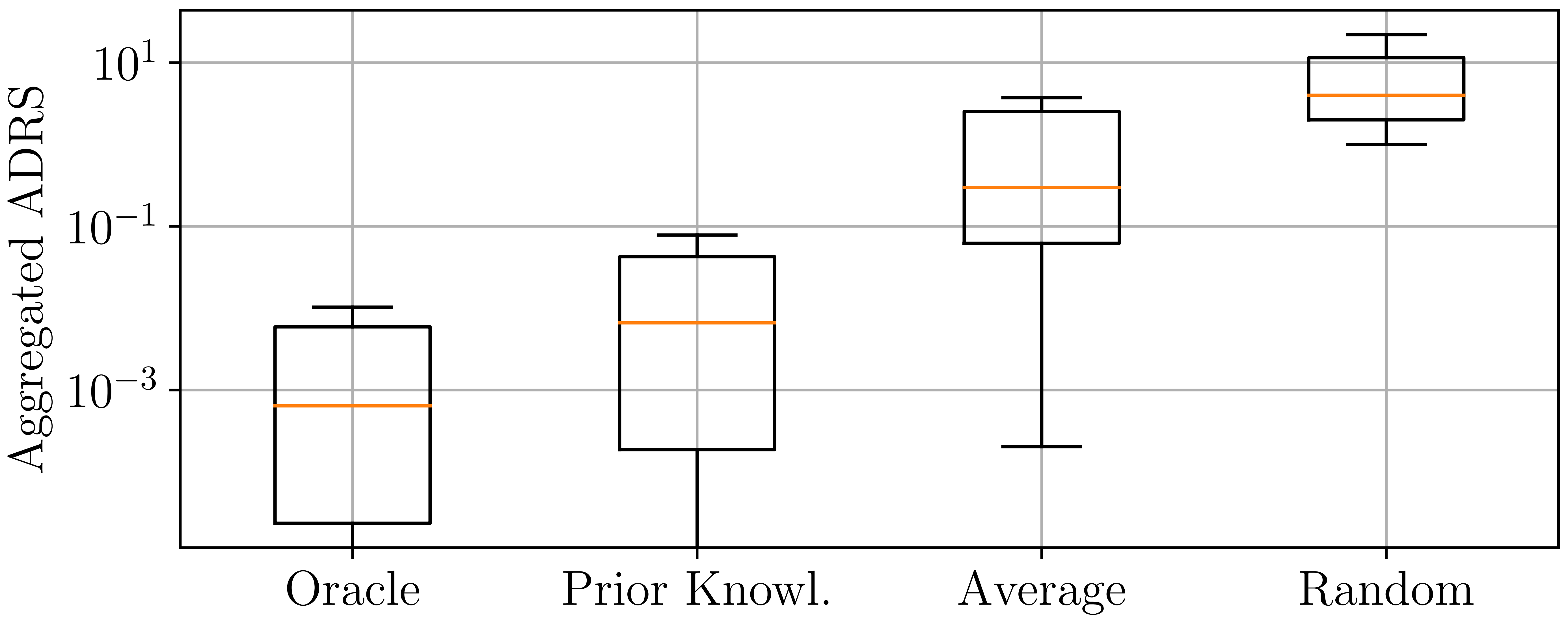
Results

Effectiveness of the similarity metric: source ranking.



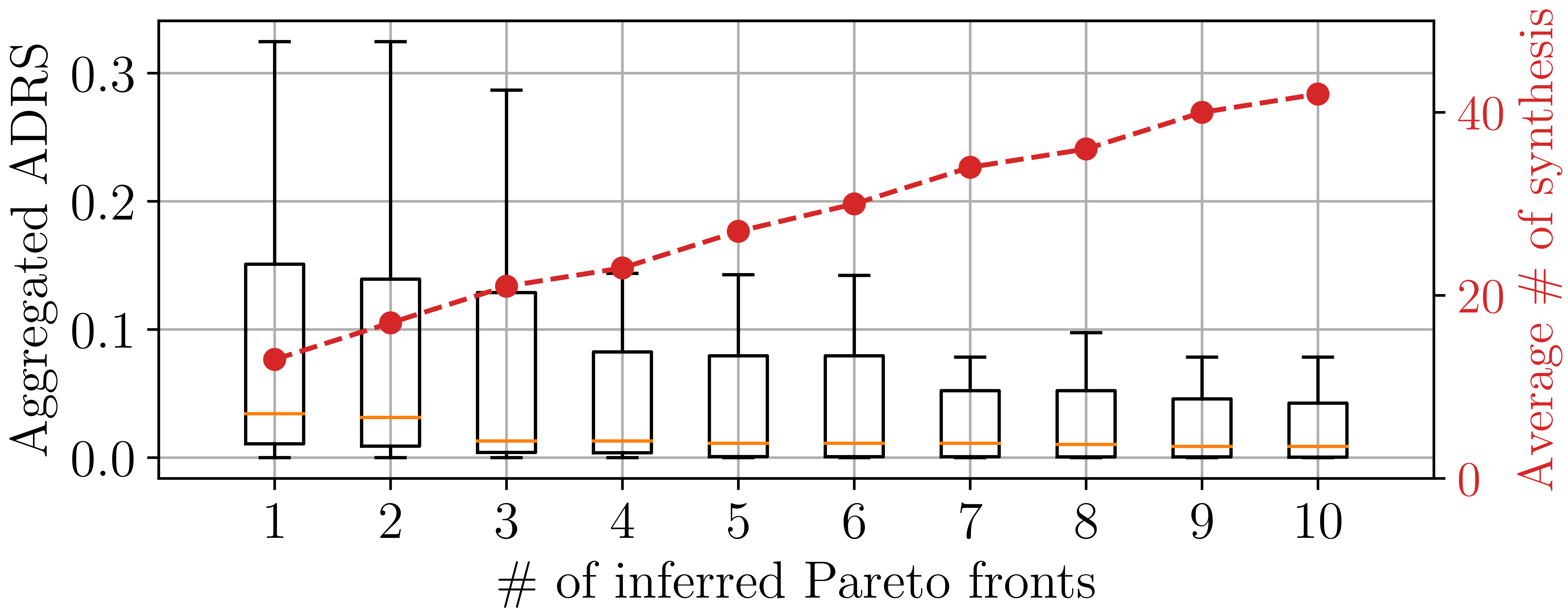
Results

Effectiveness of the similarity metric: selection criterion.



Results

Effectiveness of the similarity metric: influence of multiple Pareto frontier rank inference.



Results

Comparison of our methodology with respect to SoA ones for similar problems size.

$\|CS\|$	Prior Knowl.	Lattice [10]	Cluster [9]	RF-TED [8]	Zhong [11]
		Refinement-Based			Model-Based
< 200	7	36	37	155	NA
< 700	10	64	64	391	19
< 1800	22	230	290	1588	31
< 6000	19	460	460	1903	32
< 16000	NA	NA	NA	NA	35
< 32000	38	NA	NA	NA	NA

Number of synthesis required to reach an ADRS goal of 0.04.

[8] H. Liu et al. Design Automation Conference, 2013.
[9] L. Ferretti et al. IEEE Transactions on Emerging Topics in Computing, 2018.
[10] L. Ferretti et al. International Conference on Computer Design, 2018.
[11] Zhong et al. International Conference on Computer Design, 2014.

Table of Contents

Database of DSEs will be released soon!

Thank you for your attention!