Scientific seminar
SUSU Big Data and Machine Learning Lab

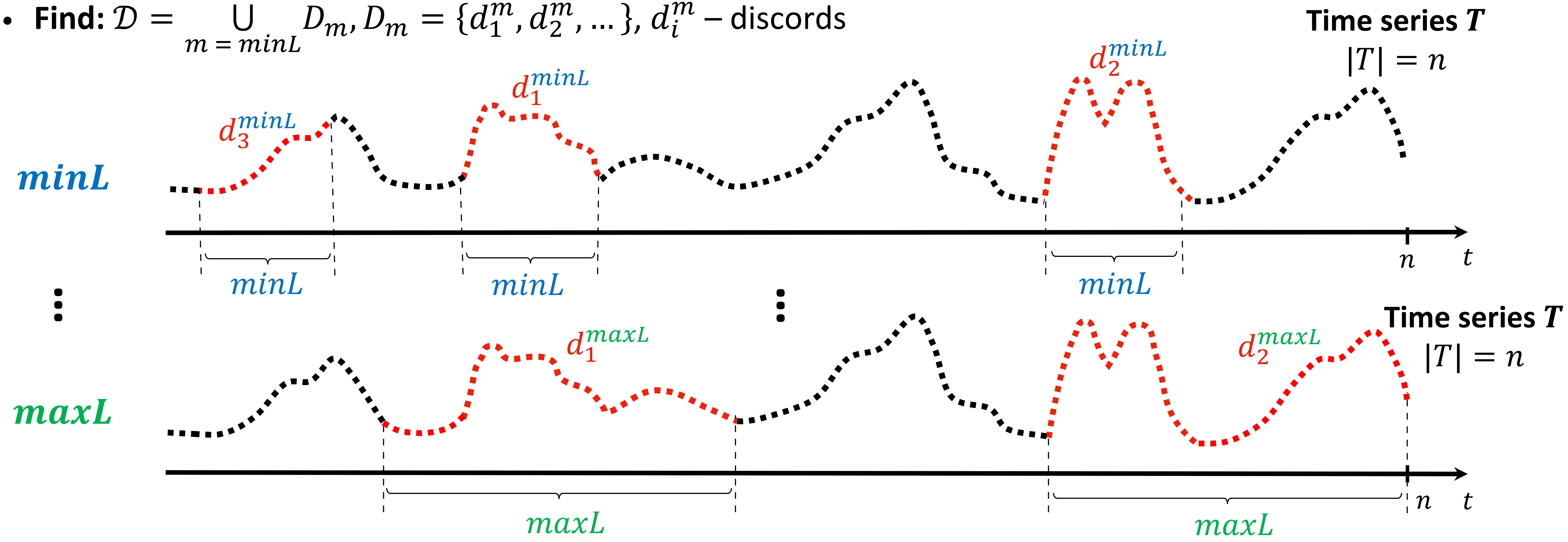
Current results of the project "Development of models, methods and algorithms for time series anomaly discovery"

Yana Kraeva, Mikhail Zymbler

Chelyabinsk – 2023

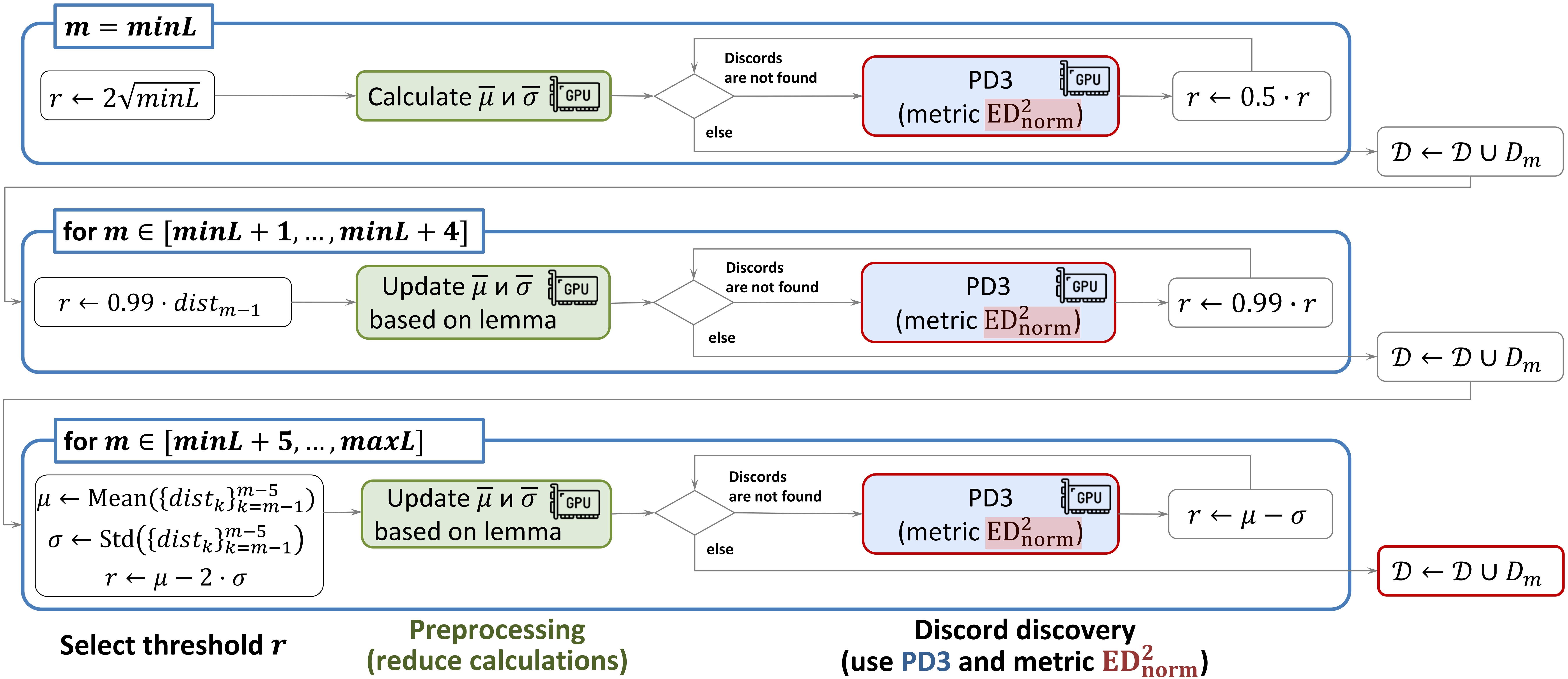
Task of arbitrary length discord discovery[1)]

- **Discord** is a subsequence of a given length whose nearest neighbor is at least at a given threshold far away
- **Given:** time series T, **discord length range** $minL, \ldots, maxL$
- **Find:** $\mathcal{D} = \bigcup_{m=minL}^{maxL} D_m, D_m = \{d_1^m, d_2^m, \ldots\}, d_i^m$ – discords



[1)] Nakamura T., *et al*. MERLIN: parameter-free discovery of arbitrary length anomalies in massive time series archives. IEEE ICDM 2020. pp. 1190-1195.

PALMAD (Parallel Arbitrary Length MERLIN-based Anomaly Discovery)[1)]

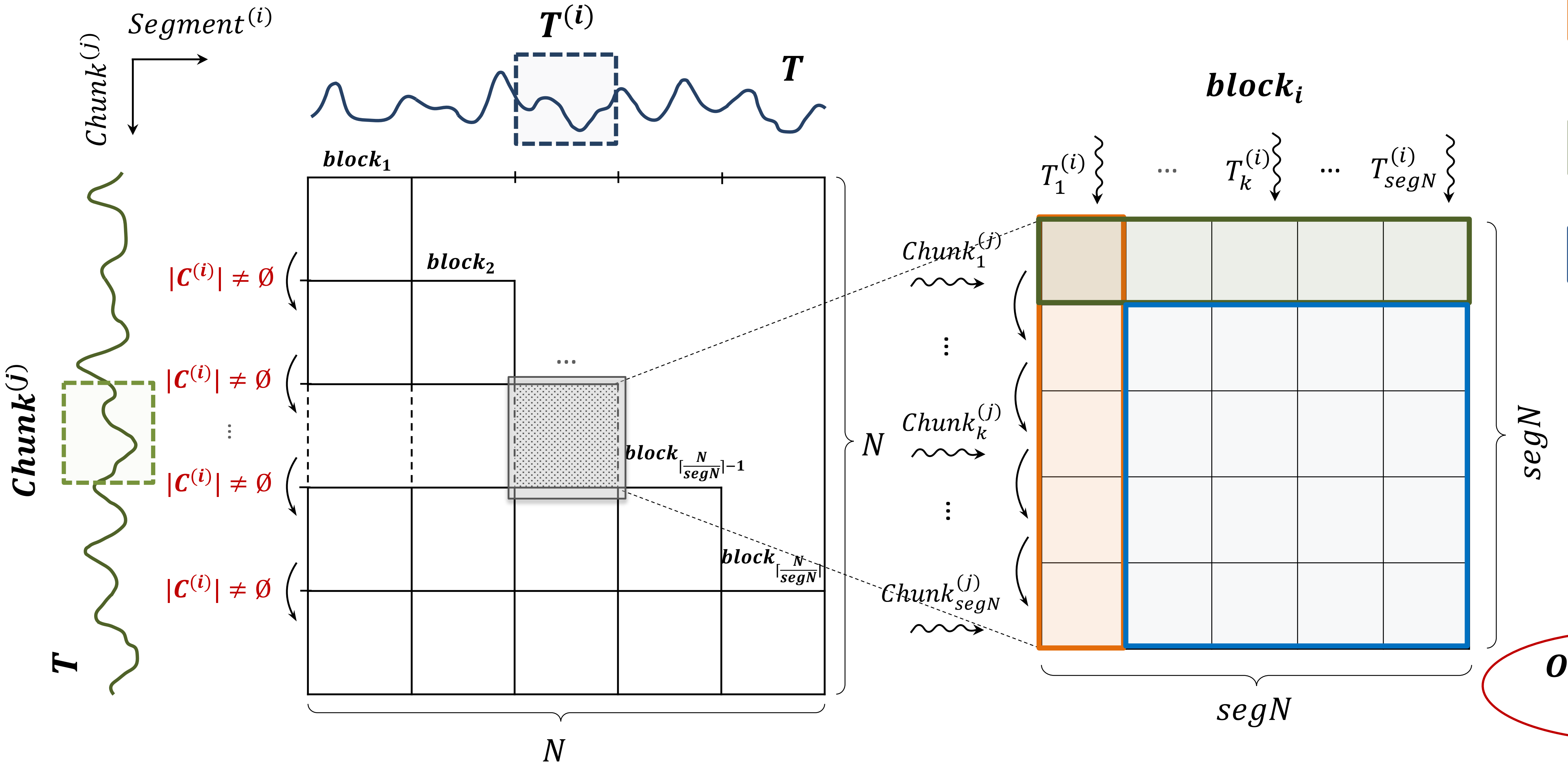


[1)] Zymbler M., Kraeva Y. High-performance Time Series Anomaly Discovery on Graphics Processors. CoRR. 2023. Vol. abs/2304.01660. arXiv: 2304.01660.

PD3: Parallel DRAG-based Discord Discovery[1)]

1. Two phases: the candidate selection and discords refinement
2. Data parallelism: SIMT (Single Instruction Multiple Threads)
3. Efficient calculations of scalar products $T_{i,m} \cdot T_{j,m}$

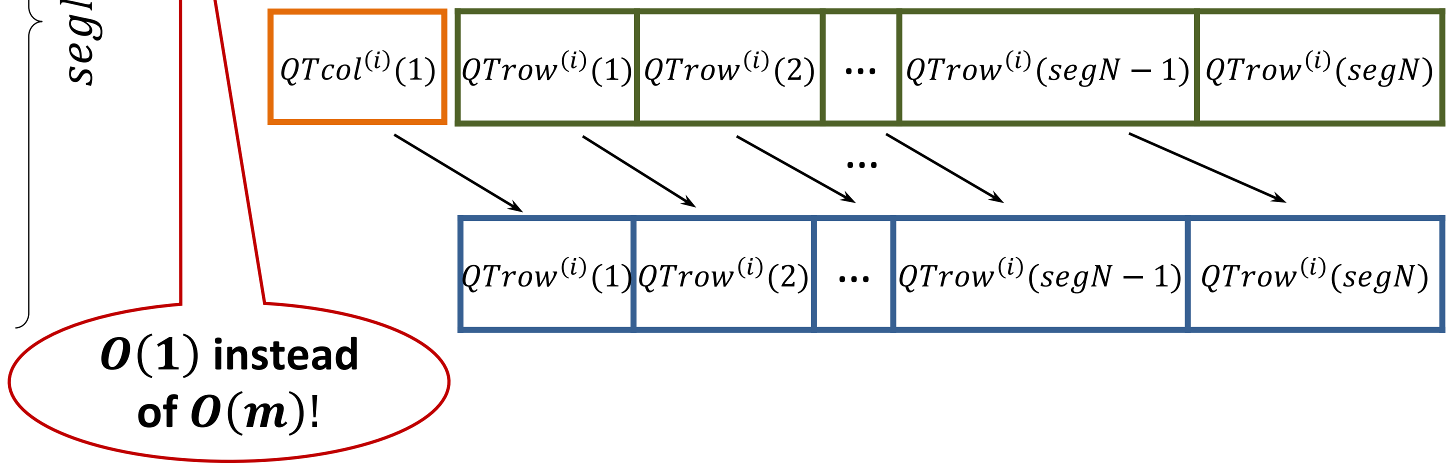
$$\mathrm{ED}^2_{\mathrm{norm}}\left(T_{i,m}, T_{j,m}\right) = 2m\left(1 - \frac{T_{i,m} \cdot T_{j,m} - m\mu_i\mu_j}{m\sigma_i\sigma_j}\right)$$



$$\boldsymbol{QTrow^{(i)}(tid)} = \sum_{k=1}^{m} T^{(i)}_{tid}(k) \cdot Chunk^{(j)}_{1}(k)$$

$$\boldsymbol{QTcol^{(i)}(tid)} = \sum_{k=1}^{m} T^{(i)}_{1}(k) \cdot Chunk^{(j)}_{tid}(k)$$

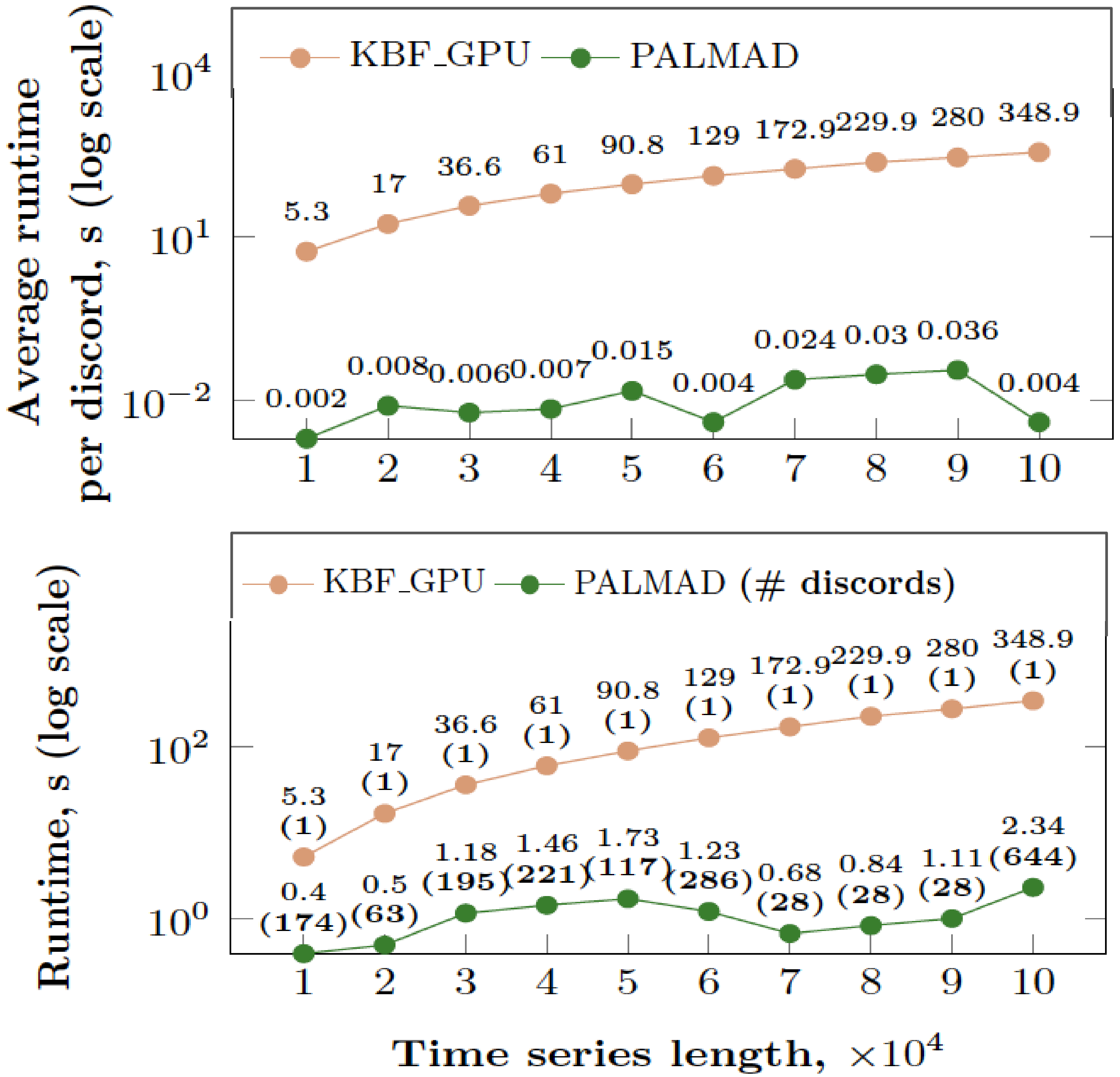
$$\boldsymbol{QTrow^{(i)}(tid)} = QTrow^{(i)}(tid-1) - T^{(i)}_{tid-1}(1) \cdot Chunk^{(j)}_{tid-1}(1) + T^{(i)}_{tid}(m) \cdot Chunk^{(j)}_{tid}(m)$$



1) Kraeva Y., Zymbler M. Parallel algorithm for time series discord discovery on a graphics processor. Pattern Recognition and Image Analysis. 2023. Vol. 33, no. 2. Accepted for publication.

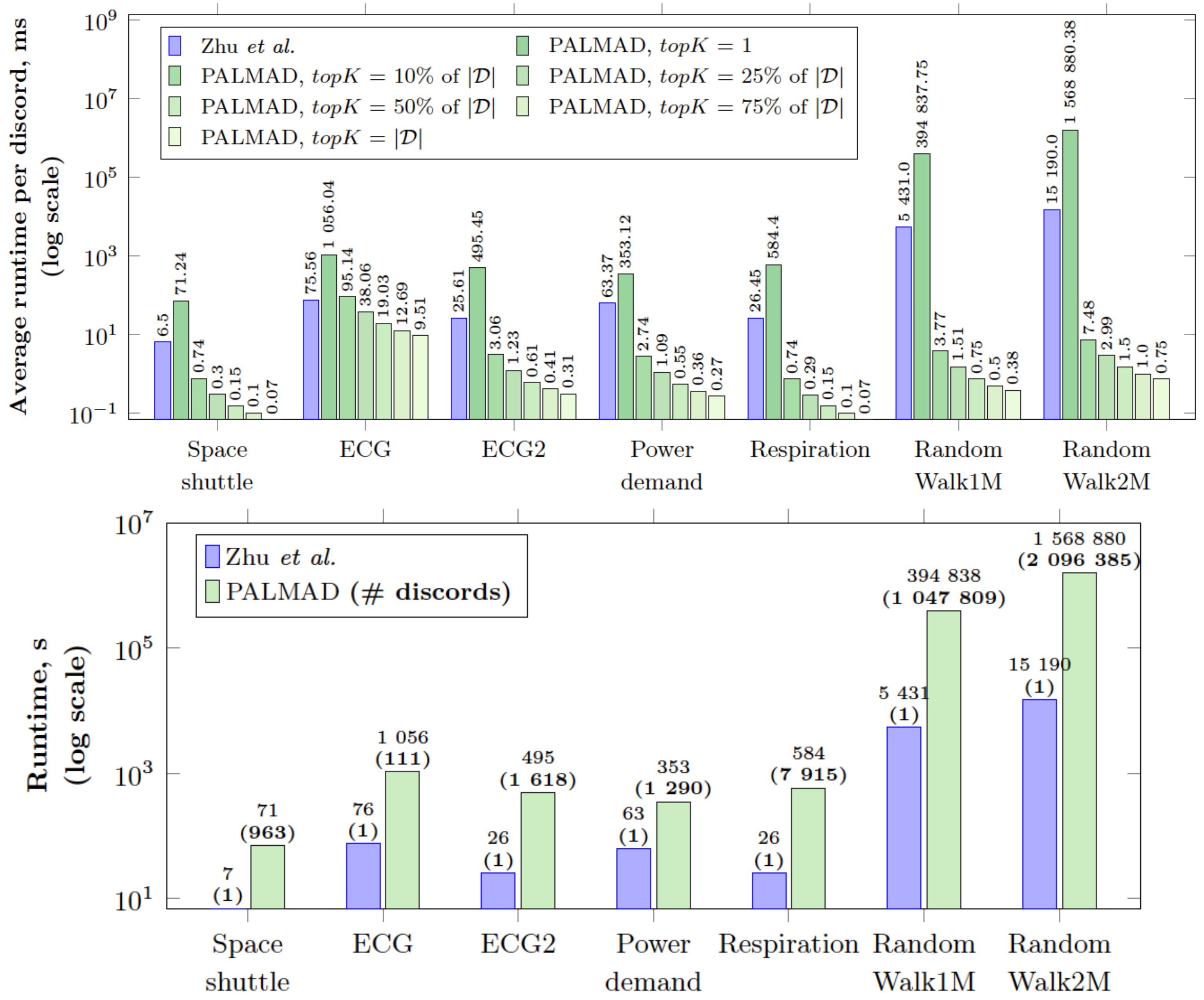
Experiments: Comparison with analogs

- **KBF_GPU**



PALMAD significantly outruns KBF_GPU

- **Zhu et al.**



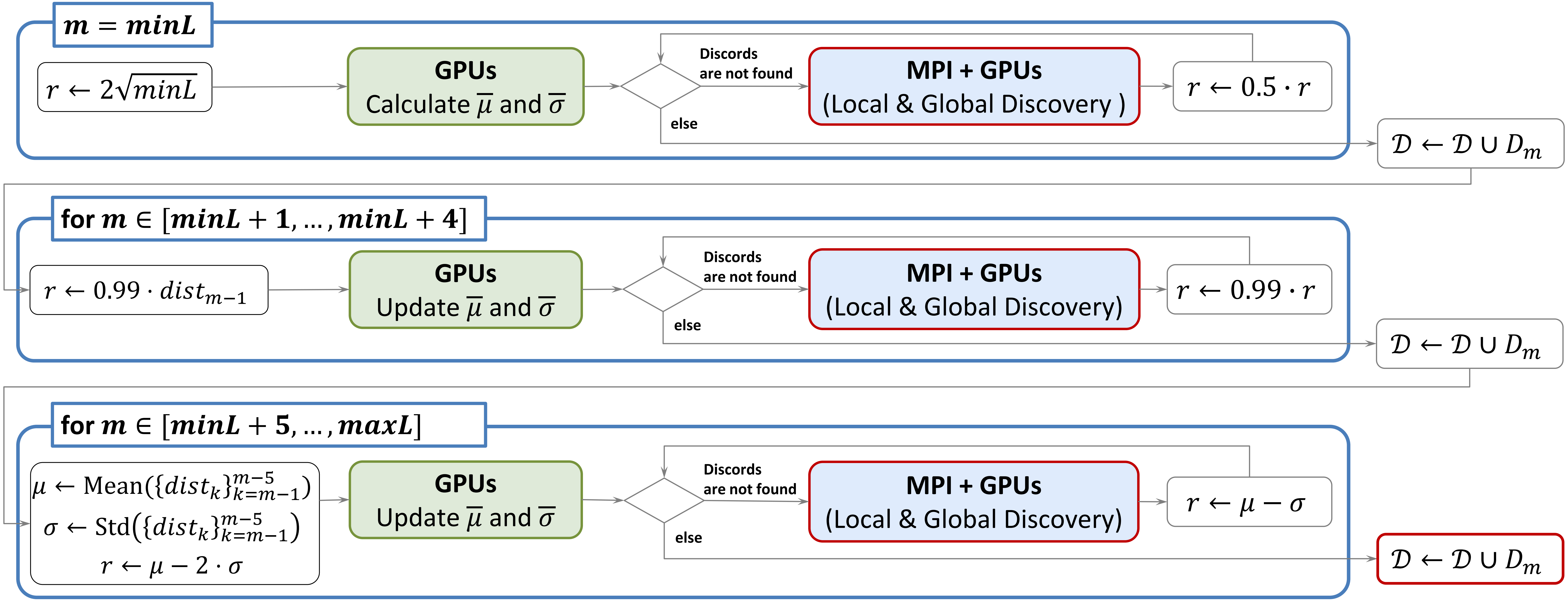
PALMAD significantly outruns Zhu et al.
w.r.t. the average running time to discover one discord

Limitations of PALMAD

PALMAD cannot

- **deal with a time series that larger than RAM of a GPU**
- discover anomalies online

From PALMAD to PADDi (PALMAD-based Anomaly Discovery on Distributed GPUs)



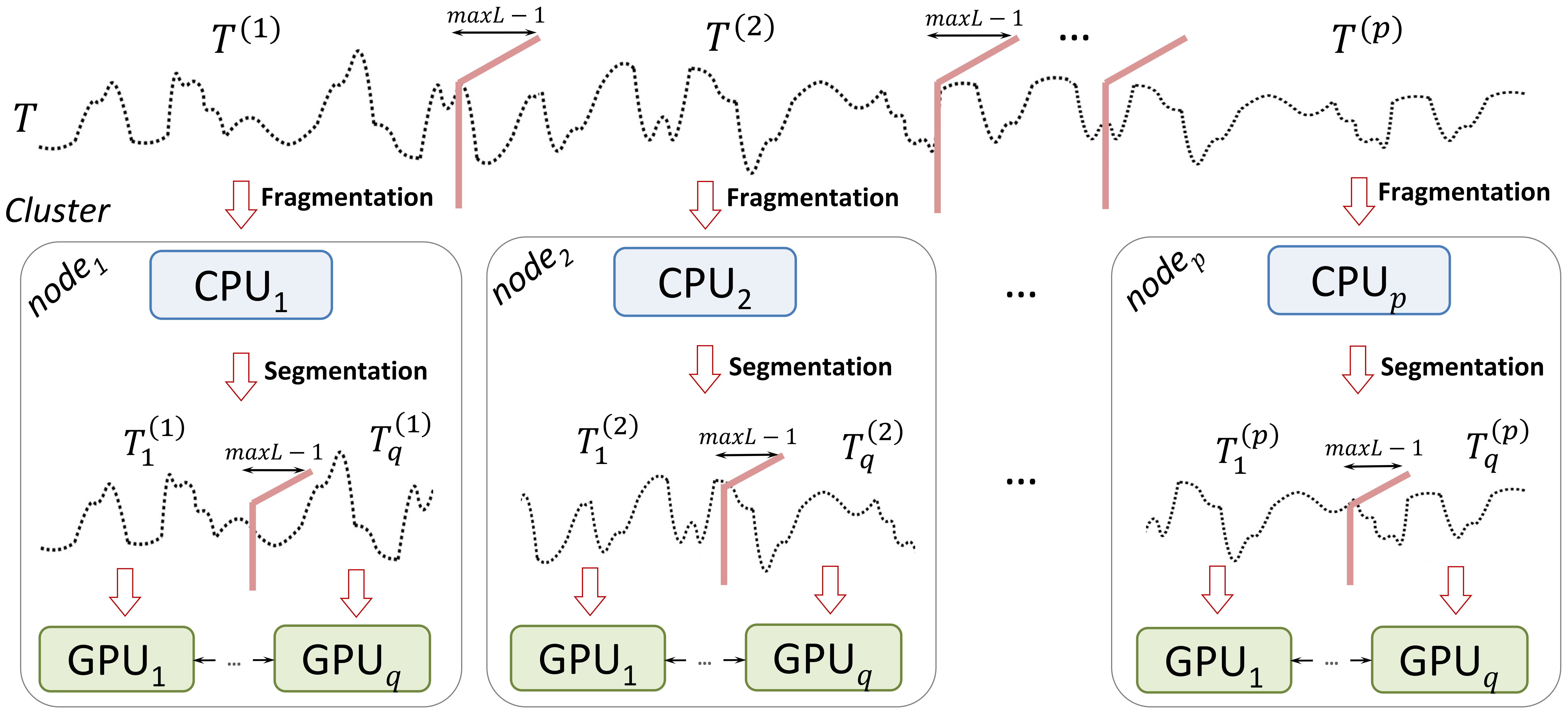
Select threshold r

Preprocessing (reduce calculations by lemma)

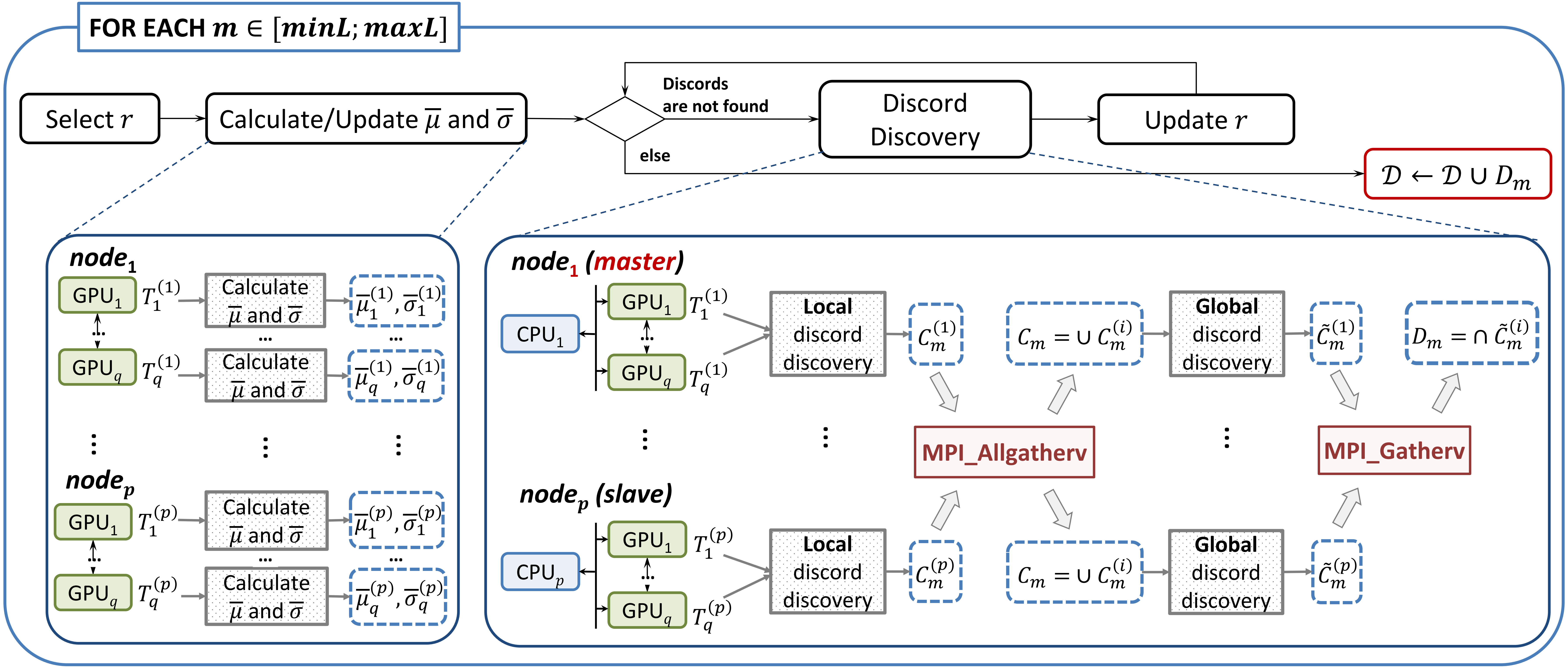
Discord discovery (use PD3 and metric $\mathrm{ED}^2_{\mathrm{norm}}$)

[1] Zymbler M., Kraeva Y. High-performance Time Series Anomaly Discovery on Graphics Processors. CoRR. 2023. Vol. abs/2304.01660. arXiv: 2304.01660.

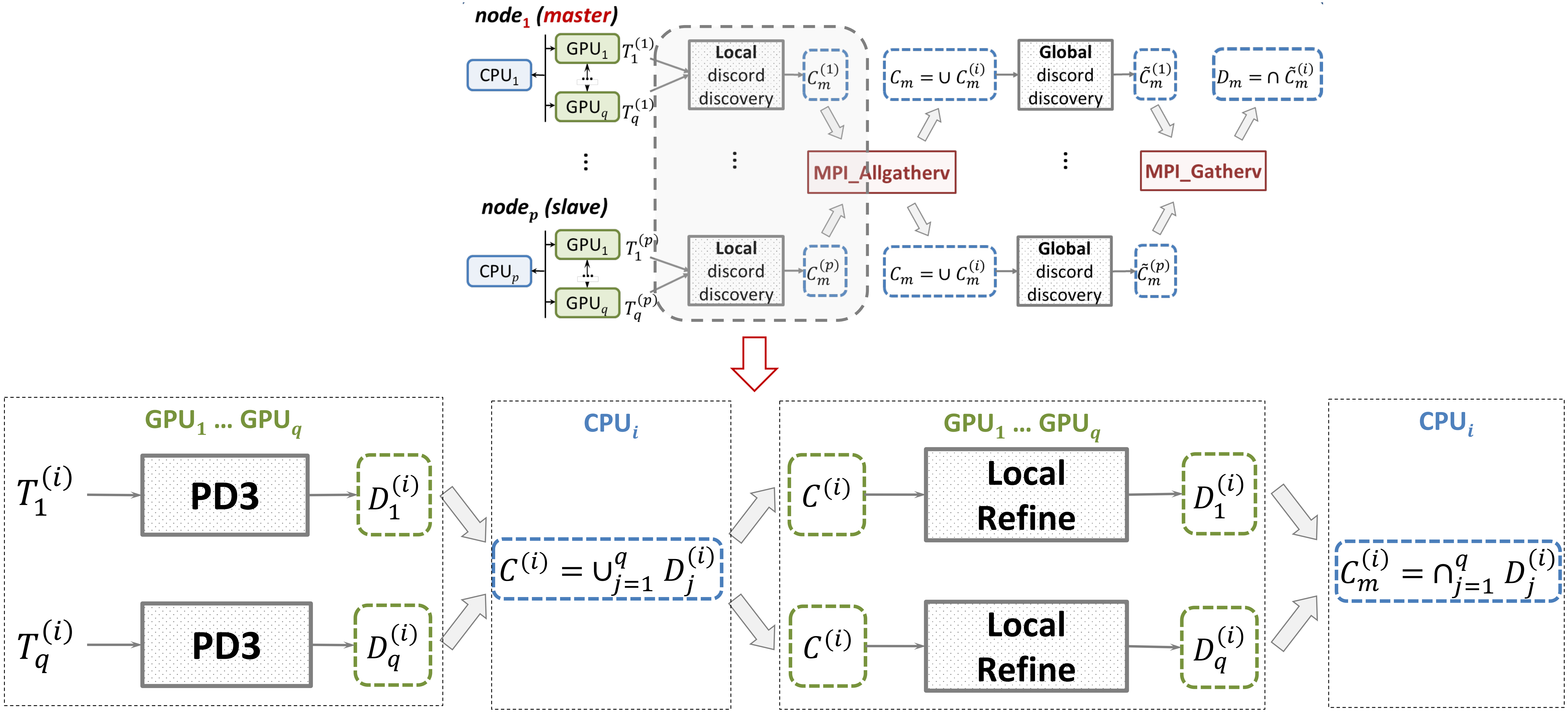
PADDi: Fragmentation and segmentation



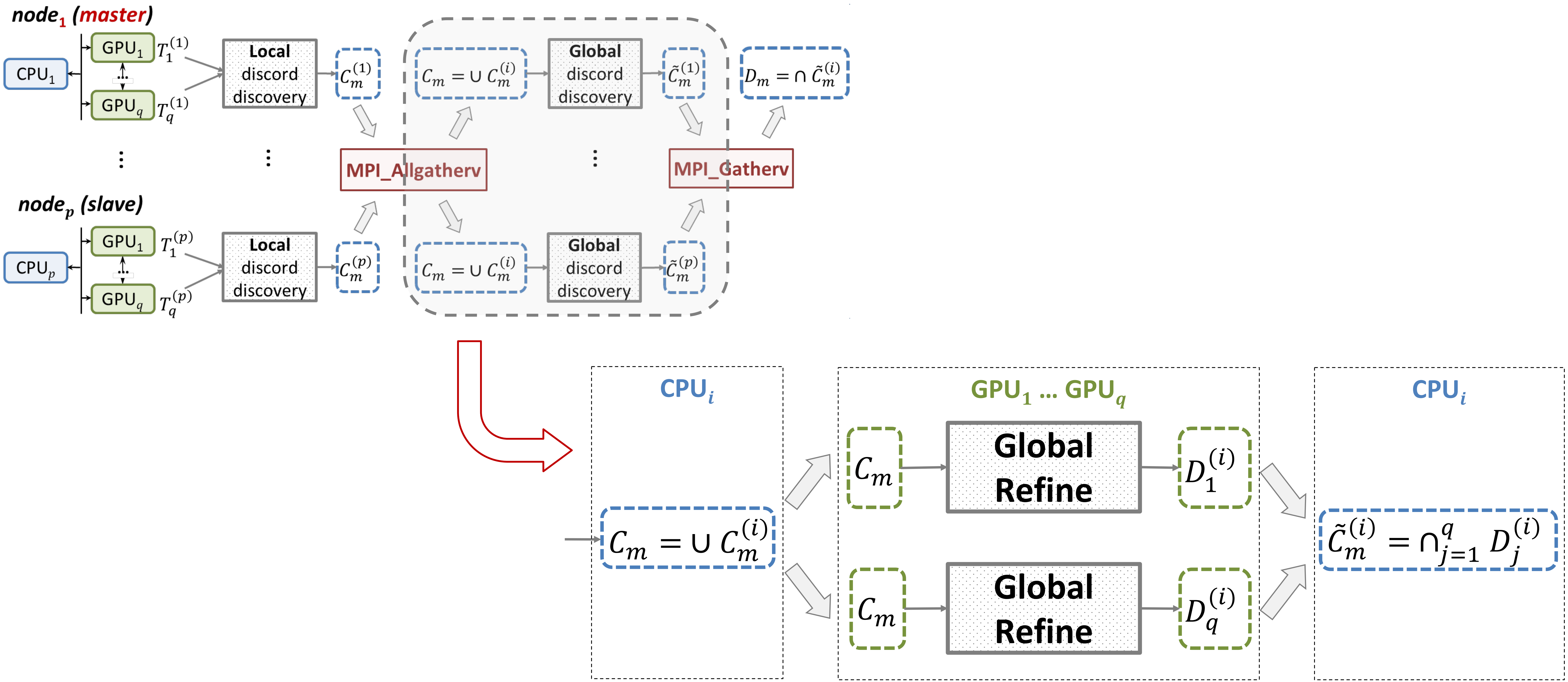
PADDi: General scheme



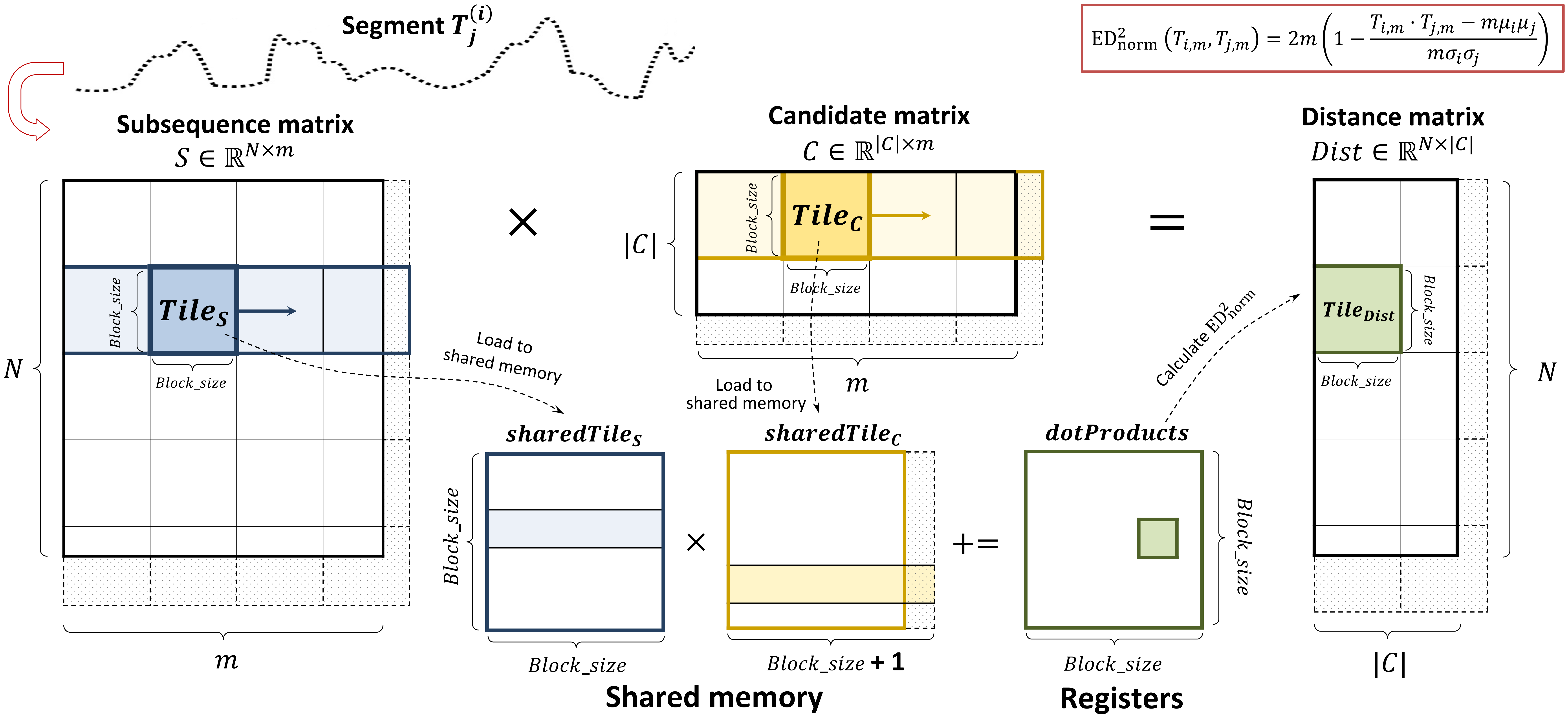
PADDi: Local Discord Discovery



PADDi: Global Discord Discovery



Local/Global Discord Refinement



Experiments

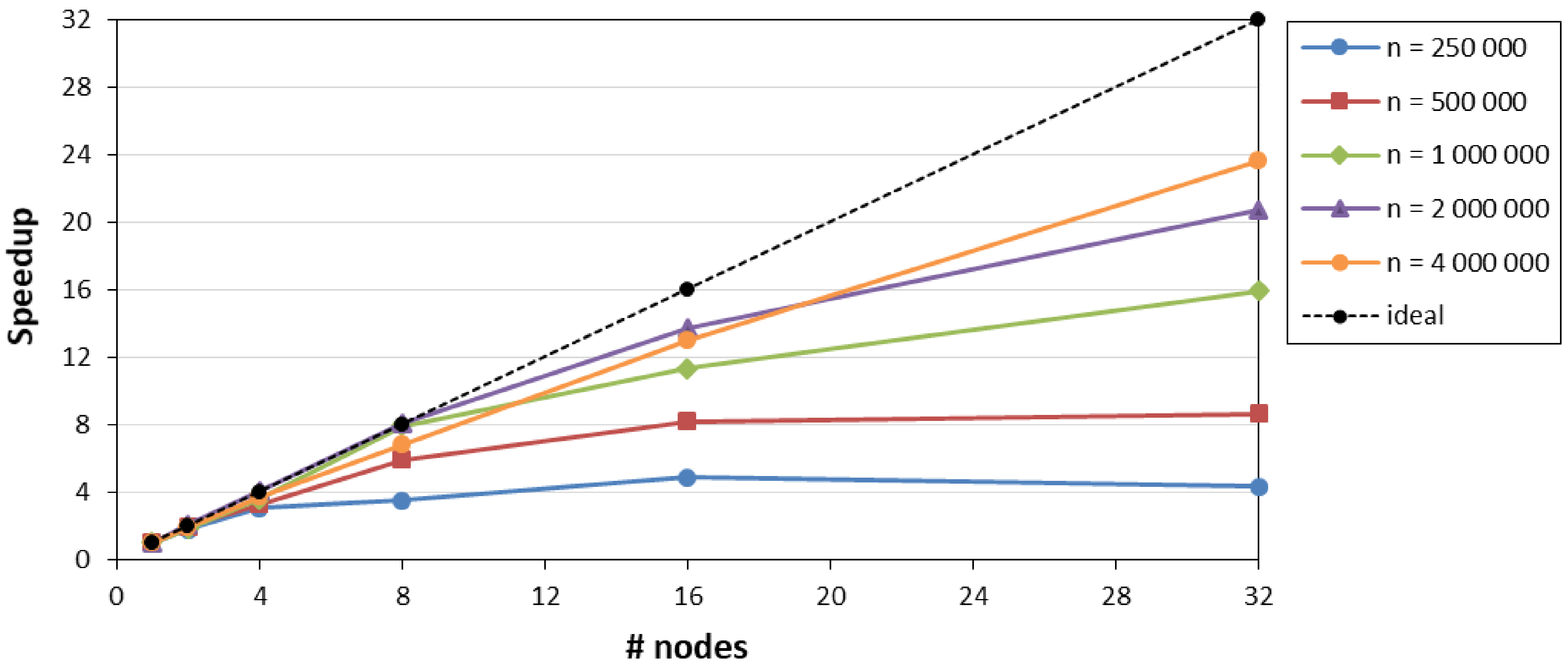
- **Hardware**
  - Cluster: Lomonosov-2, MSU
  - Node: 1 x Xeon Gold 6126 with 2 x NVIDIA Tesla P100 (3 584 CUDA-cores, @1.19 GHz, 4 TFLOPS)
- **Datasets**

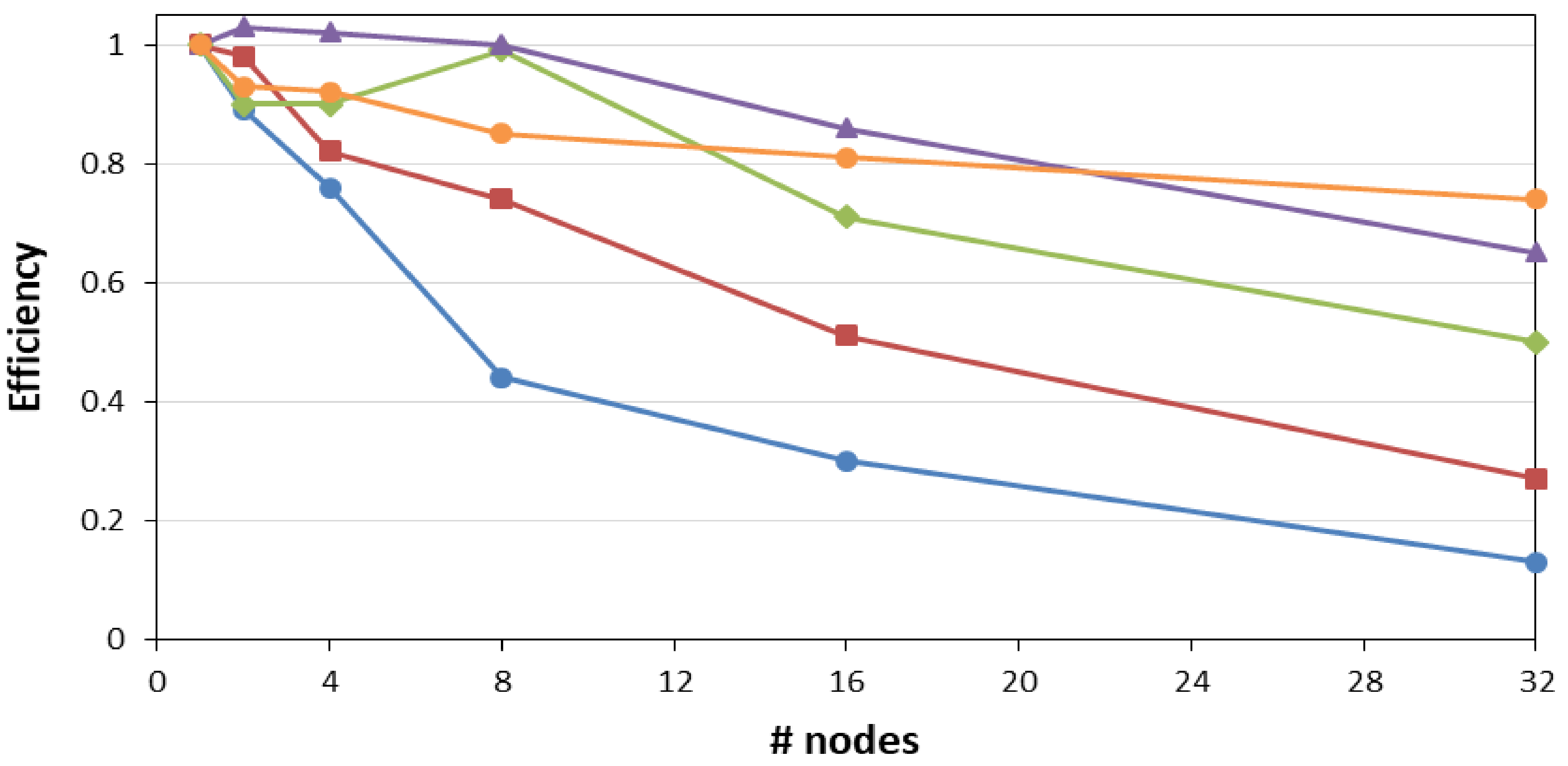
Time series	Time series length, n	Discord length range, $minL..maxL$	Domain
ECG[1)]	$4 \cdot 10^6$	64..128	Electrocardiogram of an adult patient
GAP[2)]	$2 \cdot 10^6$	256..512	Power demand in France

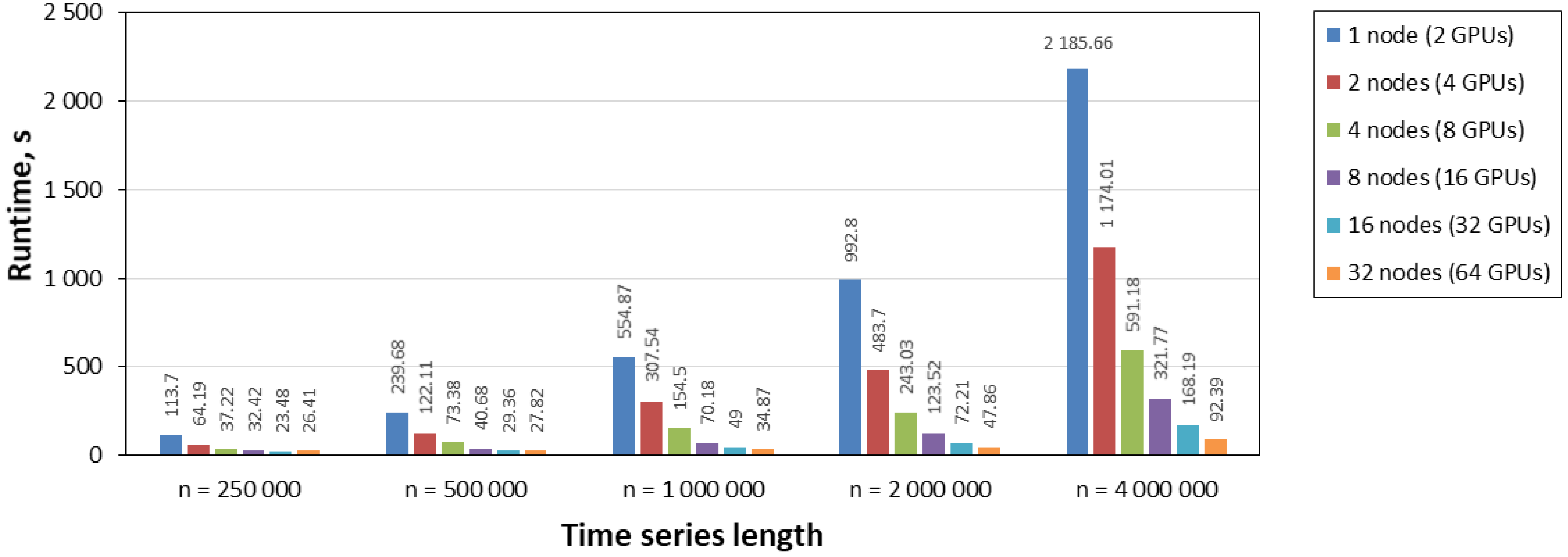
- **Measures**
  - Performance: 10-runs averaged runtime without I/O
  - Speedup: $s(p) = {}^{t_1}/_{t_p}$, where t_1 – runtime on one node, t_p– runtime on p nodes
  - Parallel efficiency: $e(p) = s(p) \;/\; p$

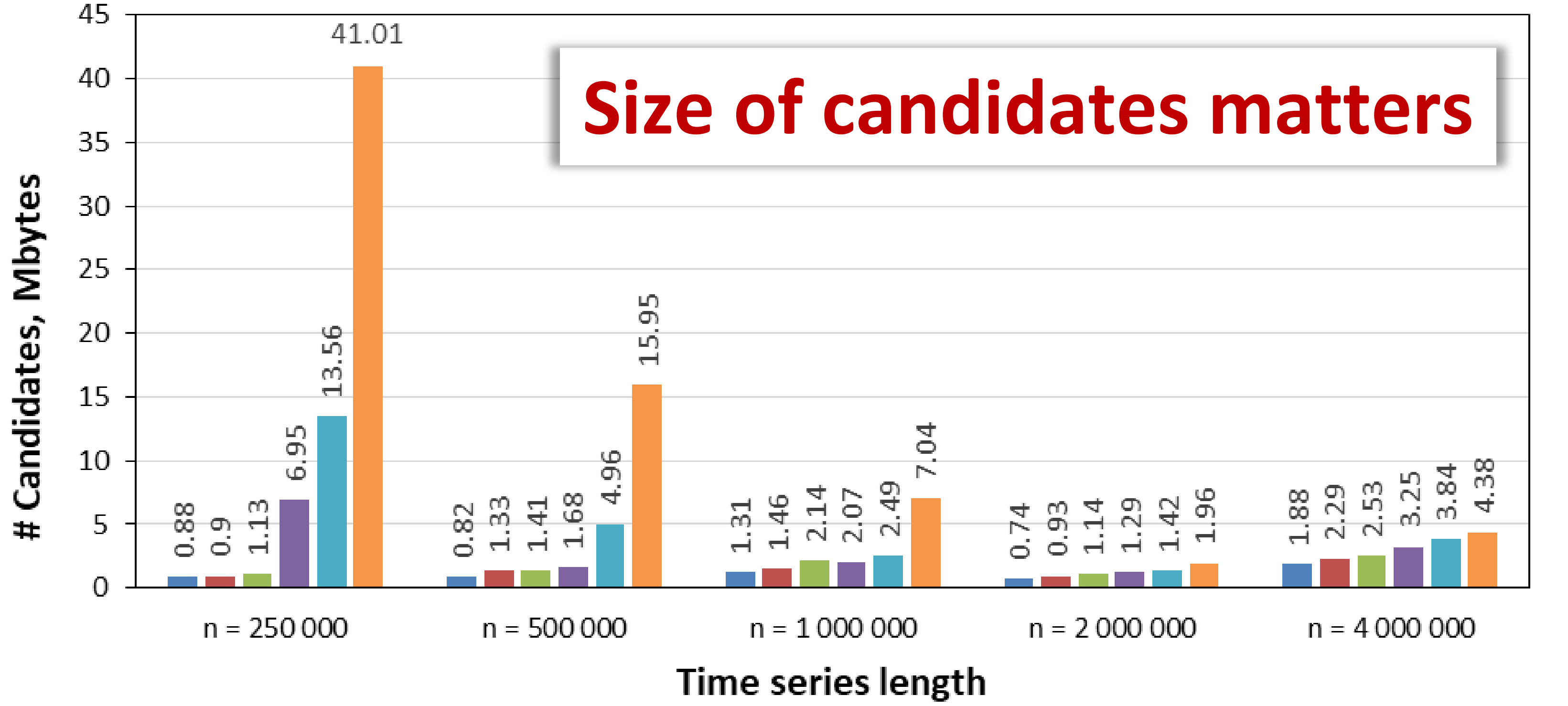
[1)] Lu Y. et al. Matrix Profile XXIV: Scaling Time Series Anomaly Detection to Trillions of Datapoints and Ultra-fast Arriving Data Streams. ACM SIGKDD 2022. pp. 1173-1182.
[2)] Linardi M. et al. Matrix Profile X: VALMOD - Scalable Discovery of Variable-Length Motifs in Data Series. SIGMOD 2018. pp. 1053-1066.

Scalability: ECG

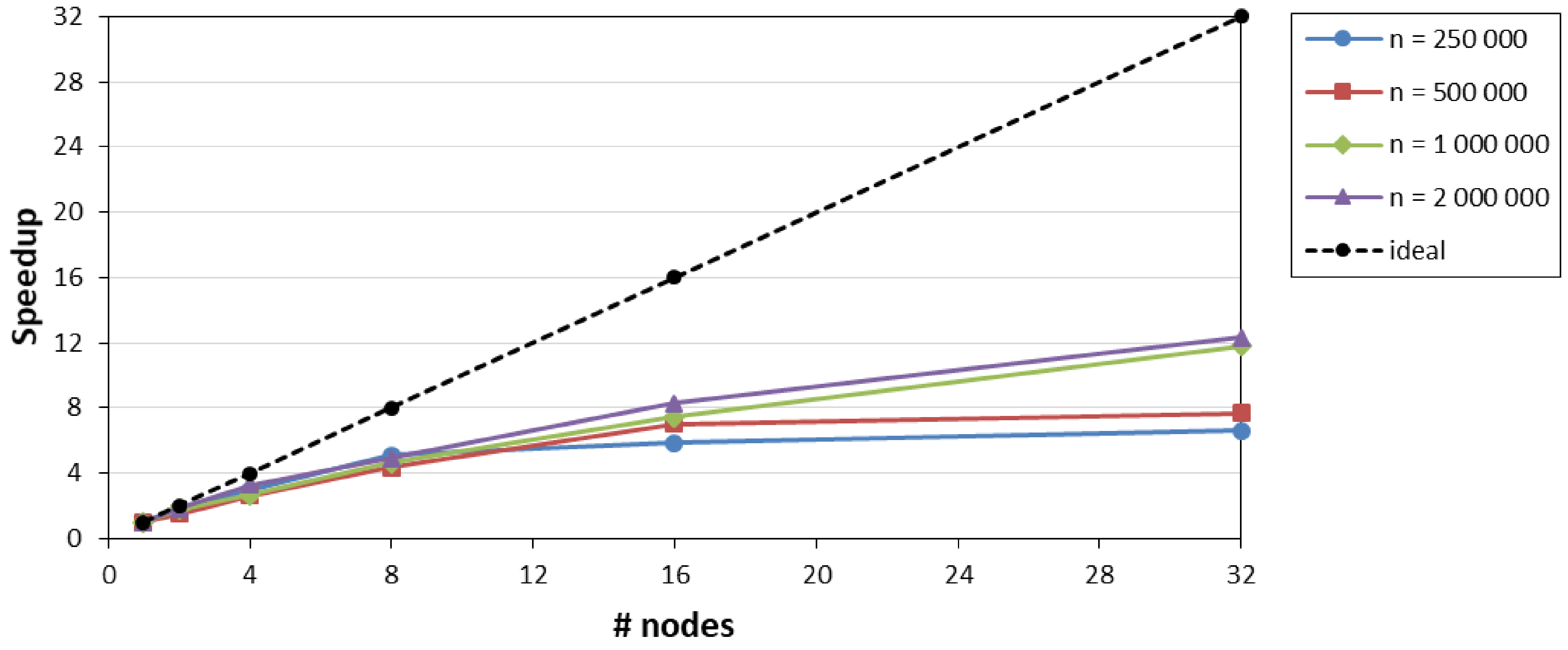


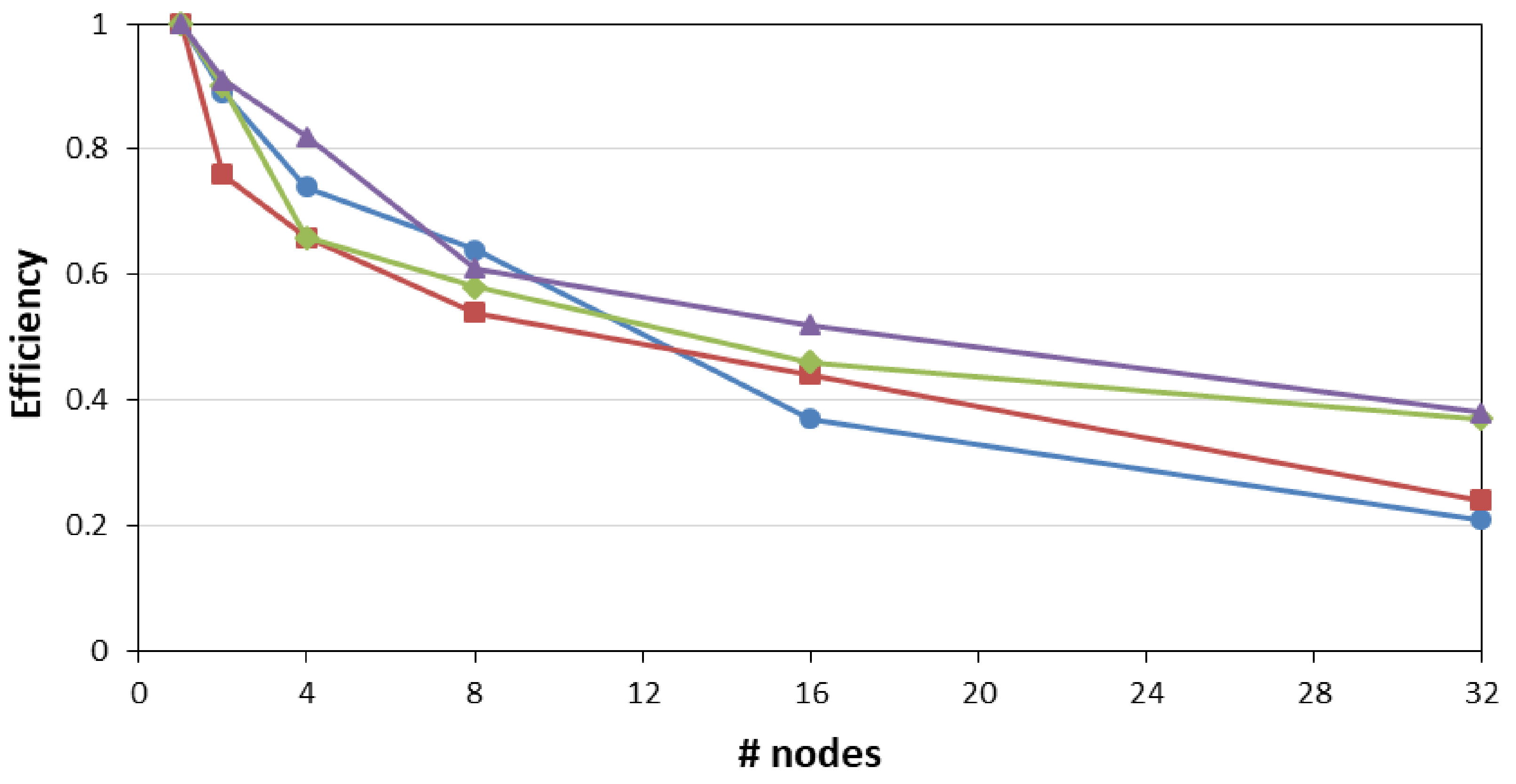


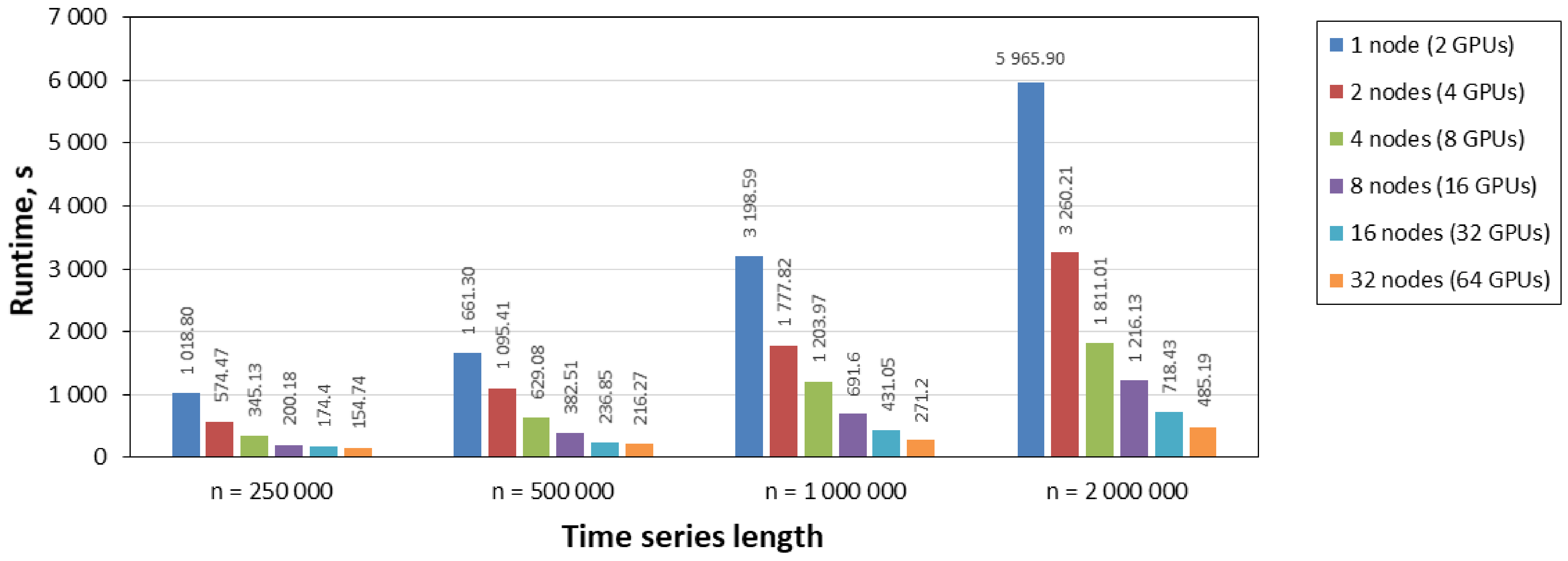


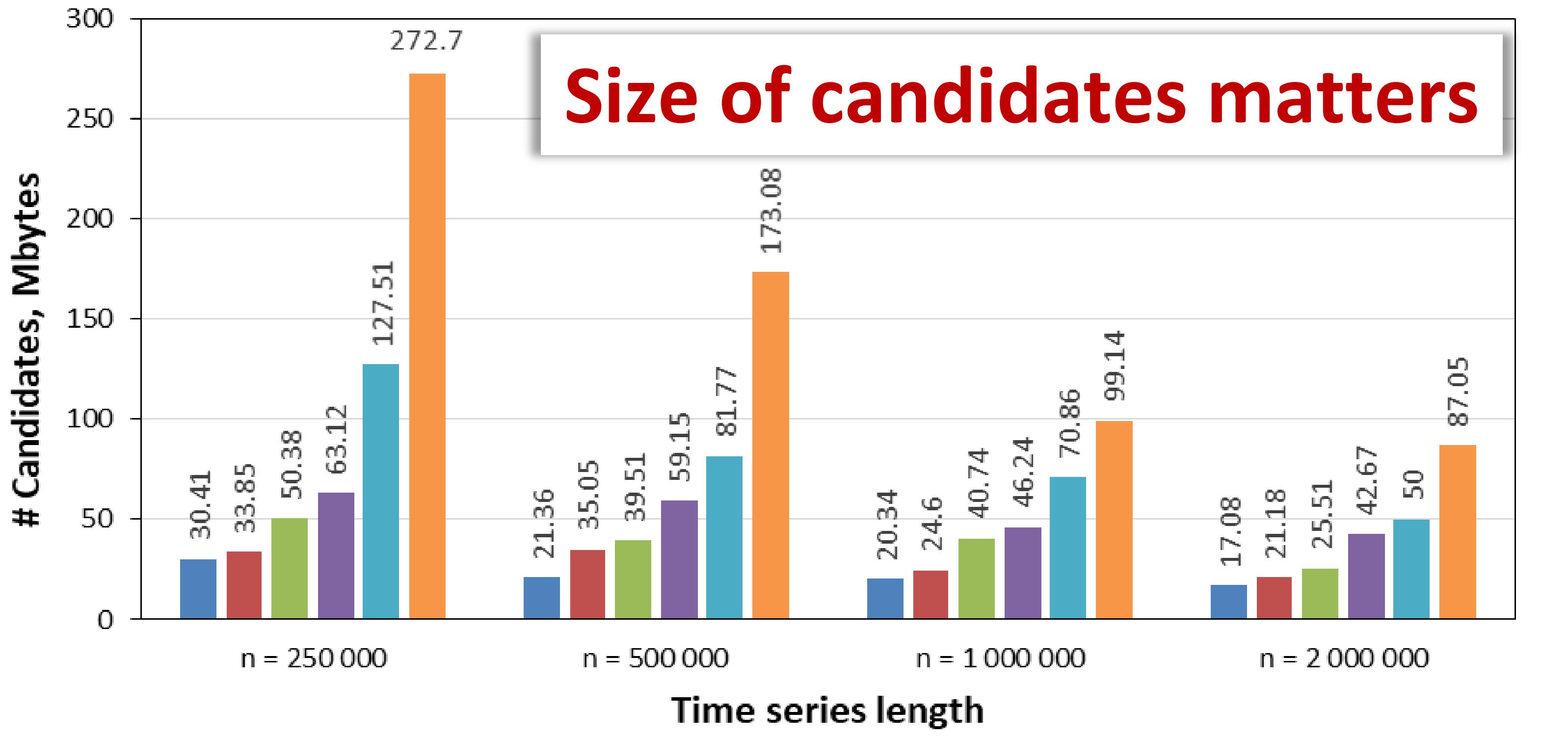


Scalability: GAP







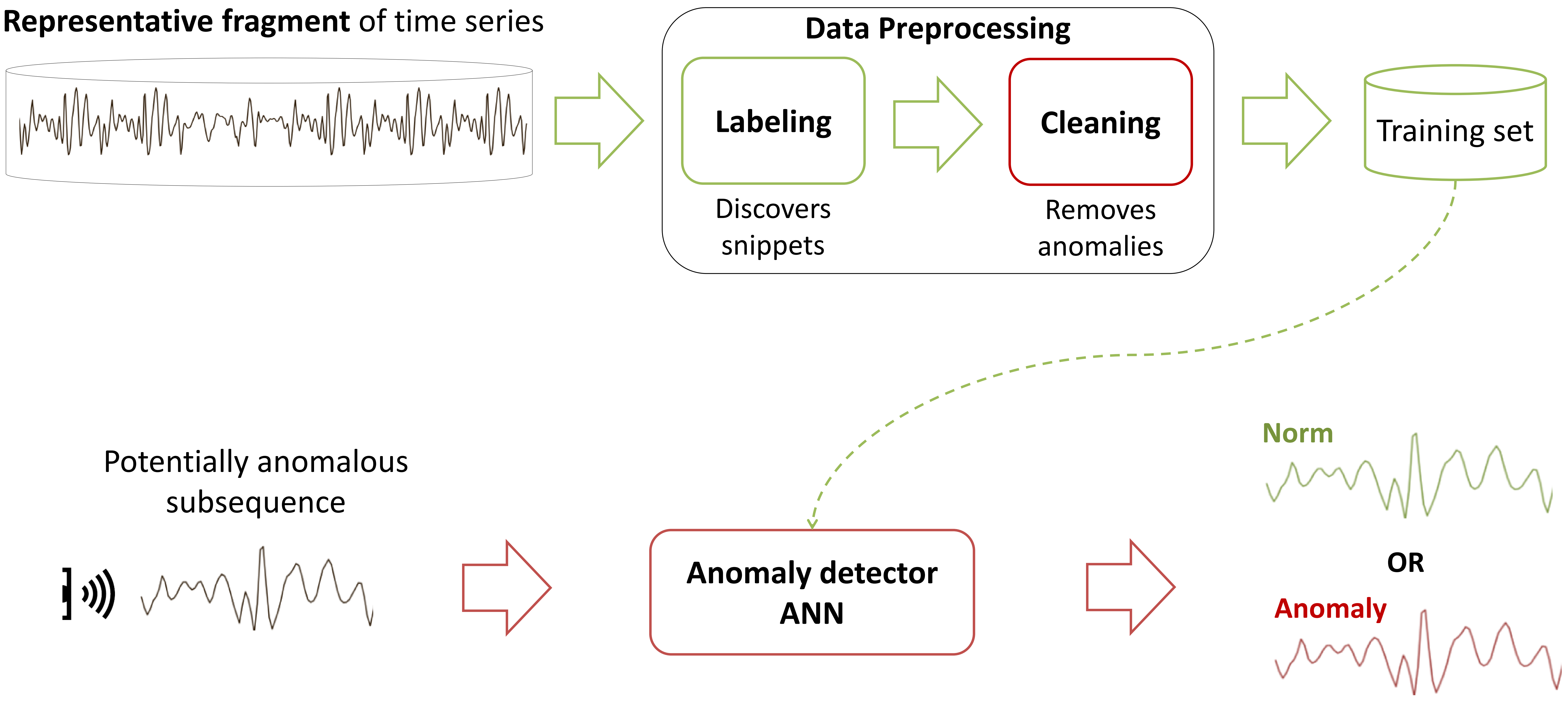


Limitations of PALMAD

PALMAD cannot

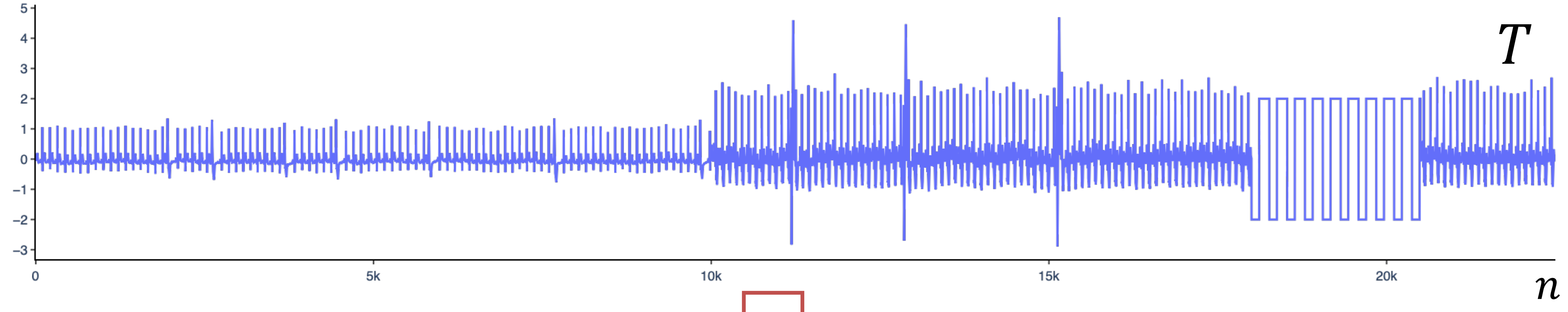
- deal with a time series that larger than RAM of a GPU
- **discover anomalies online**

Discover anomalies online: DiSSiD (Discord, Snippet, and Siamese Net-based Detector)

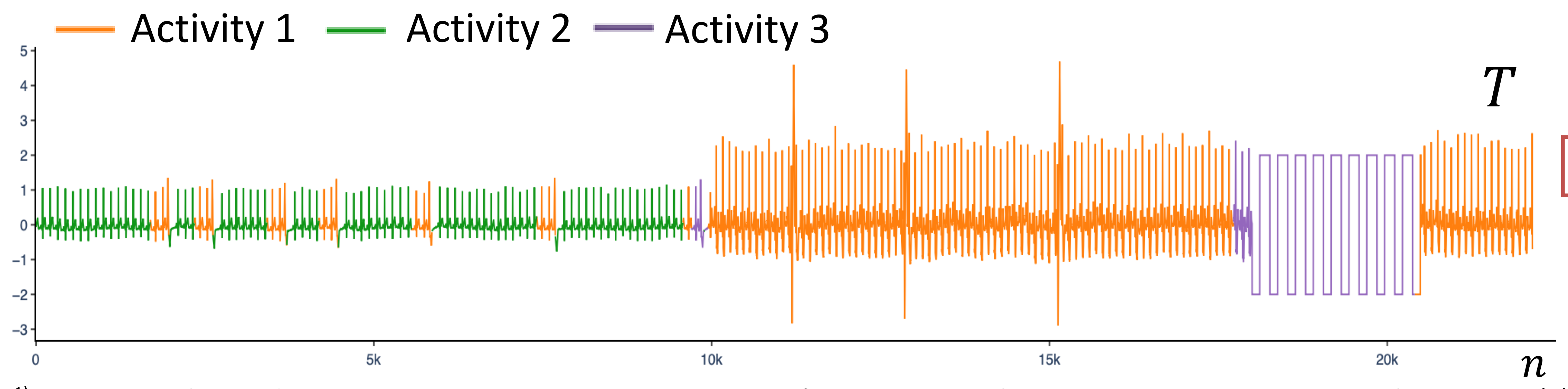


DiSSiD: Labeling

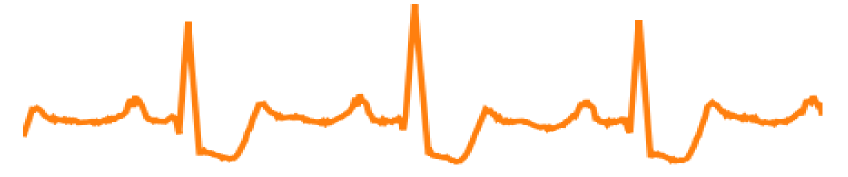
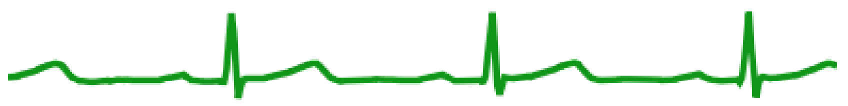
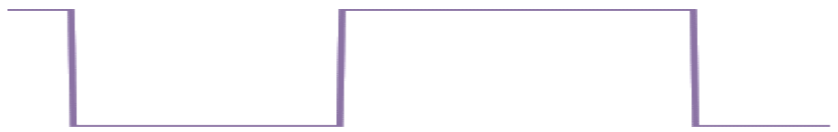
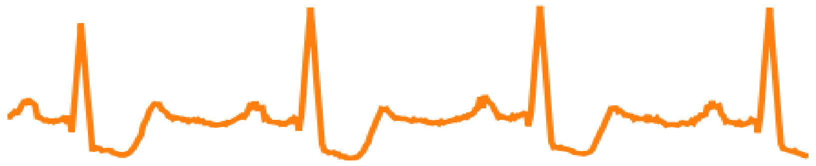
- Labeling is based on the snippet concept[1)]
- PSF (Parallel Snippet Finder)[2)] is applied for snippet discovery



PSF



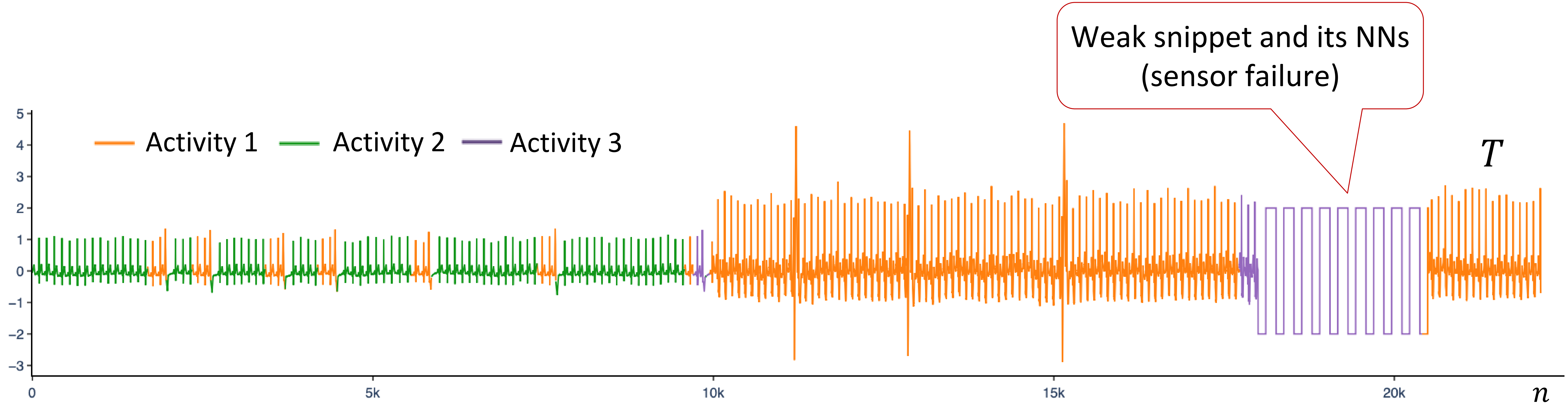
Labeled set L

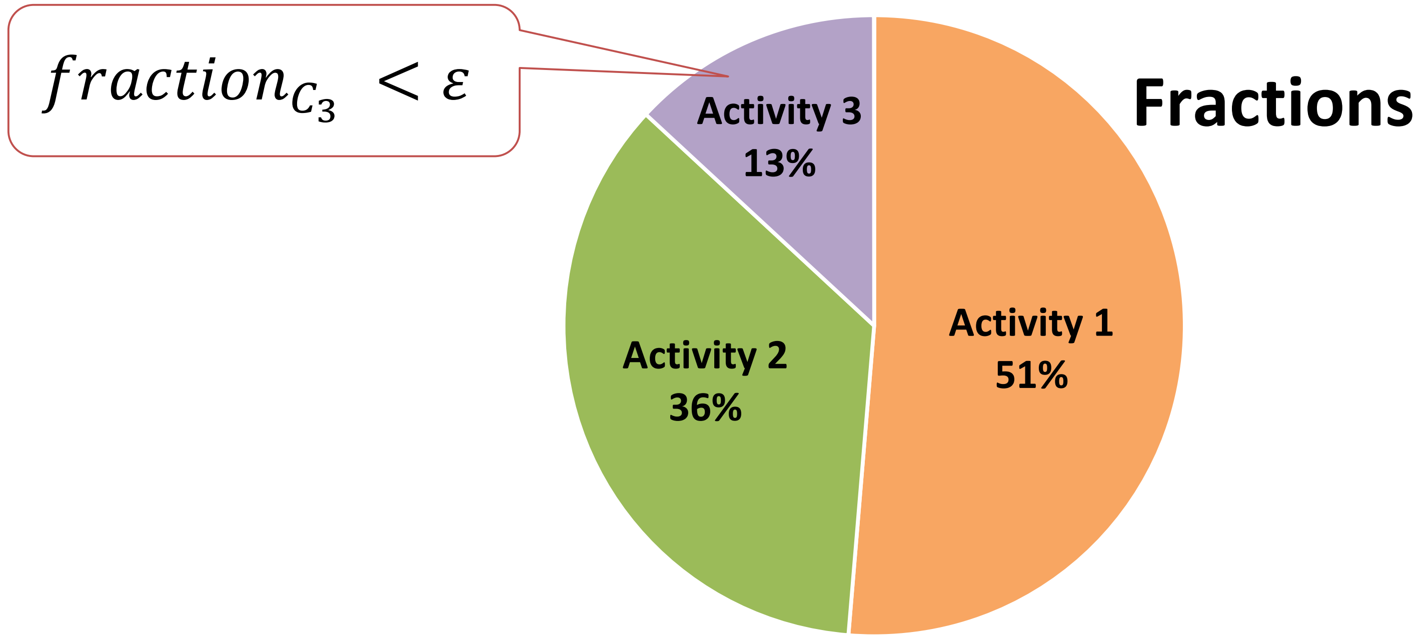
Subsequence	Label
	1
	2
	3
…	…
	1

m

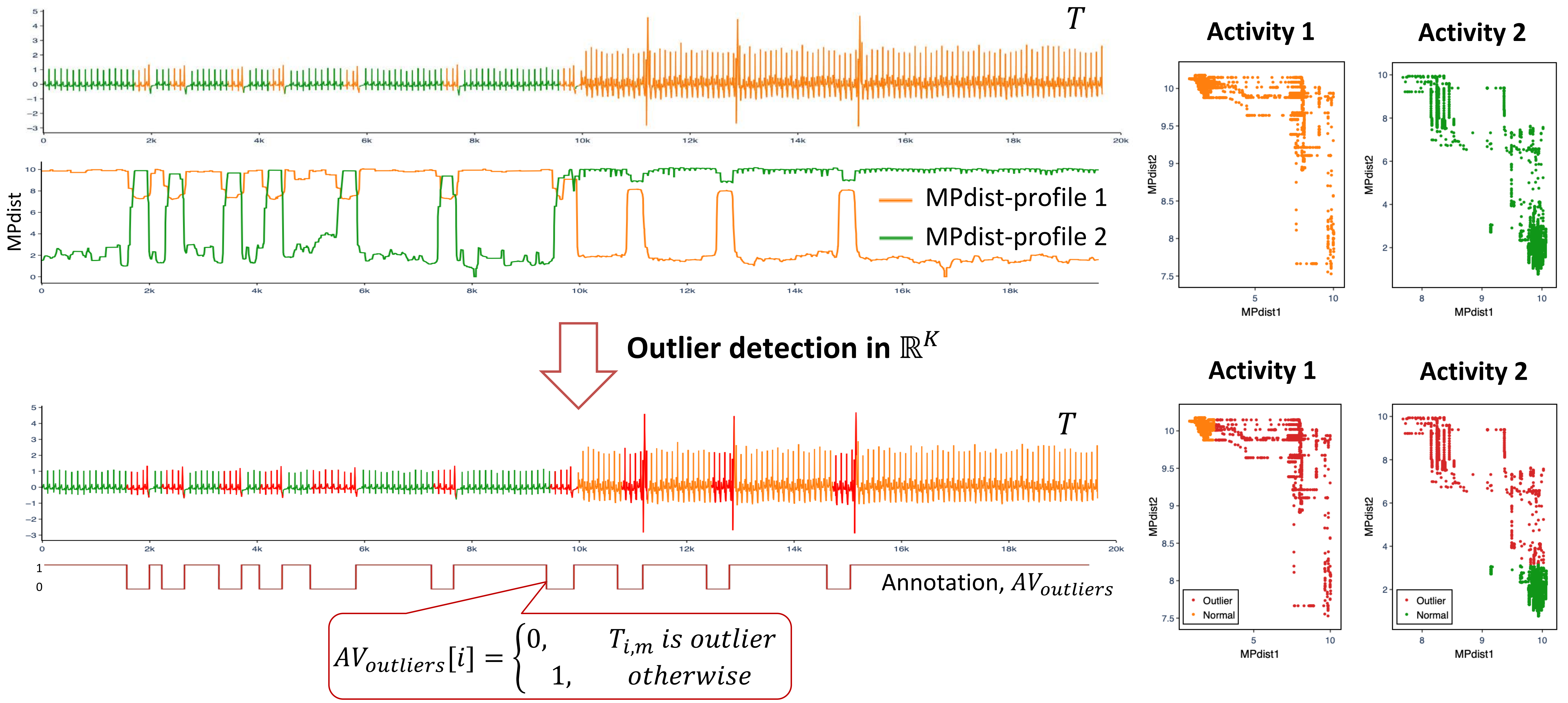
[1)] Imani S., et al. Introducing time series snippets: a new primitive for summarizing long time series. Data Min. Knowl. Discov. 34(6): 1713-1743 (2020). DOI: 10.1007/s10618-020-00702-y
[2)] Zymbler M., Goglachev A. Fast Summarization of Long Time Series with Graphics Processor. Mathematics 10(10). Article 1781. 2022. DOI: 10.3390/math10101781

DiSSiD: Cleaning by removing weak snippets (rare activities)

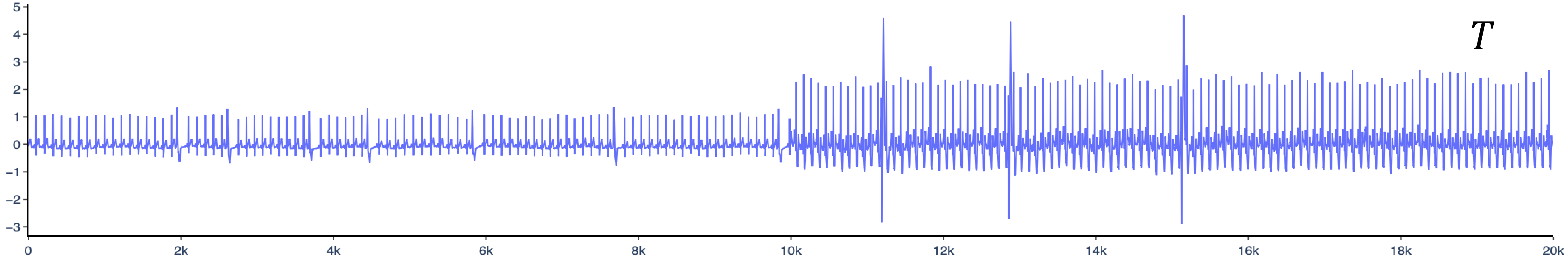




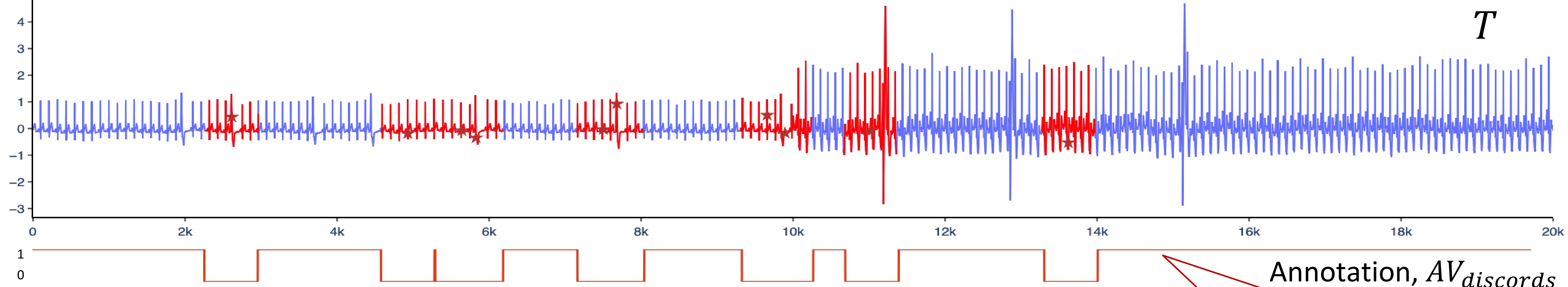
DiSSiD: Cleaning by removing outlier neighbors



DiSSiD: Cleaning by removing discords and their trivial matches



PALMAD[1], then remove discords found and subsequences located at $\pm m$ from discord

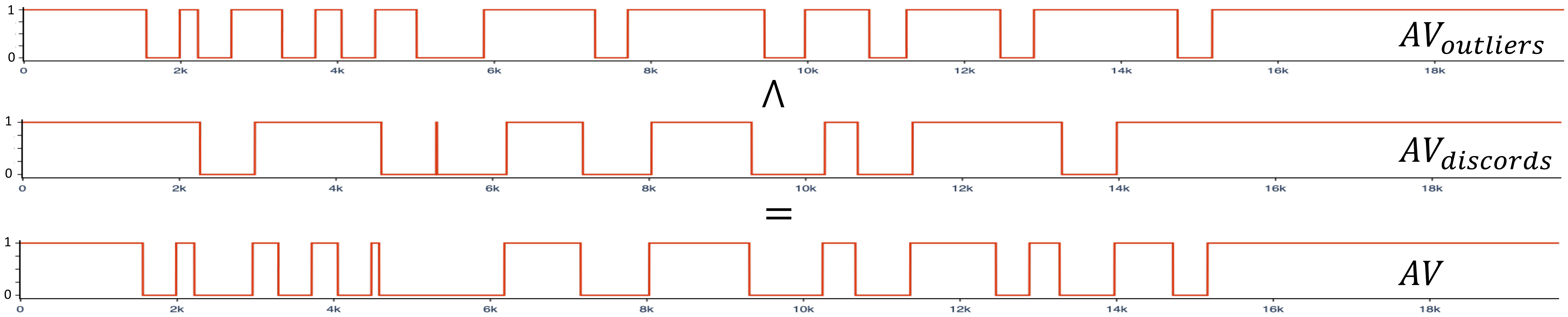


Annotation, $AV_{discords}$

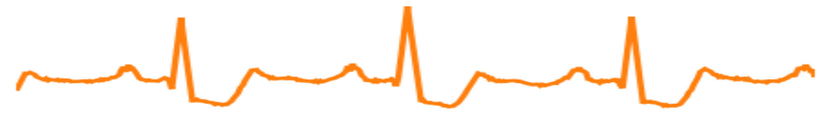
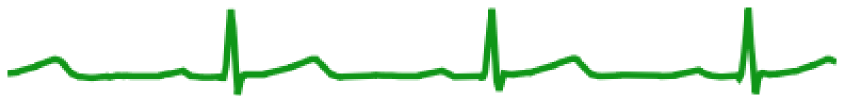
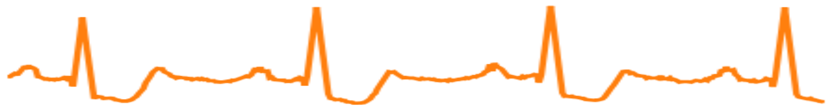
$$AV_{discords}[i] = \begin{cases} 0, & T_{i,m} \in D \\ 1, & otherwise \end{cases}$$

[1] Zymbler M., Kraeva Y. High-performance Time Series Anomaly Discovery on Graphics Processors. CoRR. 2023. Vol. abs/2304.01660. arXiv: 2304.01660.

DiSSiD: Cleaning

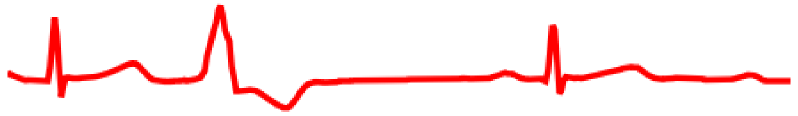


Labeled set L

Subsequence	Label
	1
	2
…	…
	1

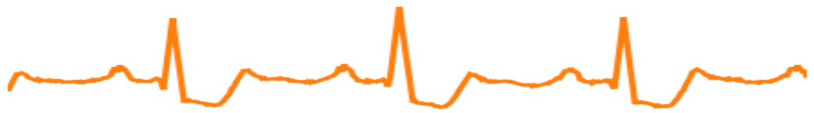
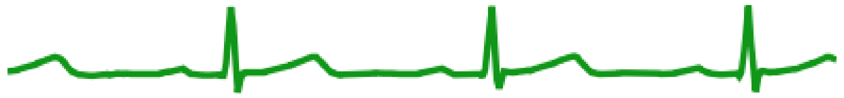
\

Anomaly set $\{s_i \in L \mid AV(i) = 0\}$

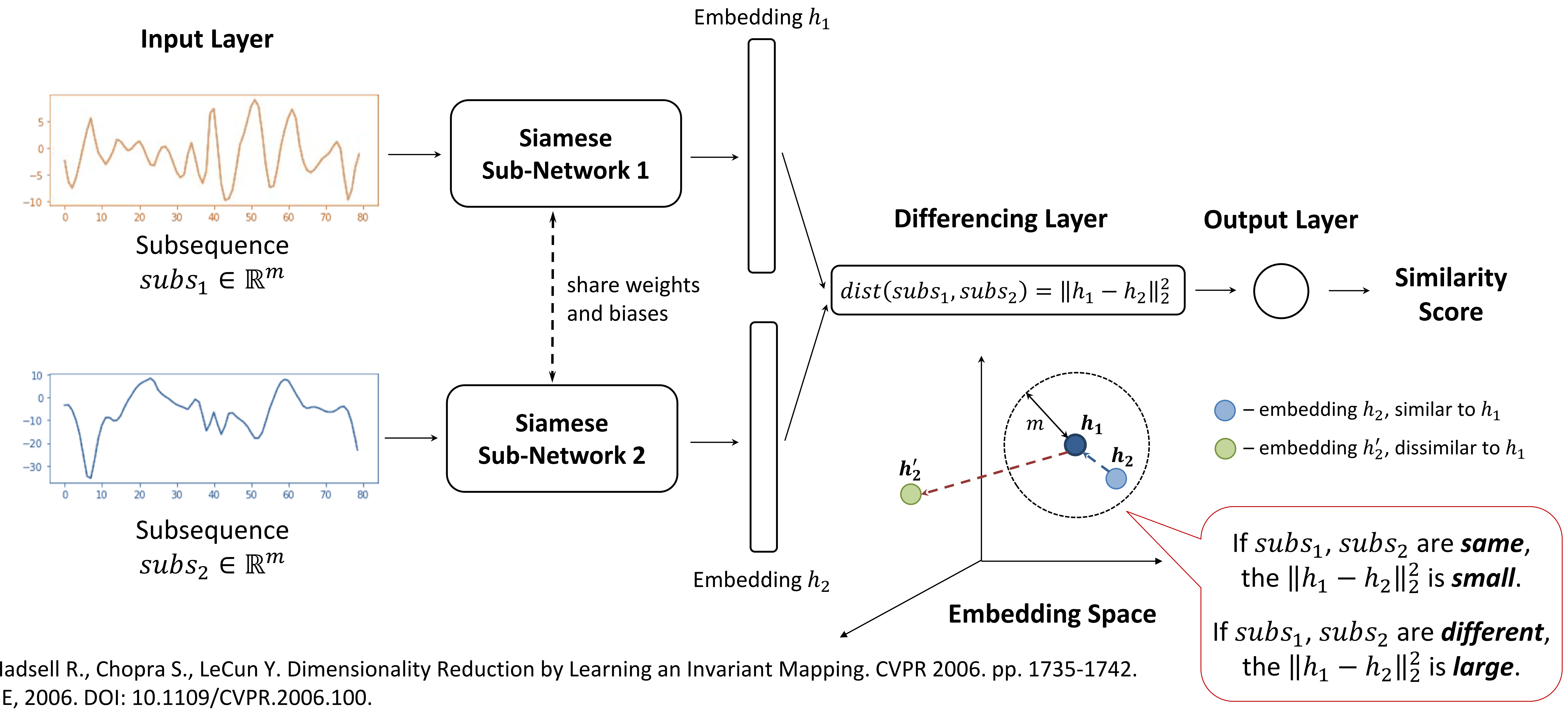
Subsequence

…


=

“Normalized” set L

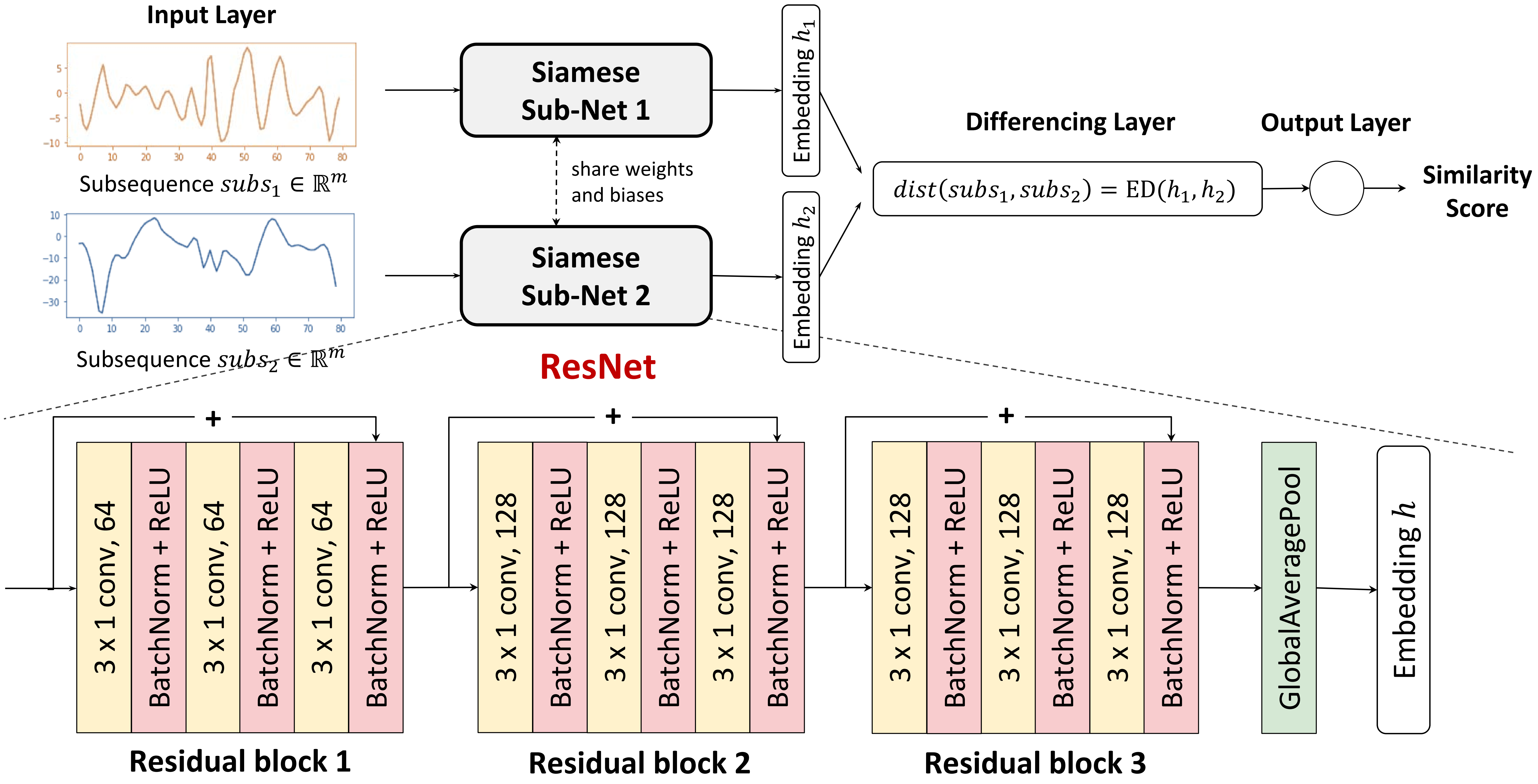
Subsequence	Label
	1
	2
…	…
	2

DiSSiD: Siamese Network*



* Hadsell R., Chopra S., LeCun Y. Dimensionality Reduction by Learning an Invariant Mapping. CVPR 2006. pp. 1735-1742. IEEE, 2006. DOI: 10.1109/CVPR.2006.100.

DiSSiD: ResNet*

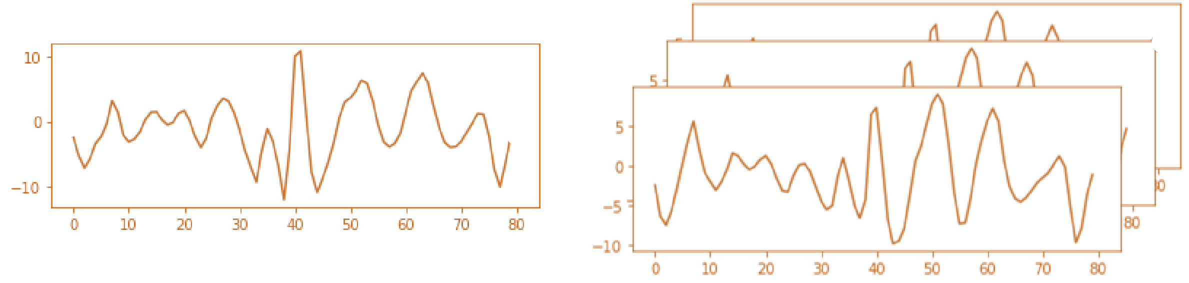


* Wang Z., et al. Time series classification from scratch with deep neural networks: A strong baseline. IJCNN 2017. pp. 1578-1585. IEEE, 2017. DOI: 10.1109/IJCNN.2017.7966039.

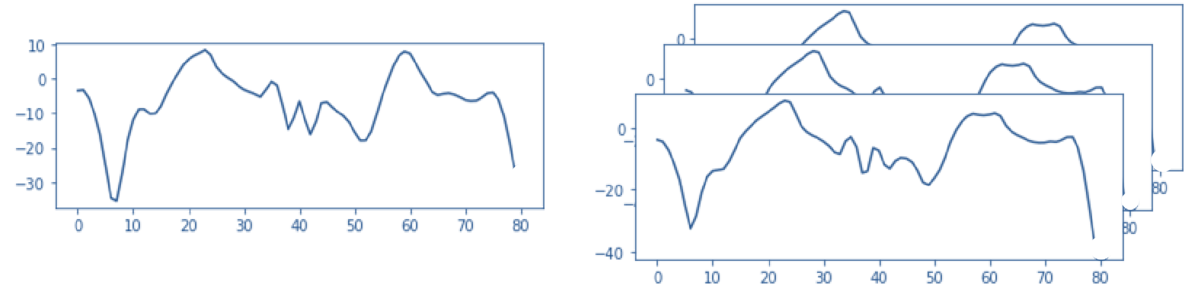
DiSSiD: Model training

“Normalized” set L

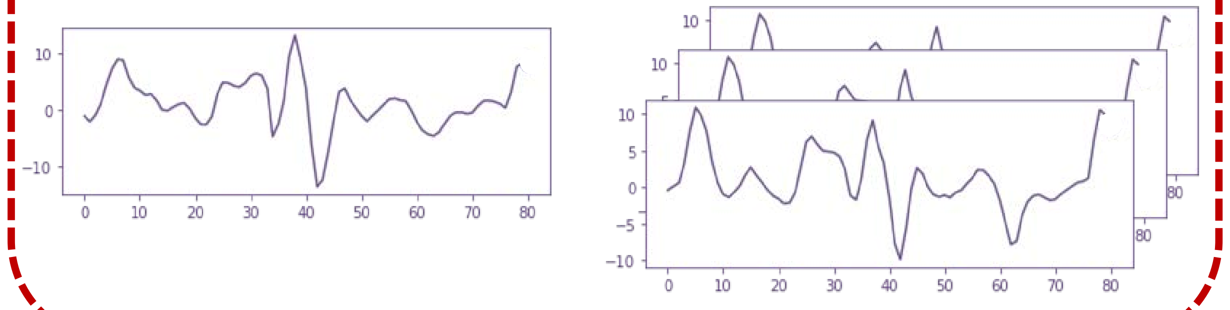
Snippet$_1$ and its **Nearest Neighbors**



Snippet$_2$ and its **Nearest Neighbors**

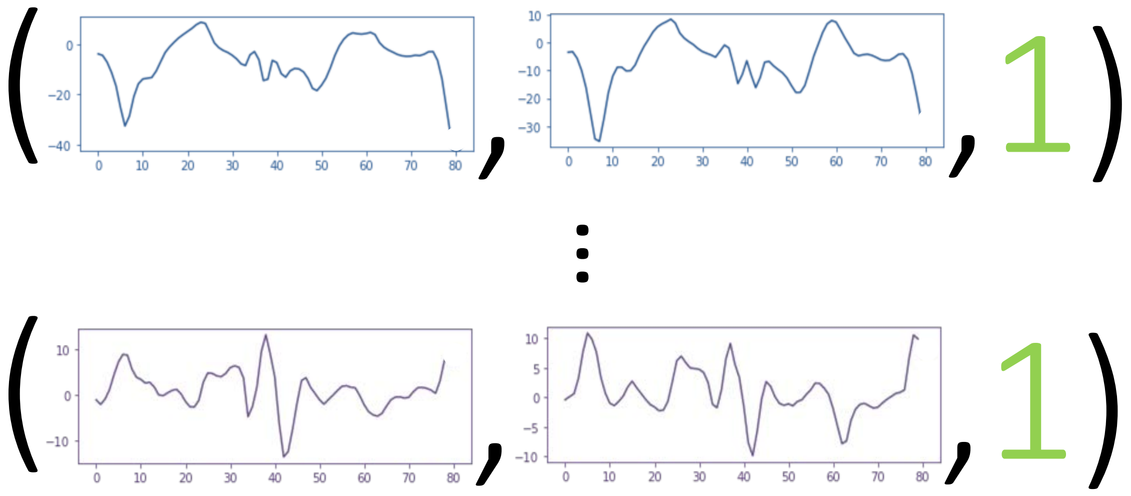


Snippet$_3$ and its **Nearest Neighbors**



Training set

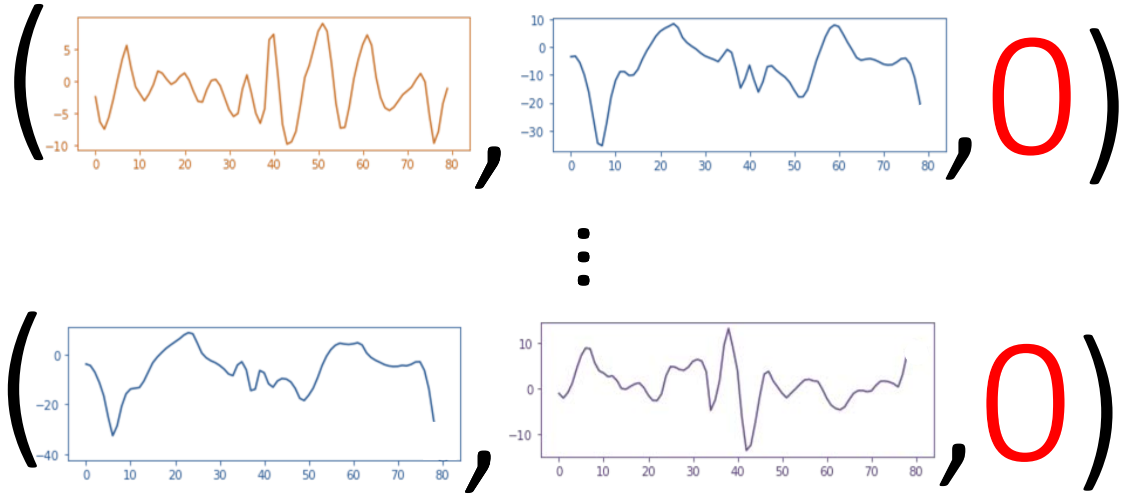
Positive samples



Max # **positive** pairs =

$$= \sum_{i=1}^{K} \binom{2}{|C_i.NN|}$$

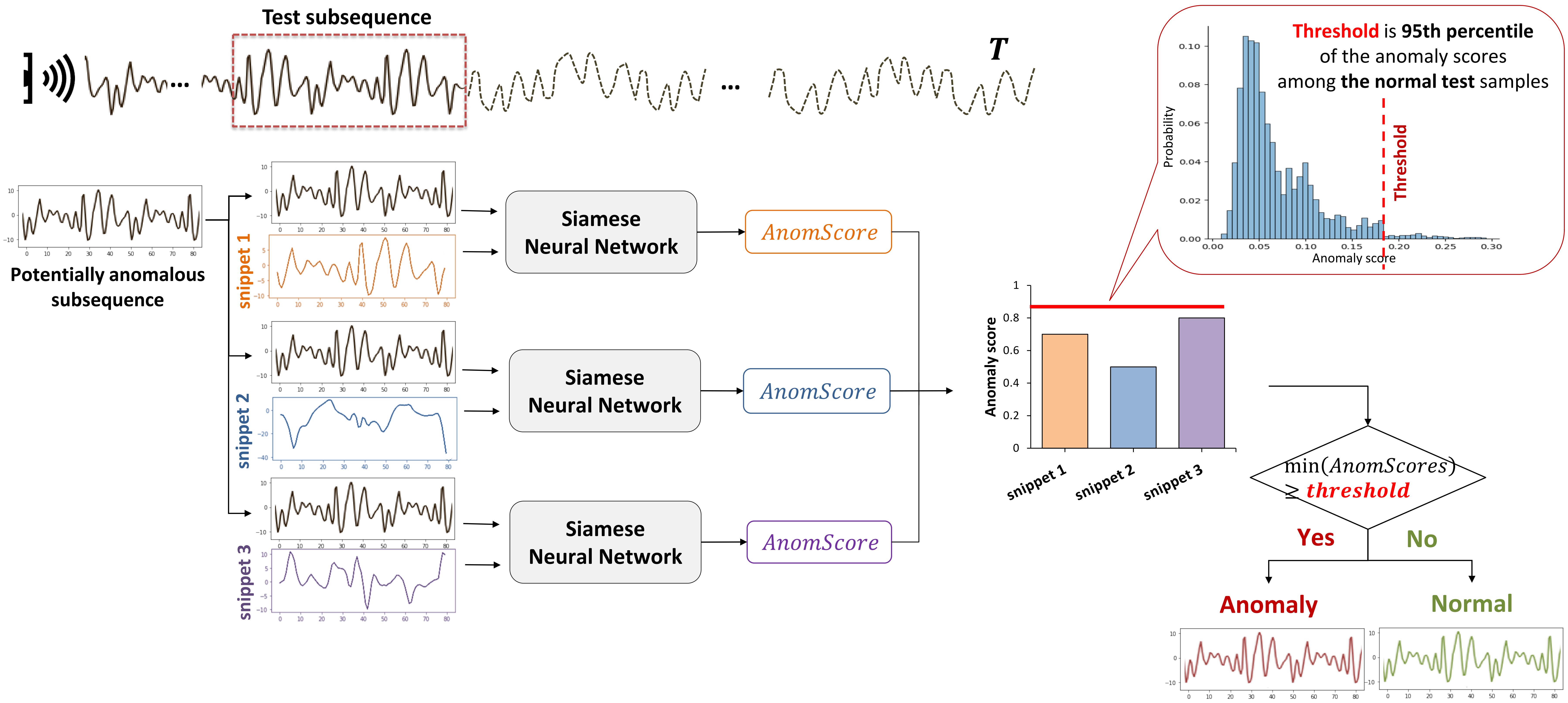
Negative samples



Max # **negative** pairs =

$$=\sum_{i=1}^{K} \sum_{j=i}^{K} |C_i\,.NN| \cdot |C_j.NN|$$

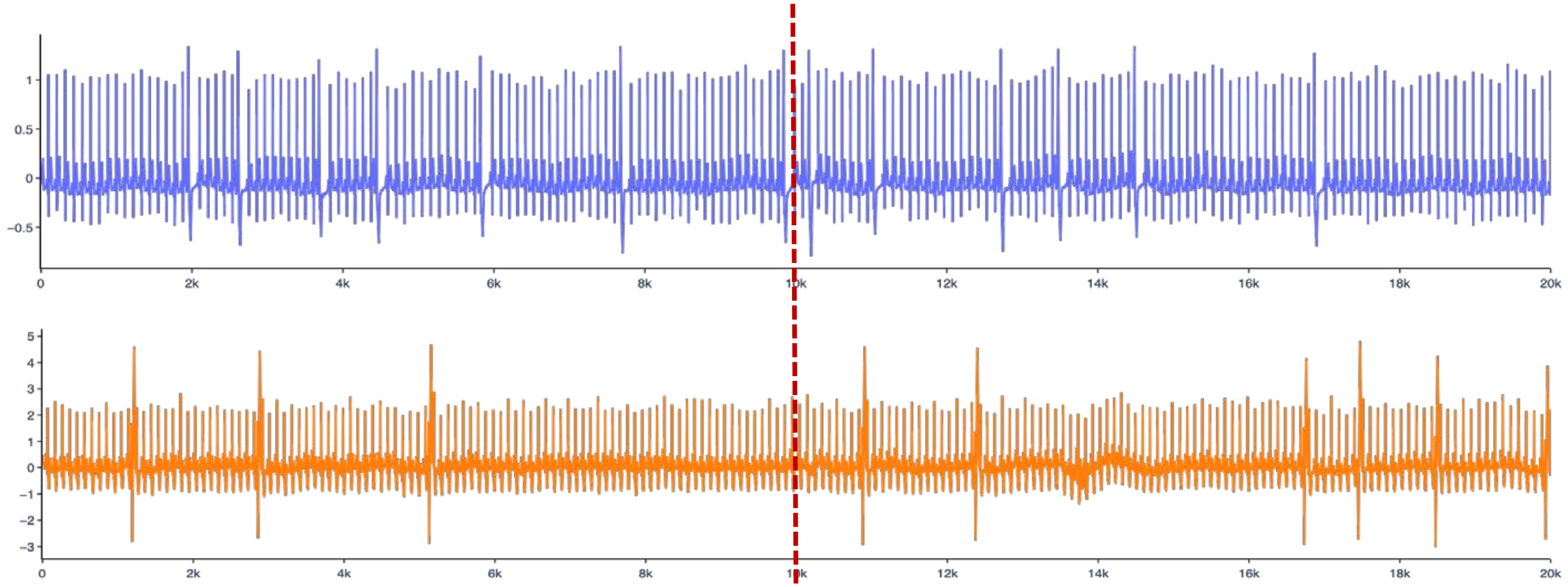
DiSSiD: Online anomaly detection



Experiments: Data generation

T_1

T_2

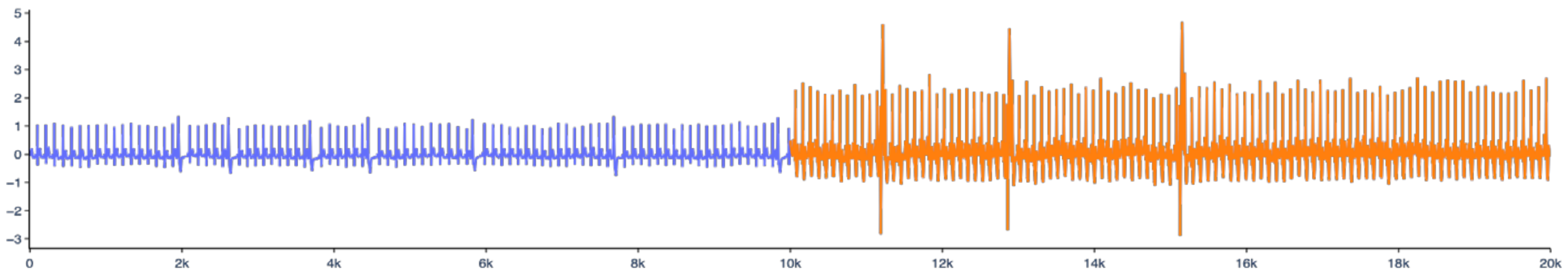


Train (50%) | **Test** (50%)

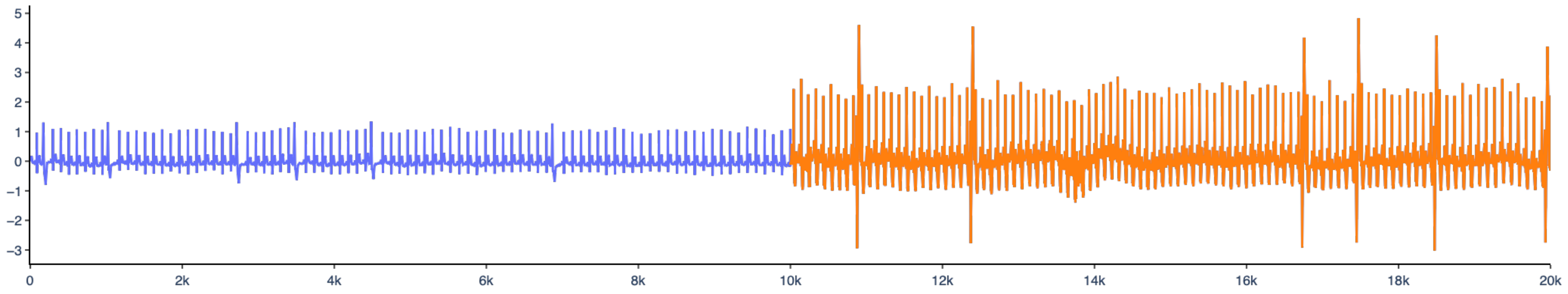
Concatenate
the corresponding parts
of time series

$$\boldsymbol{T_{train}} = T_1^1 \bullet T_2^1$$

$$\boldsymbol{T_{test}} = T_1^2 \bullet T_2^2$$



Train



Test

Experiments: Setup

- **Hardware:** Neurocomputer SUSU, NVIDIA Ampere A100 PCIe (6 912 CUDA-cores, 9.7 TFLOPS)
- **Data:** MIT-BIH Arrhythmia Database[1)]
  - For generating the labeled set:

Time series	Training time series length, n_{train}	Test time series length, n_{test}	Snippet length, m	Subsequence length, l	Snippet number, K	Discord fraction, α
ECG_803_805	$5 \cdot 10^4$	$5 \cdot 10^4$	350	75	2	0.0002
ECG_803_806	$5 \cdot 10^4$	$5 \cdot 10^4$	350	75	2	0.0002

  - For neural network learning: generate 170 000 pairs (2 positive + 2 negative pairs per 1 subsequence)
- **Measure:** $Precision = \frac{TP}{TP+FP}$, TP – true anomalies detected, FP – false positive anomalies detected
- **Neural Network Hyperparameters:**
  - **Loss function:** contractive loss

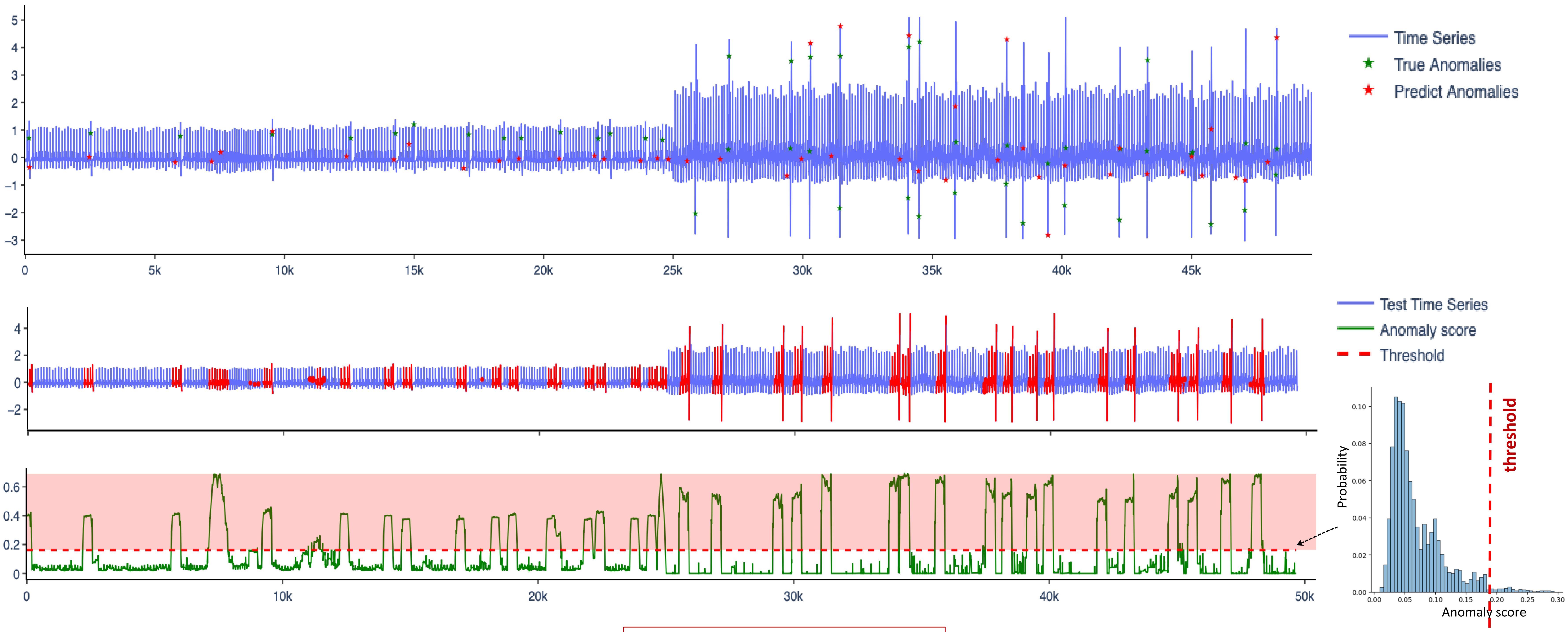
$$L(x_i, x_j, z_{i,j}) = z_{i,j}\mathrm{ED}^2(h_i, h_j) + (1 - z_{i,j})\max\left(0, \tau - \mathrm{ED}(h_i, h_j)\right)^2,$$

where τ – margin between samples of different classes

  - **Optimizer:** Adam
  - **# Epochs:** 40
  - **Batch size:** 128
  - **Learning rate:** $1 \cdot 10^{-3}$
  - **Embedding size:** 128

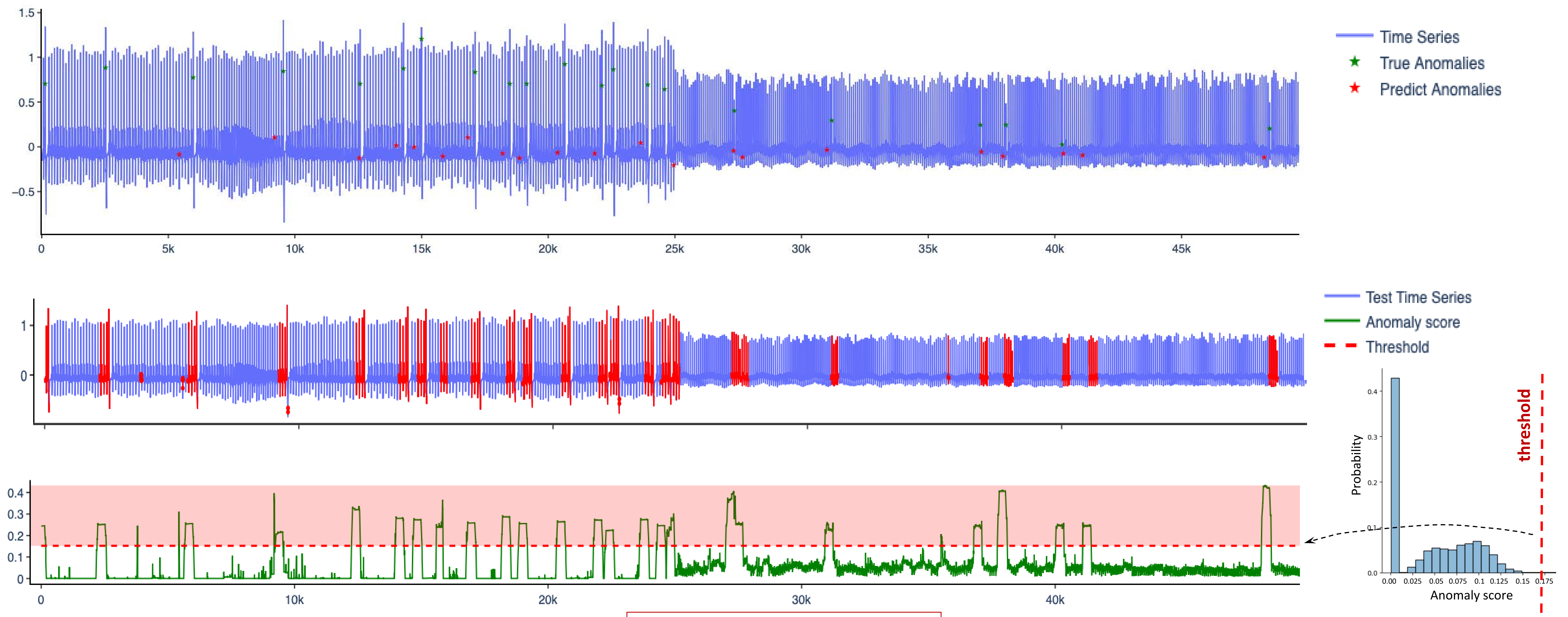
1) Goldberger A. L., et al. PhysioBank, PhysioToolkit, and PhysioNet components of a new research resource for complex physiologic signals. Circulation 101(23): 215-220. 2000.

Experiments: ECG_803_805



$$Precision = 0.915$$

Experiments: ECG_803_806



$$Precision = 0.81$$

Conclusions

- The results obtained:
  - parallel algorithm for discord discovery on HPC cluster with multi-GPU nodes
  - method for online anomaly detection
- Further research:
  - extensive experiments: time series from various domains, rivals, etc.
  - upgrade the method to make it adaptive

Thank you for your attention! Questions?

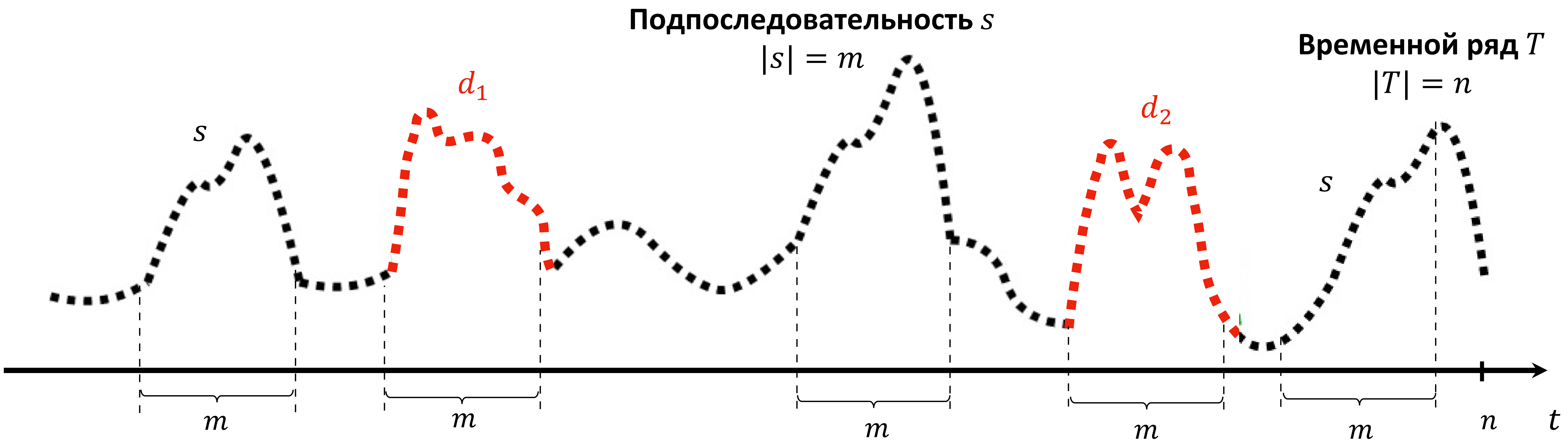
Kraeva Yana
kraevaya@susu.ru

Основные работы по теме исследования

Алгоритм	Платформа	Критика
Последовательный алгоритм		
Nakamura T., *et al.* **MERLIN**: parameter-free discovery of arbitrary length anomalies in massive time series archives. IEEE ICDM 2020. pp. 1190-1195.	CPU	Квадратичная сложность от длины ряда
Параллельные алгоритмы		
DRAG: Yankov D., *et al*. Disk aware discord discovery: finding unusual time series in terabyte sized datasets. Knowl. Inf. Syst. 17(2): 241-262. 2008.	CPU	Симуляция MapReduce
PDD: Huang T., *et al*. Parallel discord discovery. PAKDD 2016. LNCS 9652. Springer, 2016. pp. 233-244.	Spark	Низкая производительность ввиду большого количества обменов между узлами
PhiDD: Zymbler M., *et al.* A Parallel Approach to Discords Discovery in Massive Time Series Data. Computers, Materials & Continua 66(2): 1867-1876. 2021.	Кластер Intel Xeon Phi	Квадратичная пространственная сложность от длины ряда
KBF_GPU: Thuy T.T.H., *et al*. A new discord definition and an efficient time series discord detection method using GPUs. ICSED 2021. pp. 63-70.	GPU	Полный перебор подпоследовательностей ряда
Zhu B., et al. A GPU Acceleration framework for motif and discord based pattern mining. IEEE Trans. on Parallel and Distr. Systems 32(8): 1987-2004. 2021.	GPU	Поиск одного (самого важного) диссонанса ряда
PD3: Zymbler M., Kraeva Ya. Parallel algorithm for time series discord discovery on a graphics processor. Pattern Recognition and Image Analysis 33(2). 2023.	GPU	Ручной подбор длины диссонанса и порога

Постановка задачи

- **Диссонанс**[1)] – подпоследовательность ряда, расстояние от которой до ближайшего соседа не ниже порога r
- **Дано:** временной ряд T, длина диссонанса m, порог r
- **Найти:** $D = \{d_1, d_2, \ldots\}$, $d_i \in D \Leftrightarrow \forall s \in T \min_{s \cap d_i = \emptyset} \text{dist}(d_i, s) \geq r$



Количество подпоследовательностей: $N = n - m + 1$

[1)] Yankov D., Keogh E.J., Rebbapragada U. Disk aware discord discovery: finding unusual time series in terabyte sized datasets. Knowl. Inf. Syst. 17(2): 241-262. 2008.

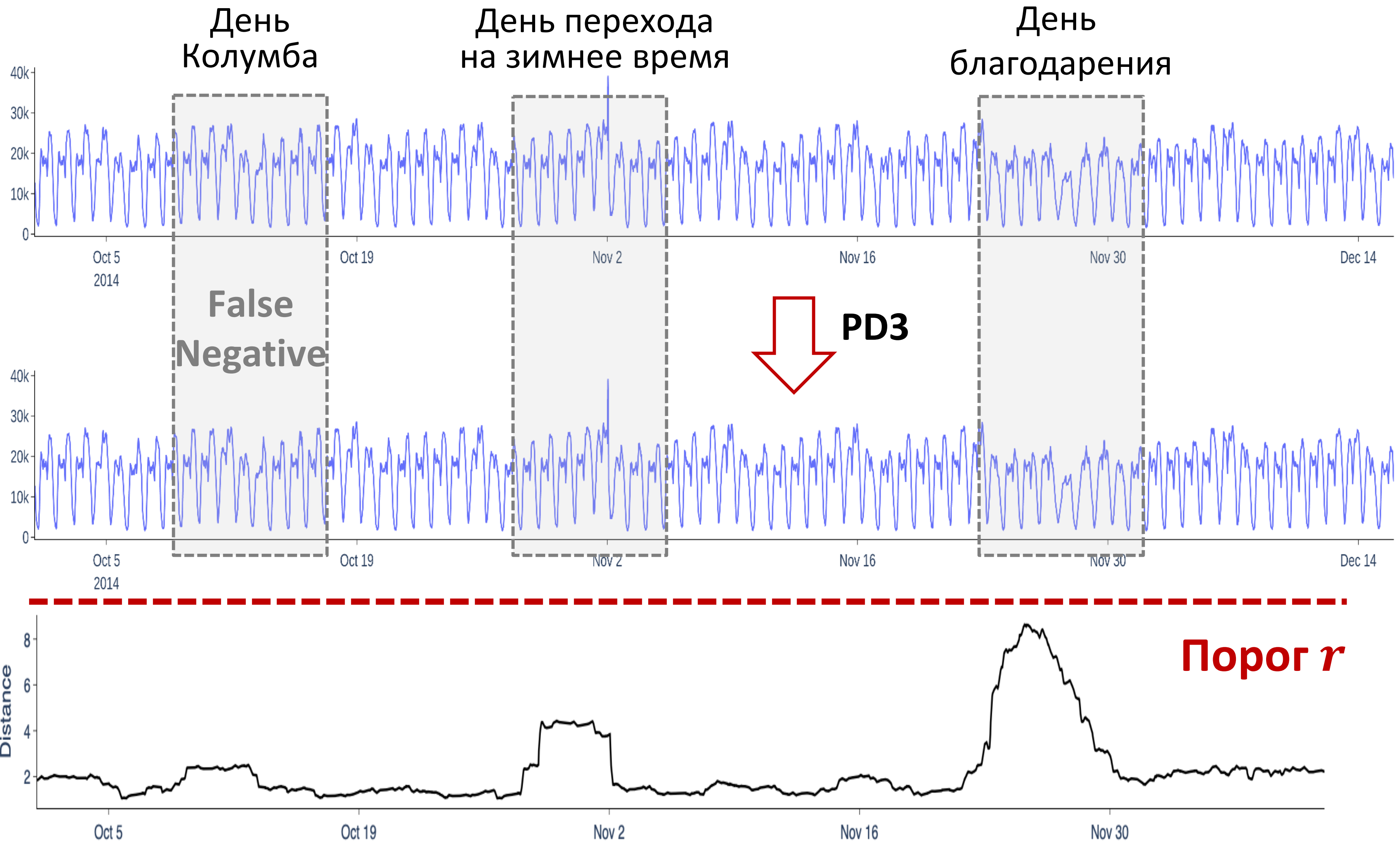
PD3 (Parallel DRAG-based Discord Discovery): Ручной подбор r

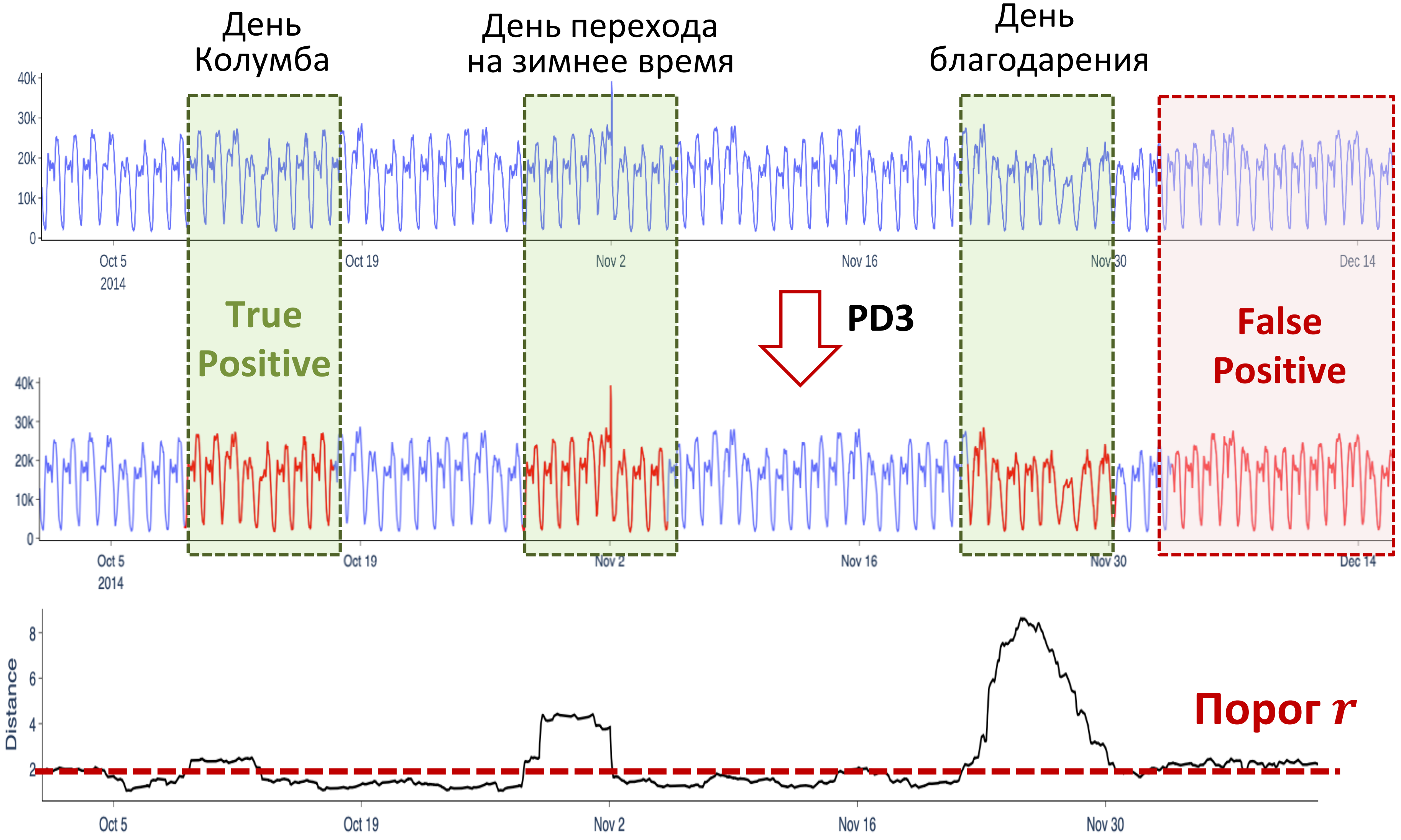
Порог r

Большой → **НЕТ** диссонансов

Маленький → Много **ЛОЖНЫХ** диссонансов

Среднее число пассажиров NY такси (осень 2014 г., каждые полчаса)

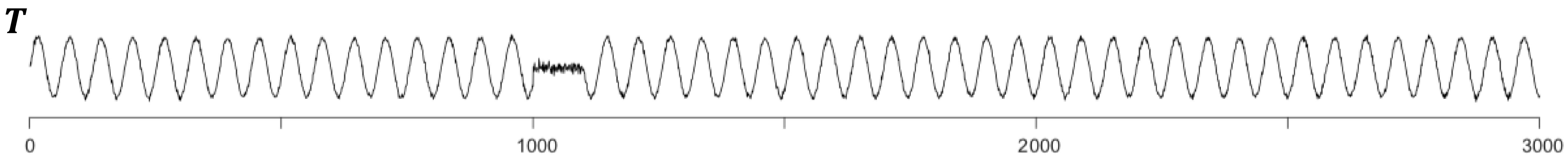




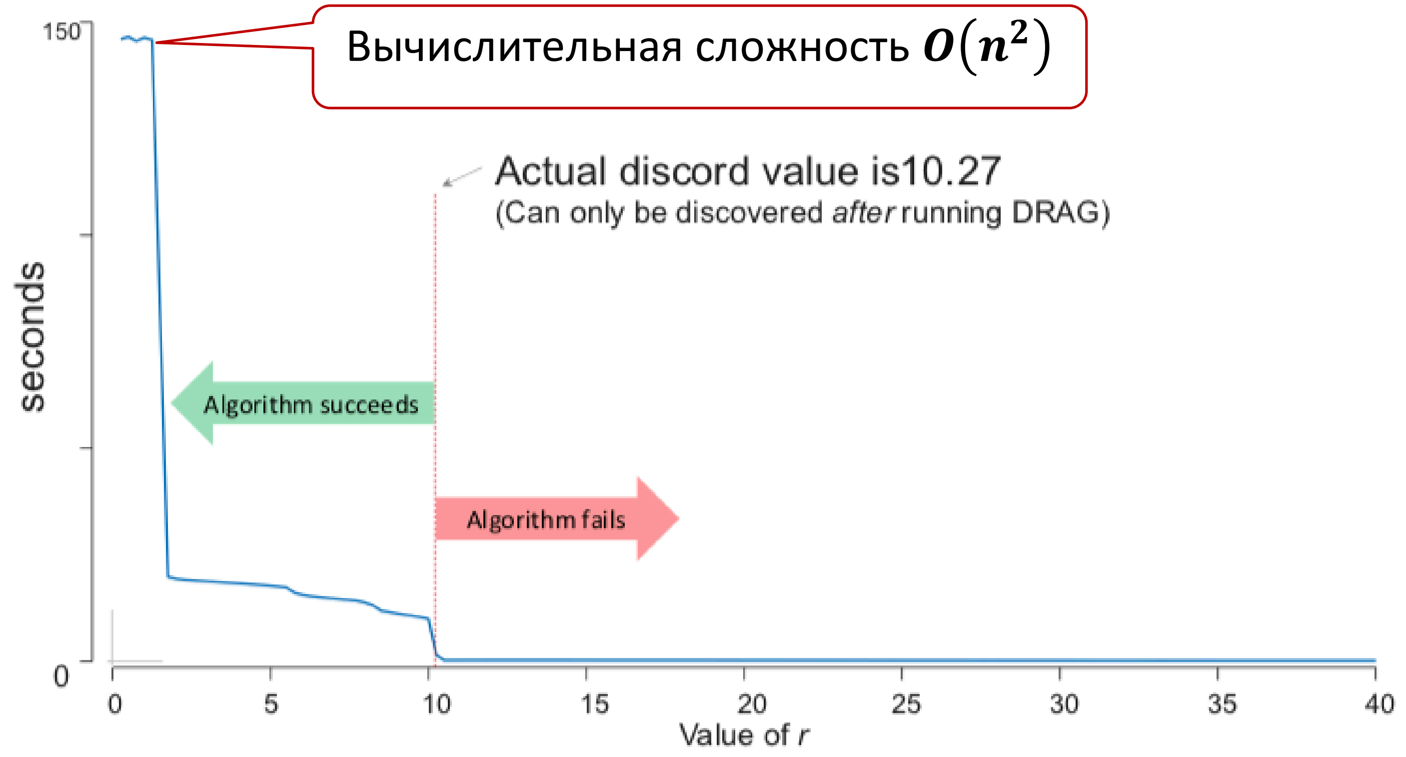
PD3 (Parallel DRAG-based Discord Discovery): Ручной подбор m

Накладные расходы на подбор параметров m и r

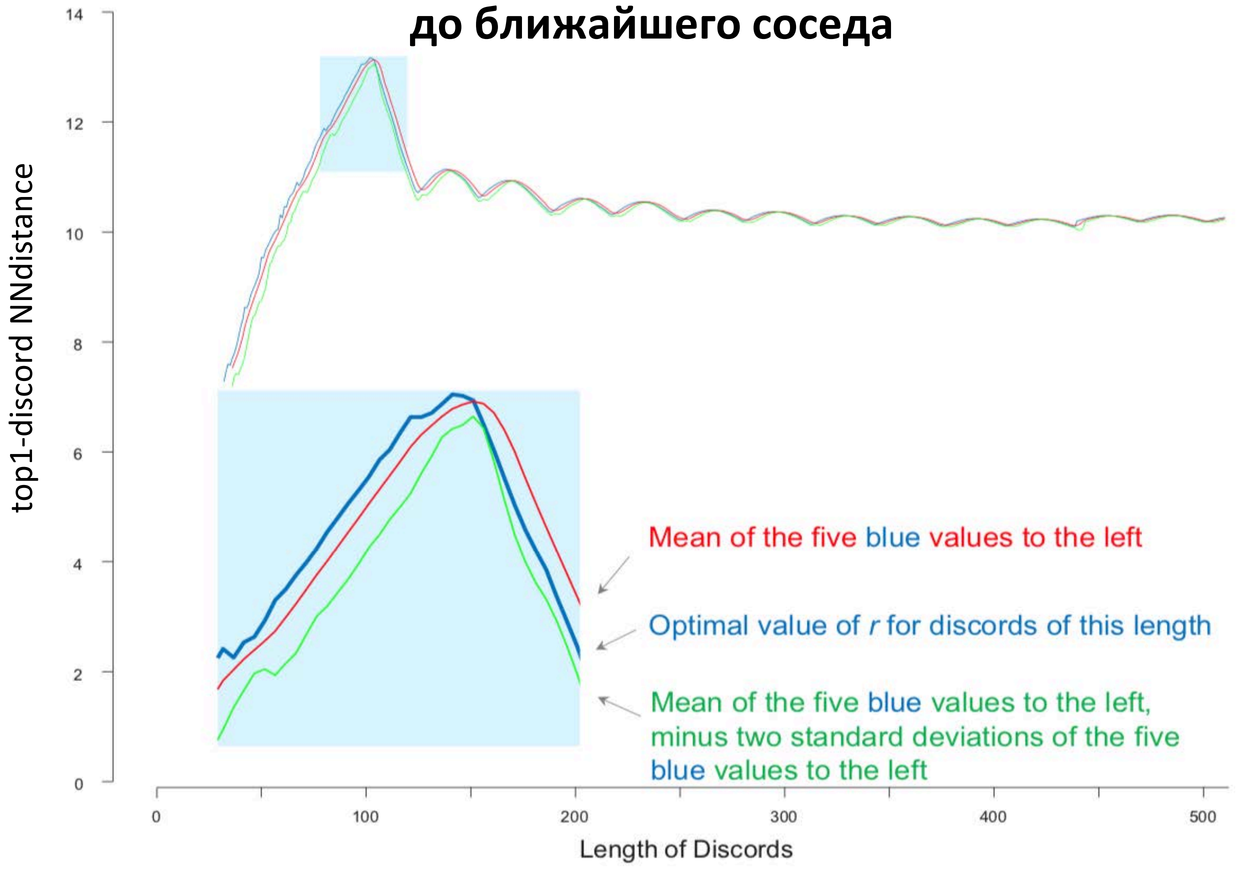
Подбор порога r



Влияние величины порога r на время выполнения DRAG



Влияние длины диссонанса на расстояние диссонанса до ближайшего соседа



Длина диссонанса, m	Порог, r
$minL$	$r = 2\sqrt{minL}$
$minL+1, \ldots, minL+4$	$r = 0.99 * nndist(m-1)$
$minL+5, \ldots, maxL$	$r = \mu - 2\sigma$

PALMAD: Parallel Arbitrary Length MERLIN-based Anomaly Discovery

1. Применение $\mathrm{ED}^2_{\mathrm{norm}}$ в качестве функции расстояния[1)]

$$\mathrm{ED}^2_{\mathrm{norm}}(T_{i,m}, T_{j,m}) = 2m\left(1 - \frac{T_{i,m} \cdot T_{j,m} - m\mu_i\mu_j}{m\sigma_i\sigma_j}\right)$$

2. Сокращение избыточных вычислений μ и σ при вычислении $\mathrm{ED}^2_{\mathrm{norm}}$

Лемма. Пусть даны ряд T, $|T| = n$ и подпоследовательности $T_{i,m}$ и $T_{i,m+1}$. Тогда

$$\mu_{T_{i,m+1}} = \frac{1}{m+1}\left(m\mu_{T_{i,m}} + t_{i+m}\right), \qquad \sigma^2_{T_{i,m+1}} = \frac{m}{m+1}\left(\sigma^2_{T_{i,m}} + \frac{1}{m+1}\left(\mu_{T_{i,m}} - t_{i+m}\right)^2\right).$$

3. Автоматизированный подбор порога r

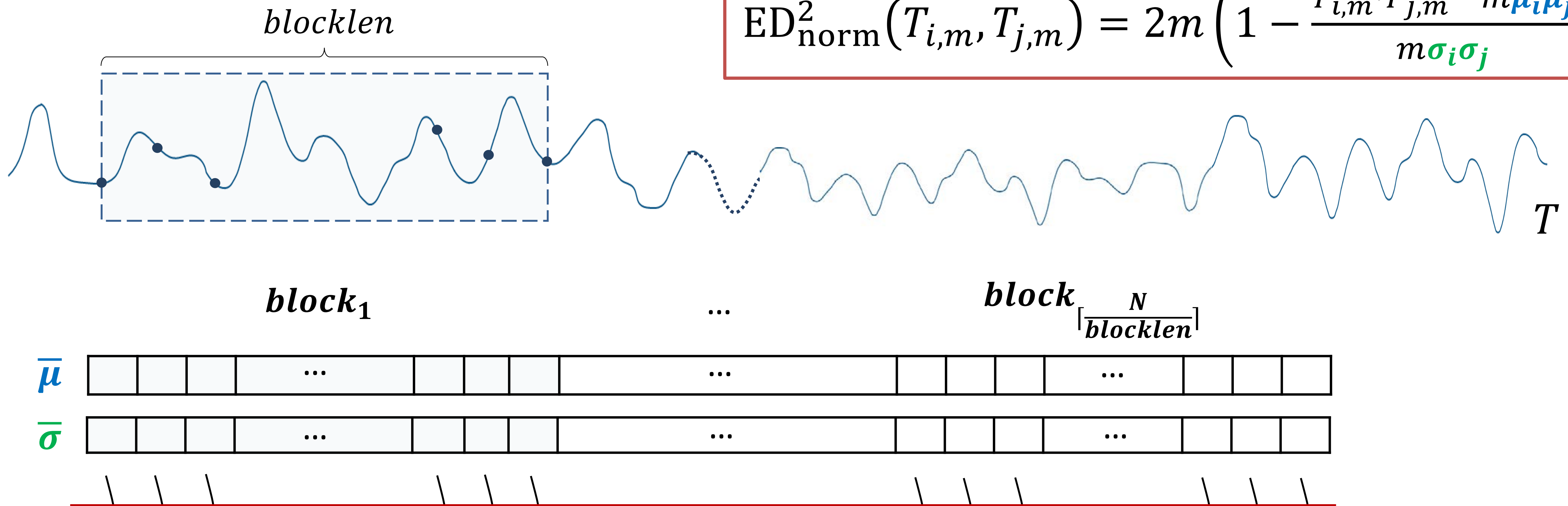
4. Тепловая карта диссонансов

[1)] Mueen A. et al. Fast approximate correlation for massive time-series data. SIGMOD 2010. pp. 171-182. ACM (2010). https://doi.org/10.1145/1807167.1807188

Сокращение избыточных вычислений $\overline{\mu}$ и $\overline{\sigma}$

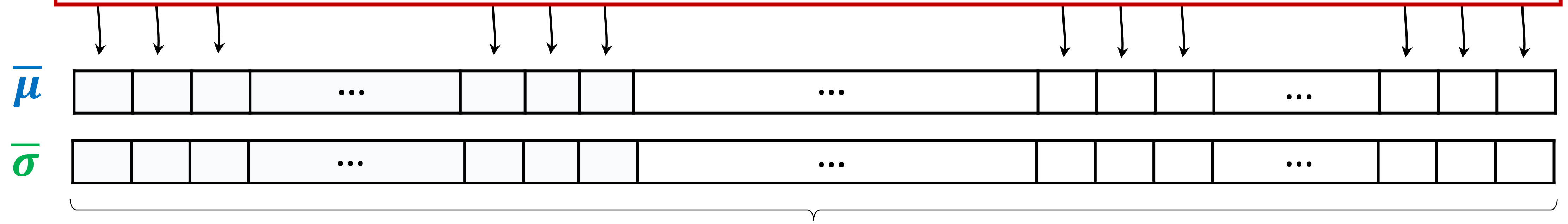
$$\mathrm{ED}^2_{\mathrm{norm}}(T_{i,m}, T_{j,m}) = 2m\left(1 - \frac{T_{i,m} \cdot T_{j,m} - m\mu_i\mu_j}{m\sigma_i\sigma_j}\right)$$

Вычисленные $\overline{\mu}$ и $\overline{\sigma}$ для подп-ей ряда длины, меньшей на 1



Лемма

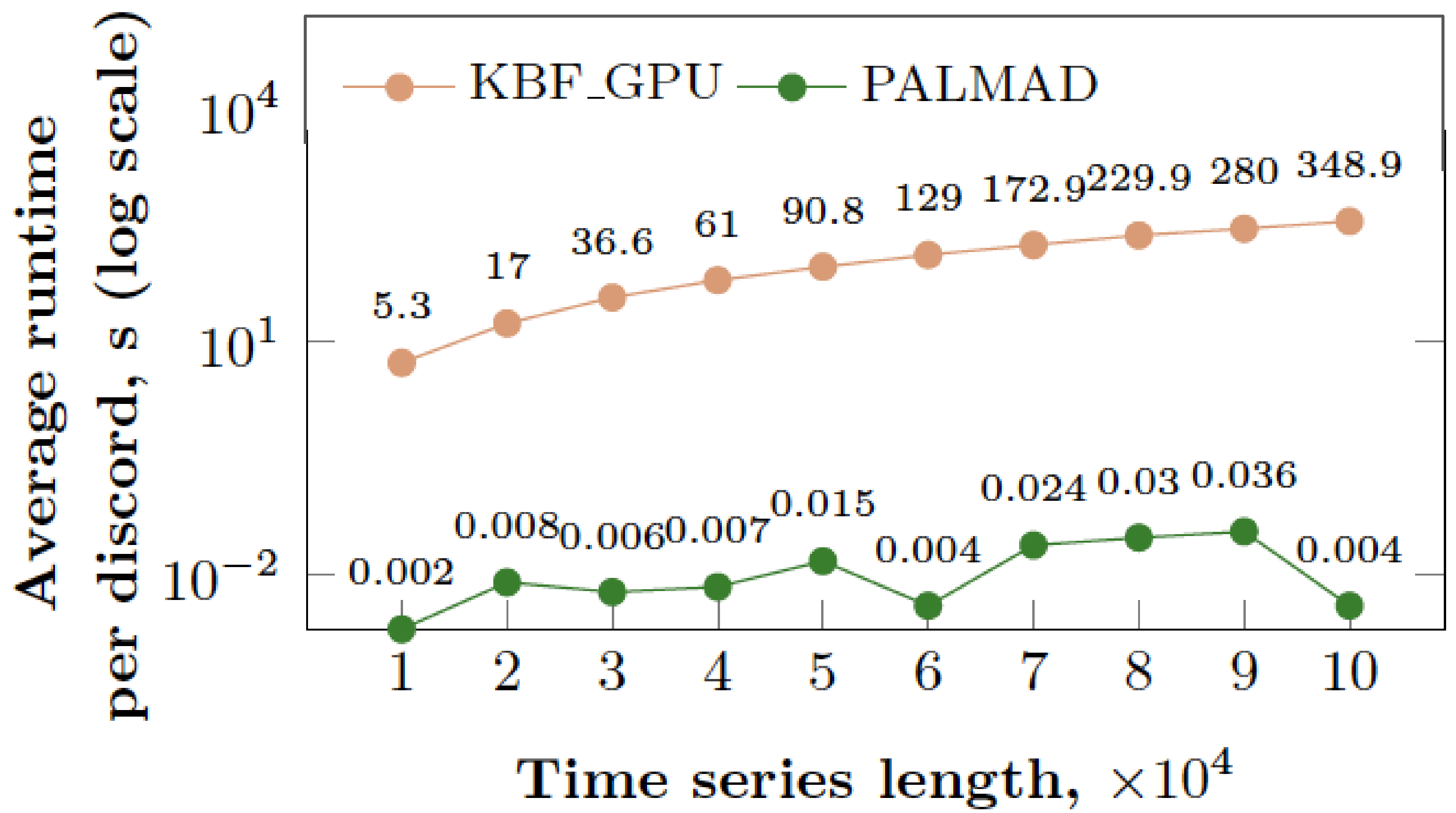
$$\mu_{T_{i,m+1}} = \frac{1}{m+1}\left(m\mu_{T_{i,m}} + t_{i+m}\right), \quad \sigma^2_{T_{i,m+1}} = \frac{m}{m+1}\left(\sigma^2_{T_{i,m}} + \frac{1}{m+1}\left(\mu_{T_{i,m}} - t_{i+m}\right)^2\right)$$



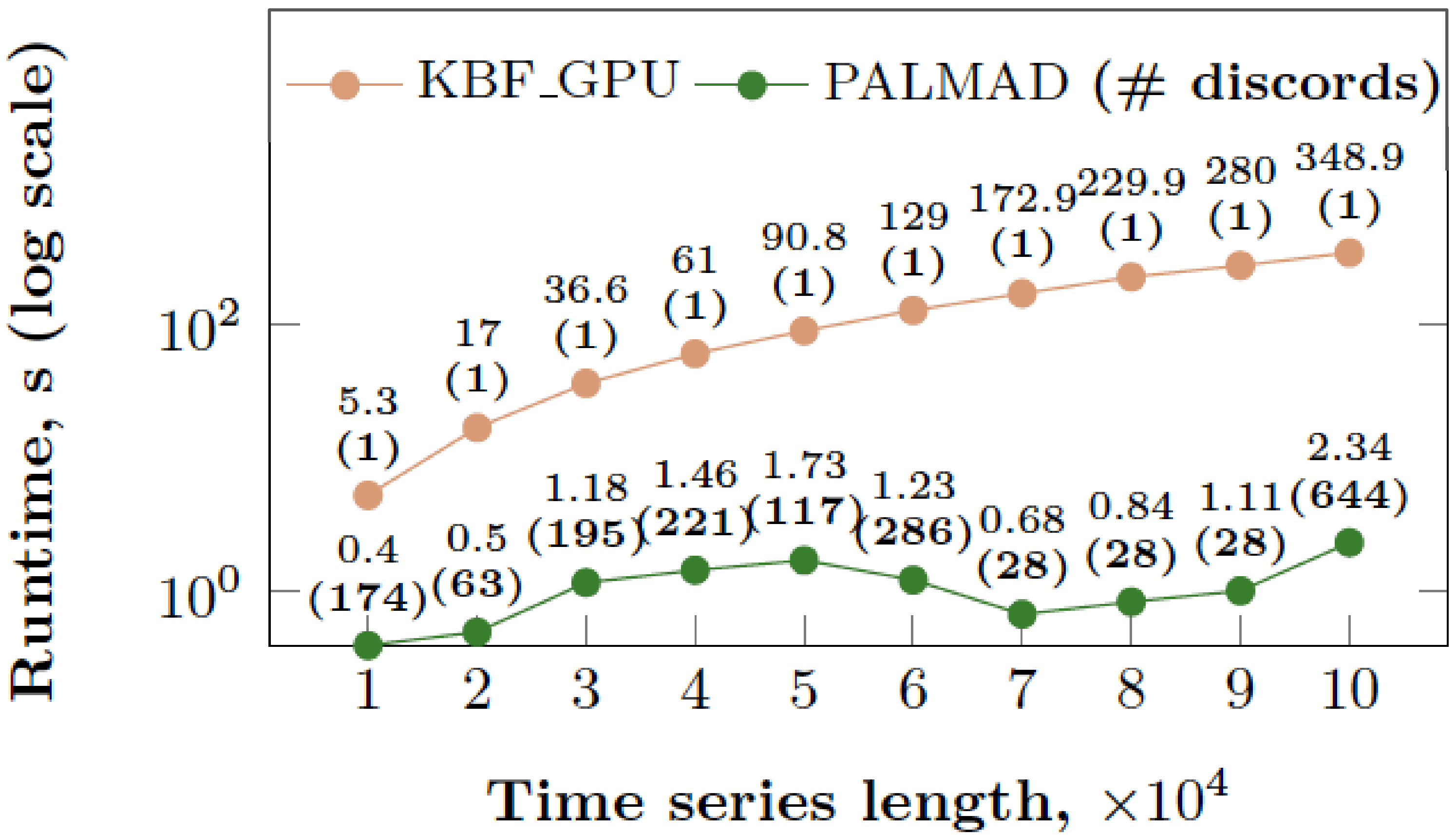
$$N = n - minL + 1$$

Производительность: сравнение с KBF_GPU[1)]

Среднее время на поиск **одного** диссонанса



Время на поиск **всех** диссонансов

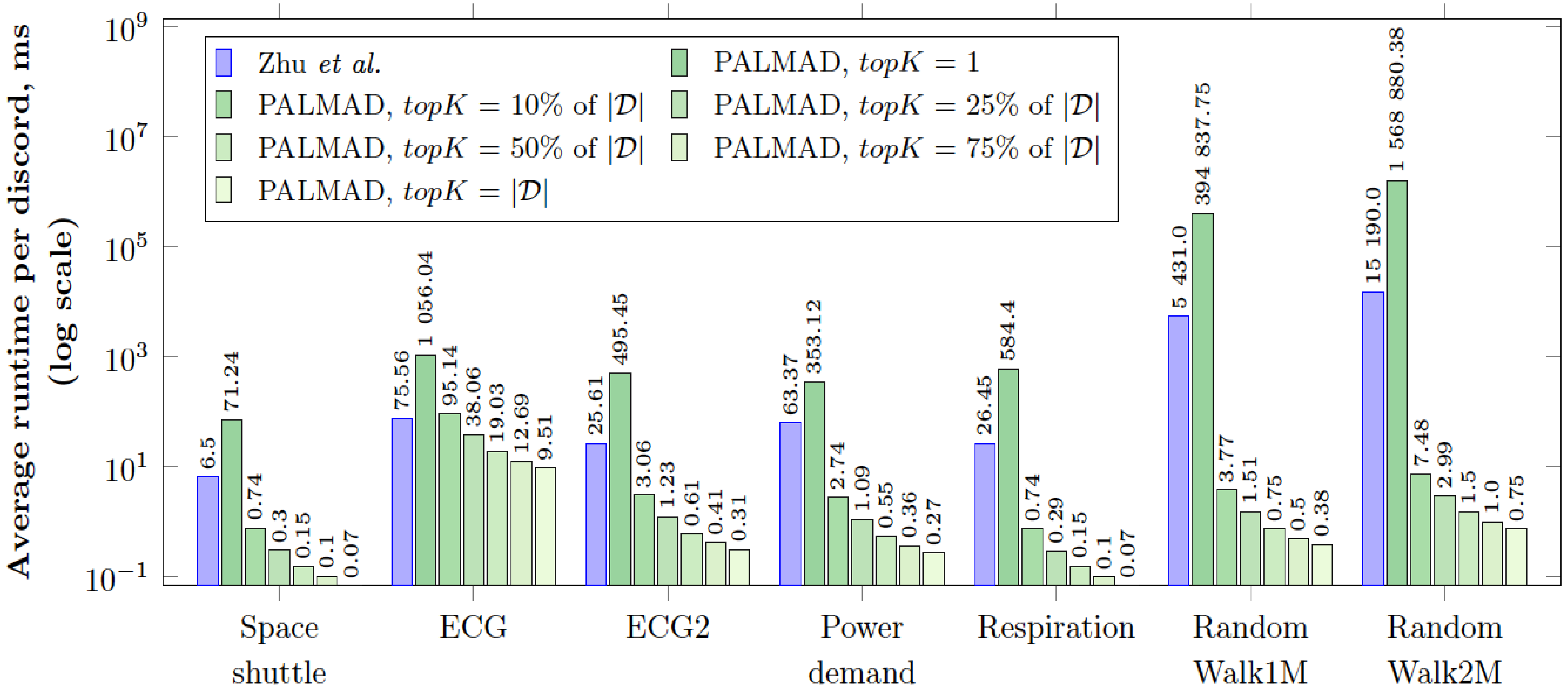


PALMAD опережает KBF_GPU как по общему времени работы, так и по среднему времени на поиск одного диссонанса

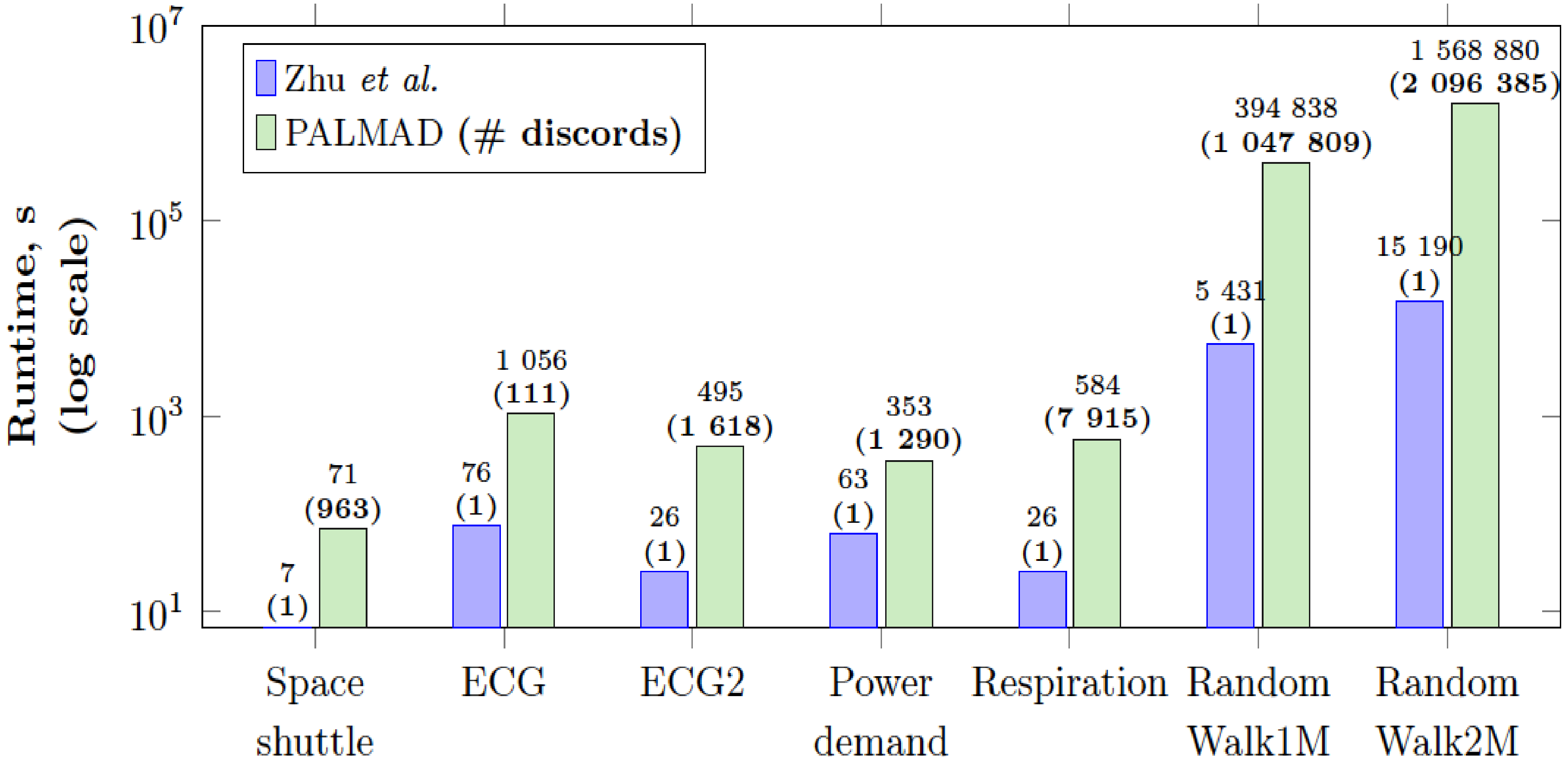
[1)] Thuy T.T.H. et al. A new discord definition and an efficient time series discord detection method using GPUs. ICSED 2021. pp. 63–70. https://doi.org/10.1145/3507473.3507483.

Производительность: сравнение с Zhu et al.[1)]

Среднее время на поиск **одного** диссонанса



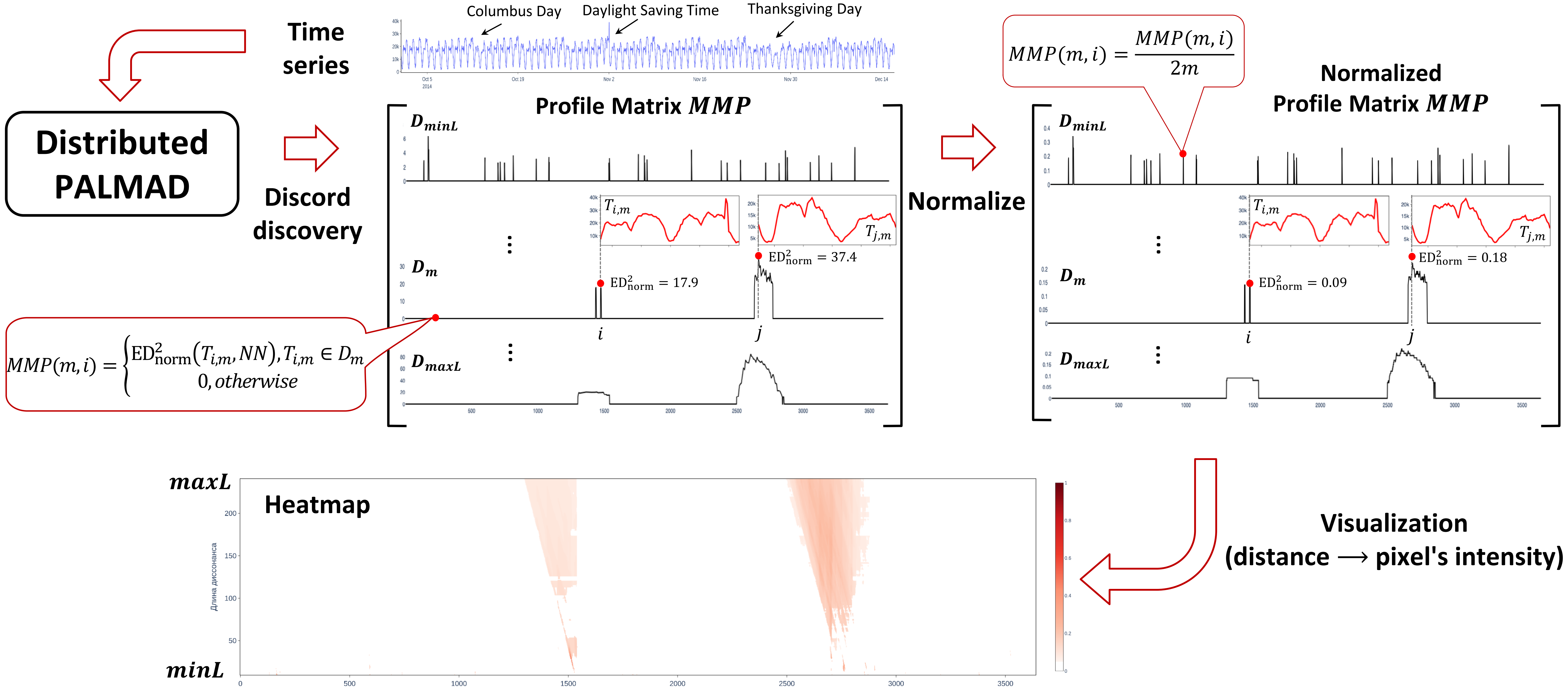
Время на поиск **всех** диссонансов



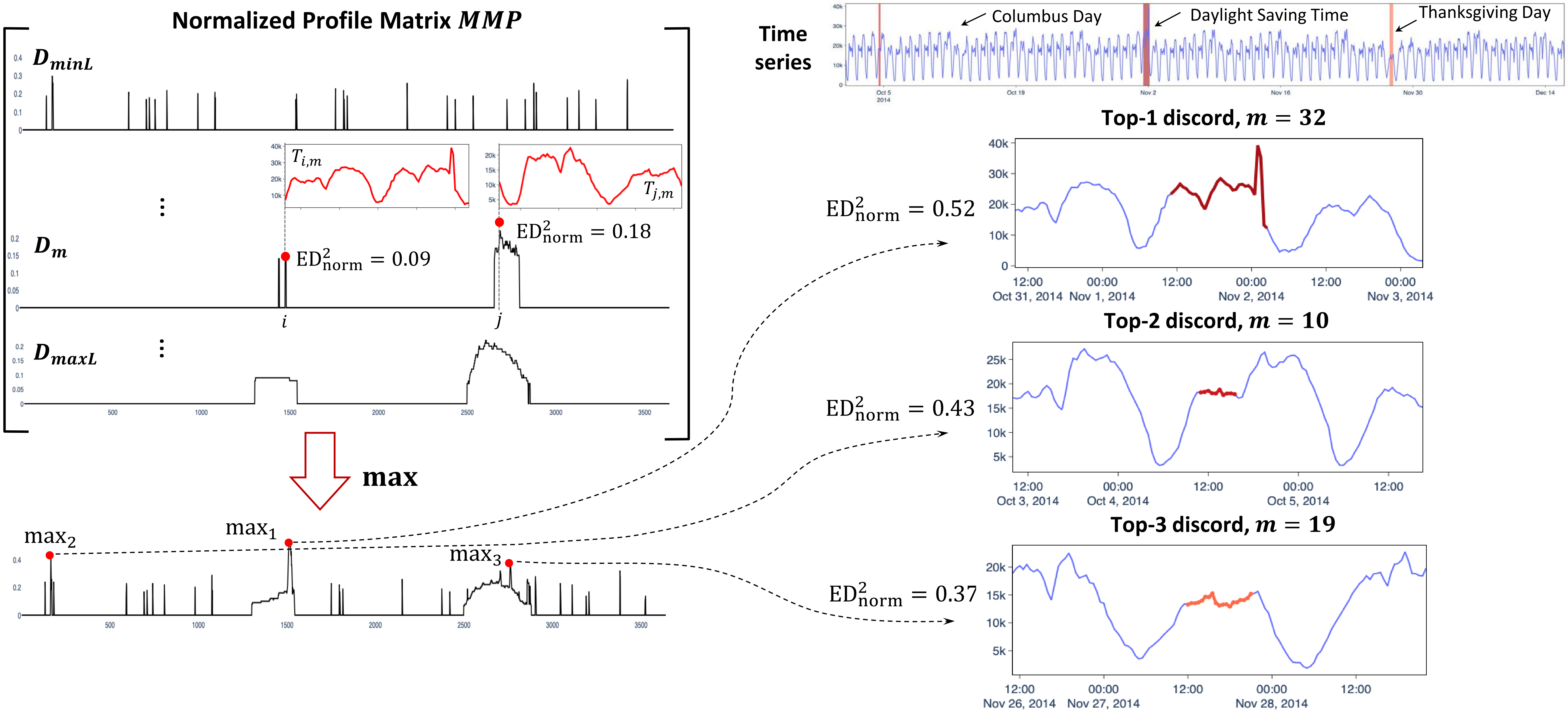
PALMAD значительно опережает алгоритм Zhu et al. по среднему времени на поиск одного диссонанса, начиная с поиска top-k диссонансов, где k=10% от фактического числа диссонансов

[1)] Zhu B. et al. A GPU Acceleration framework for motif and discord based pattern mining. IEEE Transactions on Parallel and Distributed Systems 32(8): 1987-2004. 2021. https://doi.org/10.1109/TPDS.2021.3055765.

Discord Heatmap

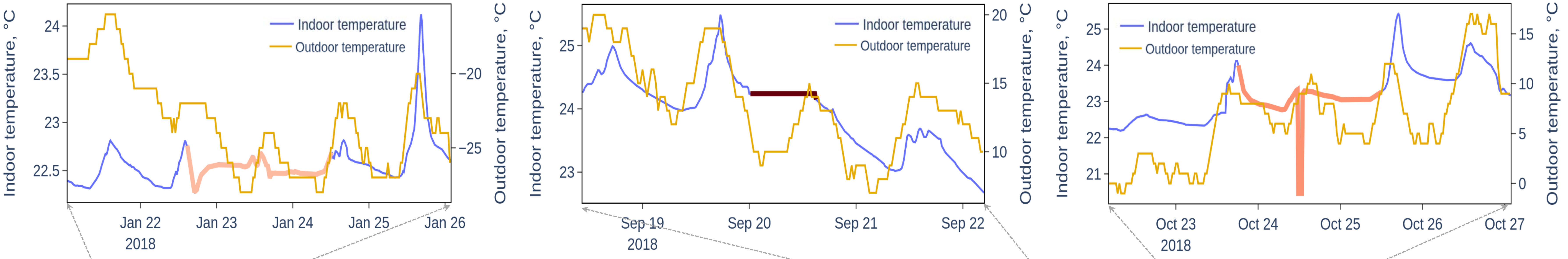


Discord ranking

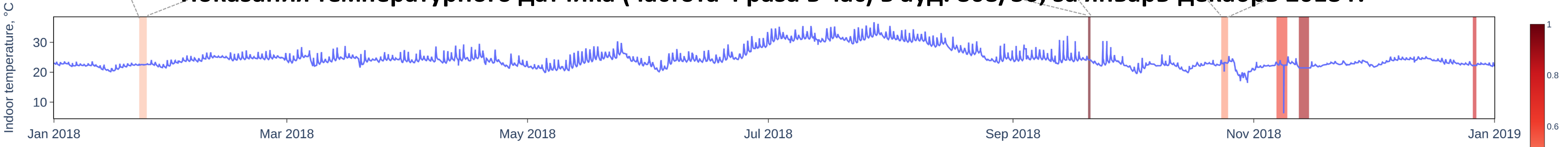


Выявление аномалий в системе отопления ЮУрГУ

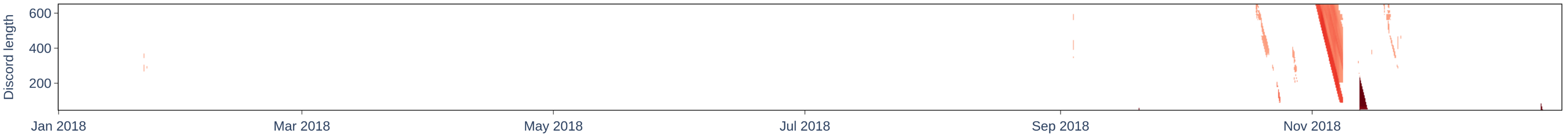
Примеры найденных аномалий длительностью 0.5-2 суток



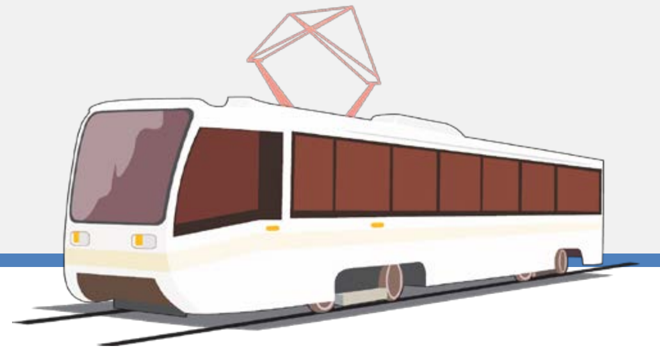
Показания температурного датчика (частота 4 раза в час) в ауд. 808/3б, за январь-декабрь 2018 г.



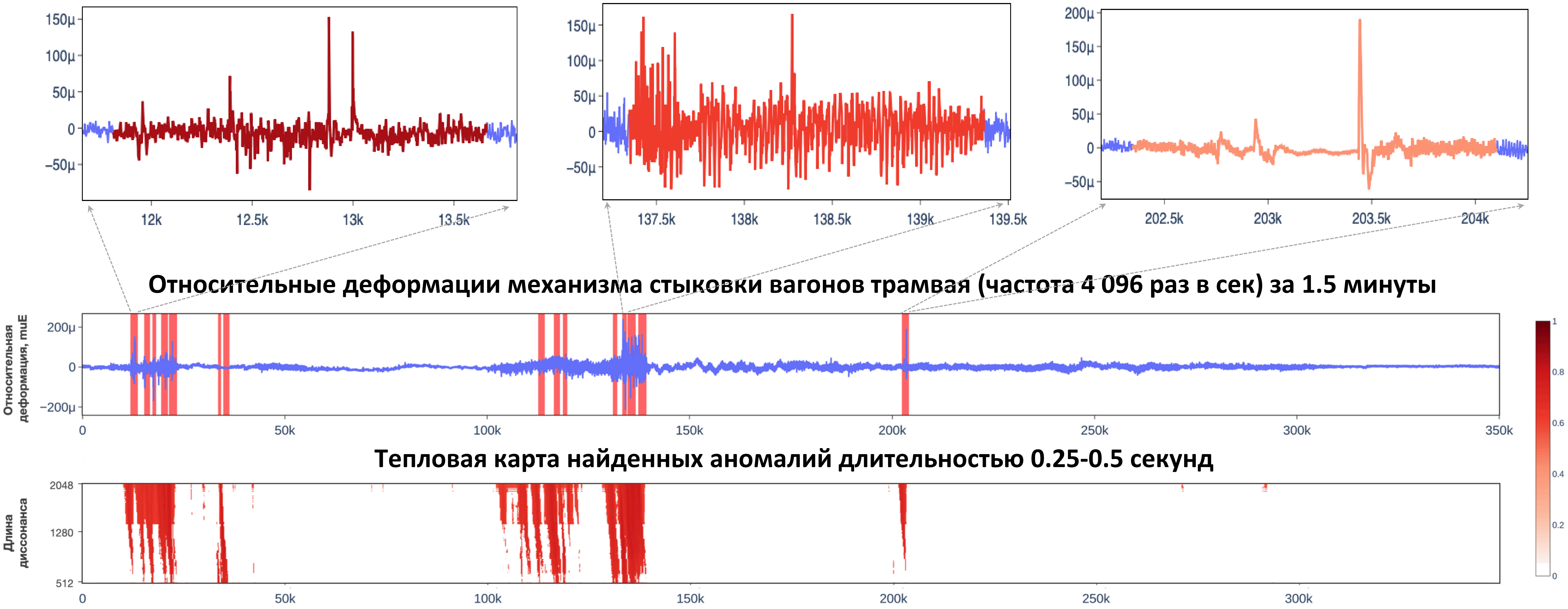
Тепловая карта найденных аномалий длительностью 0.5-2 суток



Выявление аномалий в машиностроении

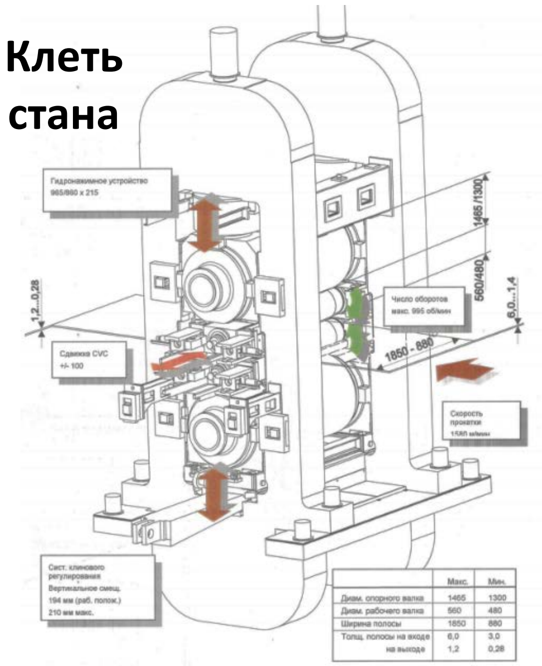


Примеры найденных аномалий длительностью 0.25-0.5 секунд

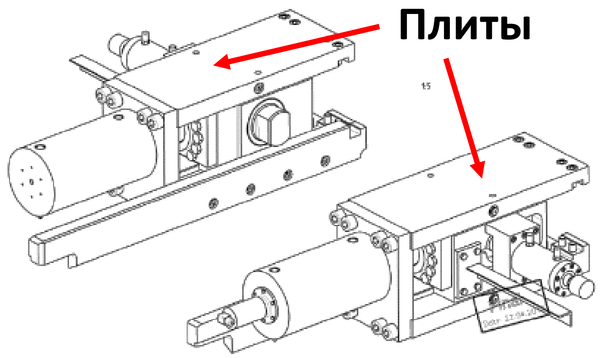


Поиск аномалий во временных рядах из цифровой индустрии

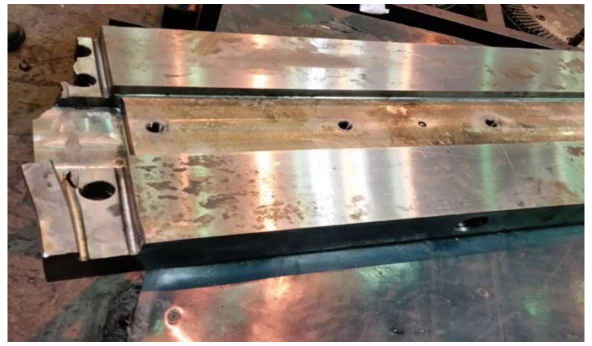
Клеть стана



Плиты

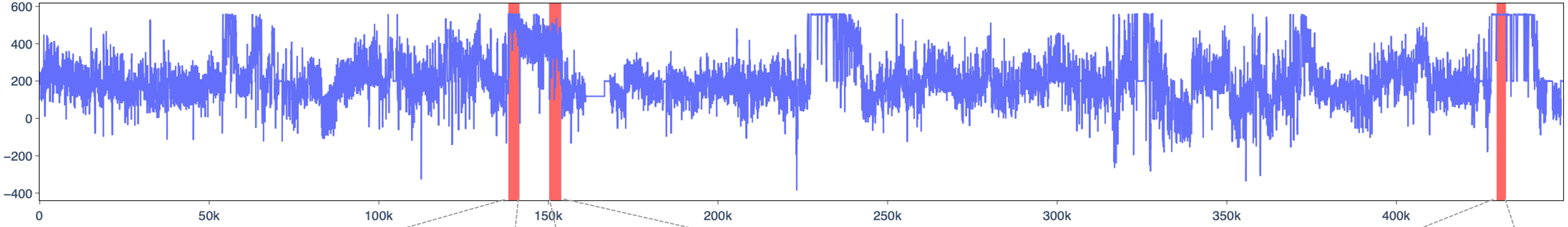


Система валков с непрерывно изменяемой кривизной

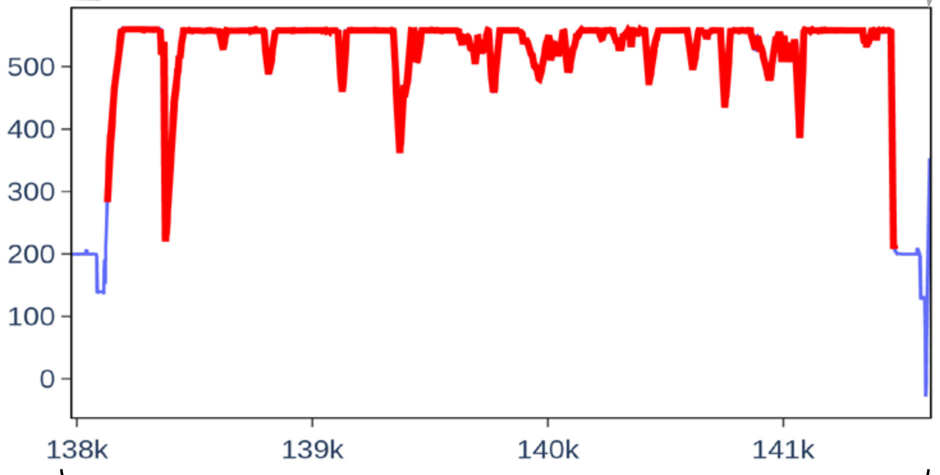


Разрушение плит

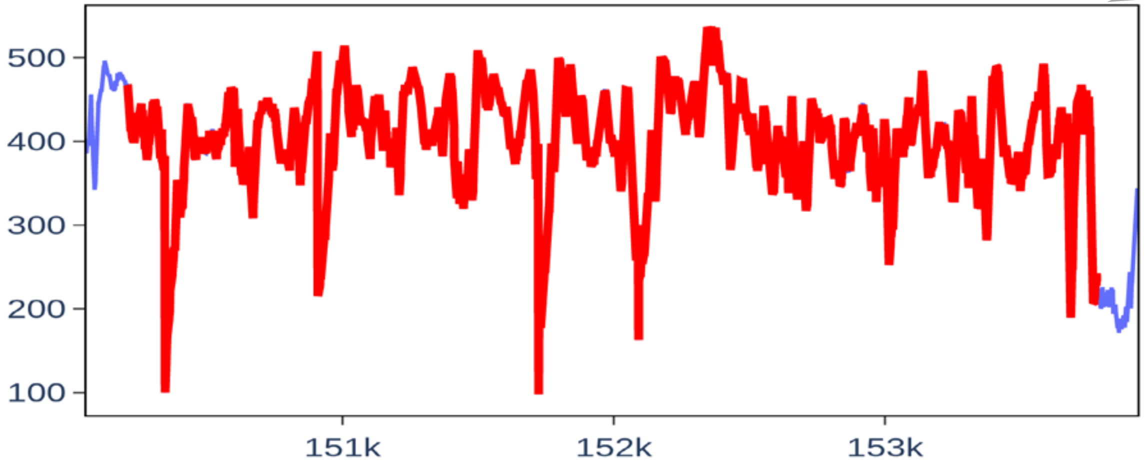
Фактическое изгибающее усилие прокатки



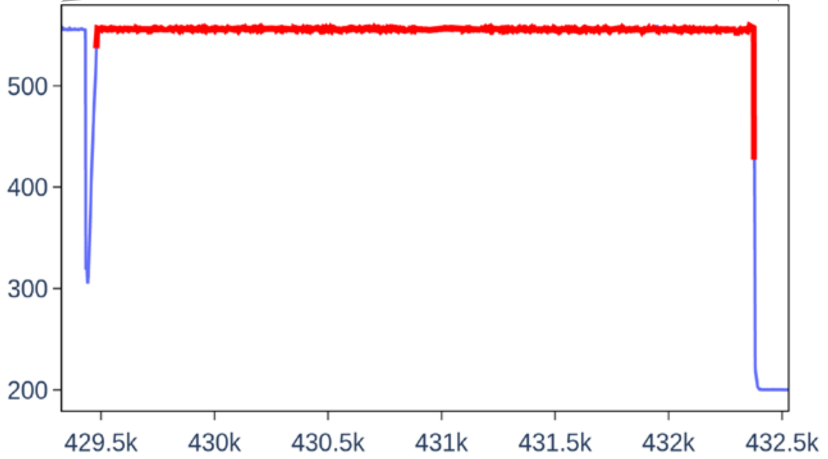
Аномальные напряжения плит



55 минут

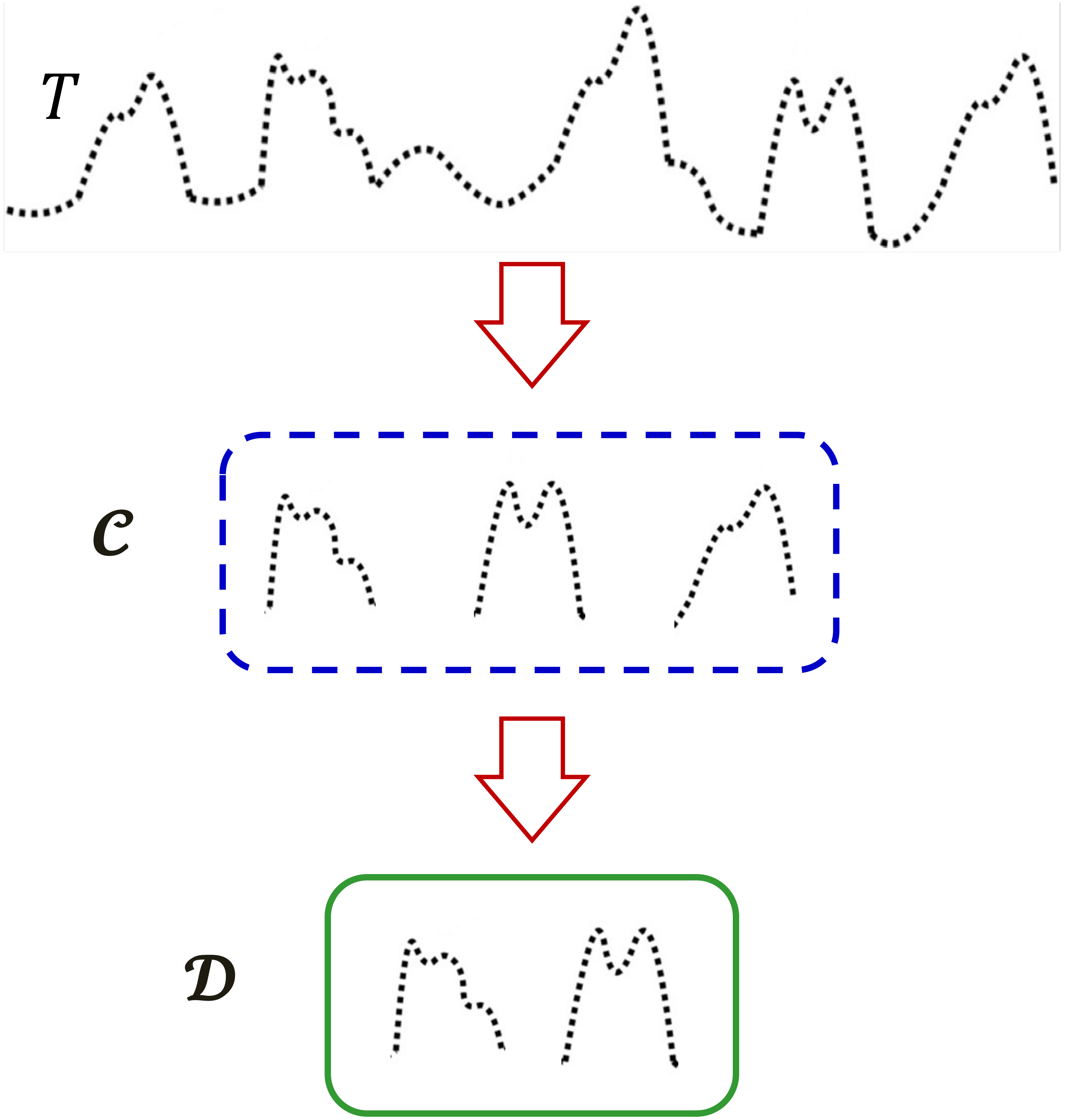


1 час



48 минут

PD3: Parallel DRAG-based Discord Discovery



1. **Отбор**

   За одно сканирование ряда сформировать **множество кандидатов** в диссонансы

2. **Очистка**

   За одно сканирование ряда **отбросить ложноположительных кандидатов**

Отбор кандидатов

<u>пока</u> не конец ряда $\boldsymbol{T}$:

текущая подпоследовательность $\boldsymbol{s}$

Кандидат := TRUE

<u>для всех</u> $\boldsymbol{c_i} \in \boldsymbol{\mathcal{C}}$ <u>и</u> $\boldsymbol{s} \cap \boldsymbol{c_i} = \emptyset$

<u>если</u> $\text{dist}(\boldsymbol{s}, \boldsymbol{c_i}) < r$ <u>то</u>

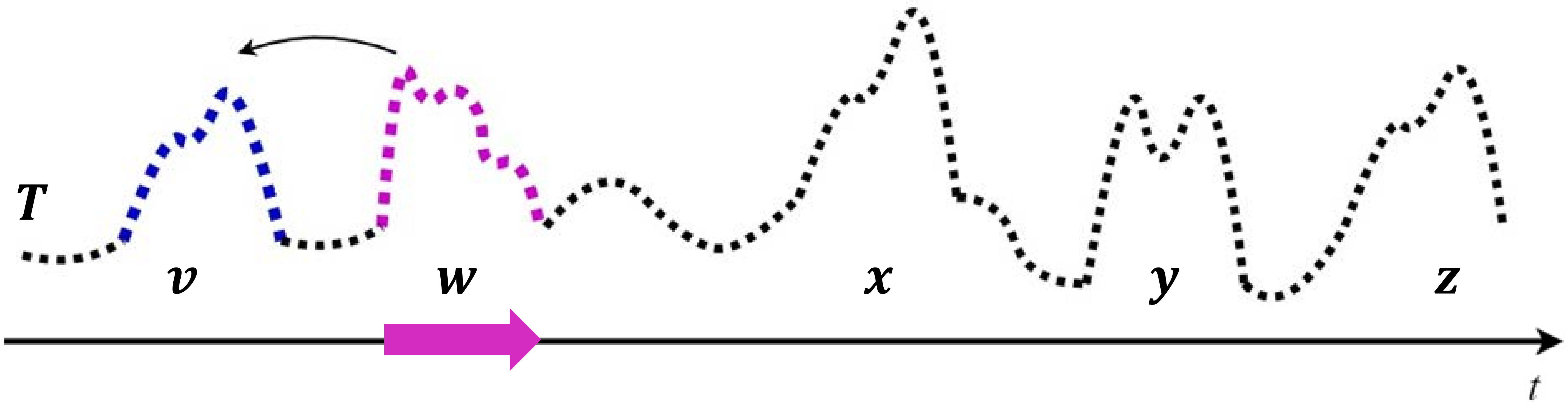
$\boldsymbol{\mathcal{C}} \coloneqq \boldsymbol{\mathcal{C}} \setminus \boldsymbol{c_i}$; Кандидат := FALSE

<u>если</u> Кандидат = TRUE <u>то</u> $\boldsymbol{\mathcal{C}} \coloneqq \boldsymbol{\mathcal{C}} \cup \boldsymbol{s}$

$\boldsymbol{\mathcal{C}} = \{\boldsymbol{v}\}$

$\text{dist}(\boldsymbol{w}, \boldsymbol{v}) \geq r$

$\boldsymbol{\mathcal{C}} = \{\boldsymbol{v}, \boldsymbol{w}\}$



Отбор кандидатов

<u>пока</u> не конец ряда $\boldsymbol{T}$:

текущая подпоследовательность $\boldsymbol{s}$

Кандидат := TRUE

<u>для всех</u> $\boldsymbol{c_i} \in \boldsymbol{\mathcal{C}}$ <u>и</u> $\boldsymbol{s} \cap \boldsymbol{c_i} = \emptyset$

<u>если</u> $\mathrm{dist}(\boldsymbol{s}, \boldsymbol{c_i}) < r$ <u>то</u>

$\boldsymbol{\mathcal{C}} \coloneqq \boldsymbol{\mathcal{C}} \setminus \boldsymbol{c_i}$; Кандидат := FALSE

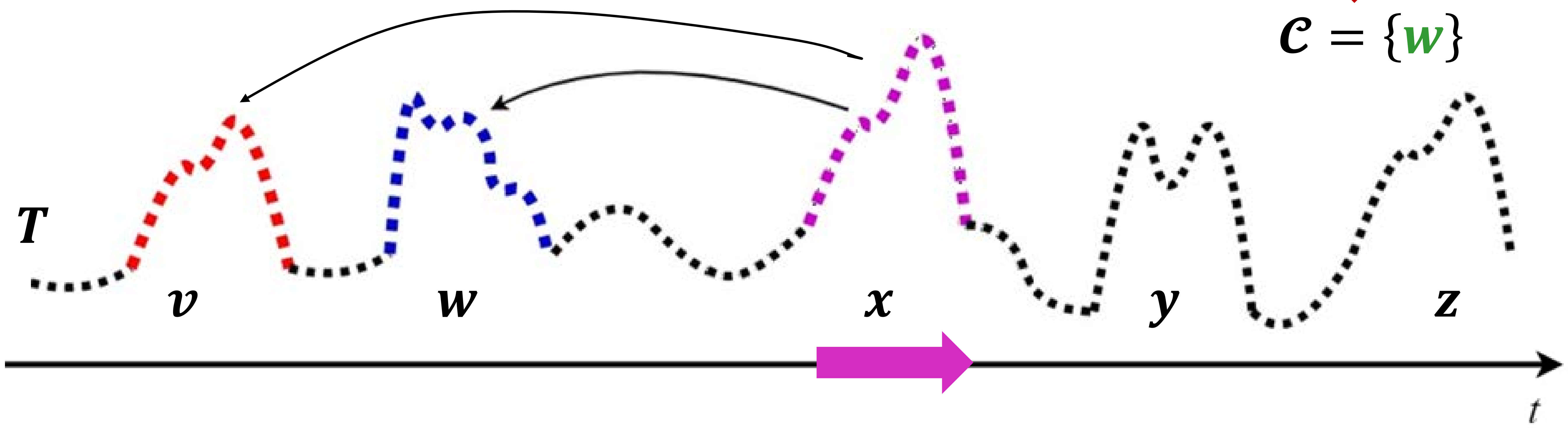
<u>если</u> Кандидат = TRUE <u>то</u> $\boldsymbol{\mathcal{C}} \coloneqq \boldsymbol{\mathcal{C}} \cup \boldsymbol{s}$

$\boldsymbol{\mathcal{C}} = \{\boldsymbol{v}, \boldsymbol{w}\}$

$\mathrm{dist}(\boldsymbol{x}, \boldsymbol{v}) < r$

$\mathrm{dist}(\boldsymbol{x}, \boldsymbol{w}) \geq r$

$\boldsymbol{\mathcal{C}} = \{\boldsymbol{w}\}$



Отбор кандидатов

<u>пока</u> не конец ряда $\boldsymbol{T}$:

    текущая подпоследовательность $\boldsymbol{s}$

    Кандидат := TRUE

    <u>для всех</u> $\boldsymbol{c_i} \in \boldsymbol{\mathcal{C}}$ <u>и</u> $\boldsymbol{s} \cap \boldsymbol{c_i} = \emptyset$

        <u>если</u> $\mathrm{dist}(\boldsymbol{s}, \boldsymbol{c_i}) < r$ <u>то</u>

            $\boldsymbol{\mathcal{C}} \coloneqq \boldsymbol{\mathcal{C}} \setminus \boldsymbol{c_i}$; Кандидат := FALSE

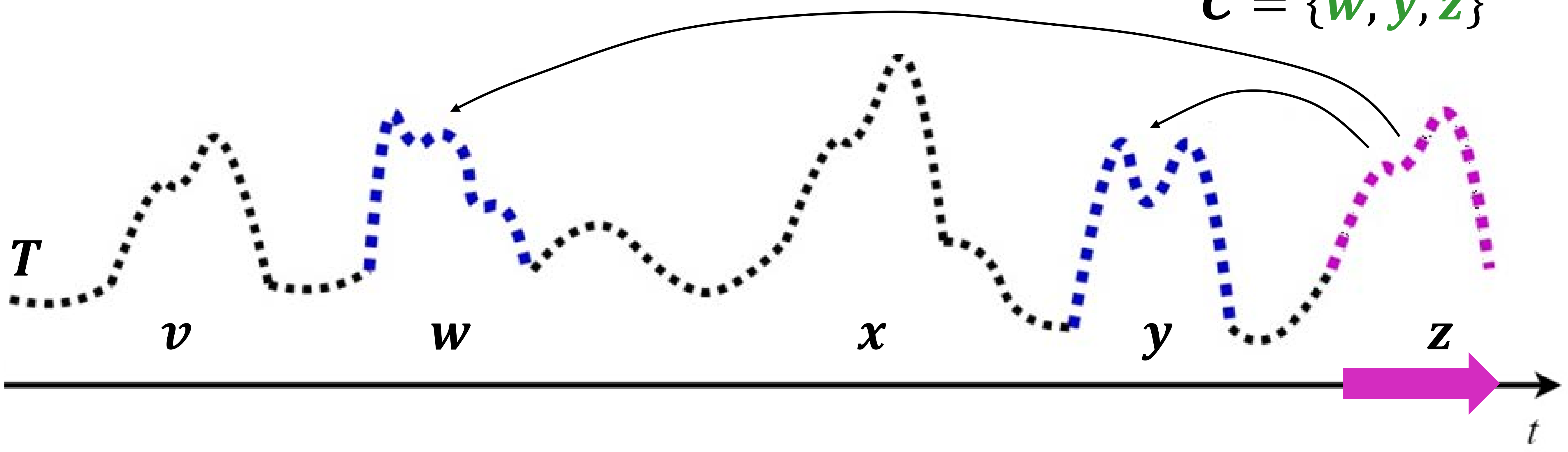
    <u>если</u> Кандидат = TRUE <u>то</u> $\boldsymbol{\mathcal{C}} \coloneqq \boldsymbol{\mathcal{C}} \cup \boldsymbol{s}$

$\boldsymbol{\mathcal{C}} = \{\boldsymbol{w}, \boldsymbol{y}\}$

$\mathrm{dist}(\boldsymbol{z}, \boldsymbol{w}) \geq r$

$\mathrm{dist}(\boldsymbol{z}, \boldsymbol{y}) \geq r$

$\boldsymbol{\mathcal{C}} = \{\boldsymbol{w}, \boldsymbol{y}, \boldsymbol{z}\}$



Очистка кандидатов

$\mathcal{D} := \mathcal{C}$

<u>пока</u> не конец ряда $\boldsymbol{T}$:

  текущая подпоследовательность $\boldsymbol{s}$

  <u>для всех</u> $\boldsymbol{d_i} \in \mathcal{D}$ <u>и</u> $\boldsymbol{s} \cap \boldsymbol{d_i} = \emptyset$

    <u>если</u> $\text{dist}(\boldsymbol{s}, \boldsymbol{d_i}) < r$ <u>то</u>

      $\mathcal{D} := \mathcal{D} \setminus \boldsymbol{d_i}$

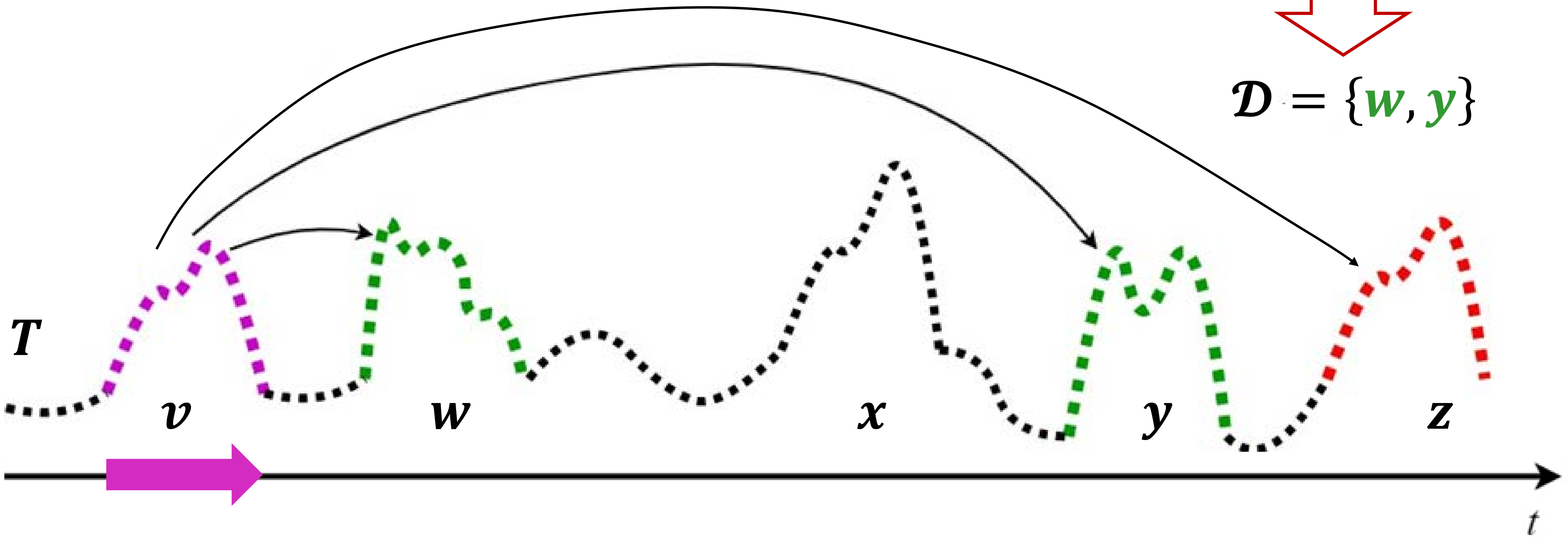
$\mathcal{D} = \{\boldsymbol{w}, \boldsymbol{y}, z\}$

$\text{dist}(\boldsymbol{v}, w) \geq r$

$\text{dist}(\boldsymbol{v}, \boldsymbol{y}) \geq r$

$\text{dist}(\boldsymbol{v}, \boldsymbol{z}) < r$

$\mathcal{D} = \{\boldsymbol{w}, \boldsymbol{y}\}$



Очистка кандидатов

$\mathcal{D} := \mathcal{C}$

<u>пока</u> не конец ряда $\boldsymbol{T}$:

  текущая подпоследовательность $\boldsymbol{s}$

  <u>для всех</u> $\boldsymbol{d_i} \in \mathcal{D}$ <u>и</u> $\boldsymbol{s} \cap \boldsymbol{d_i} = \emptyset$

    <u>если</u> $\mathrm{dist}(\boldsymbol{s}, \boldsymbol{d_i}) < r$ <u>то</u>

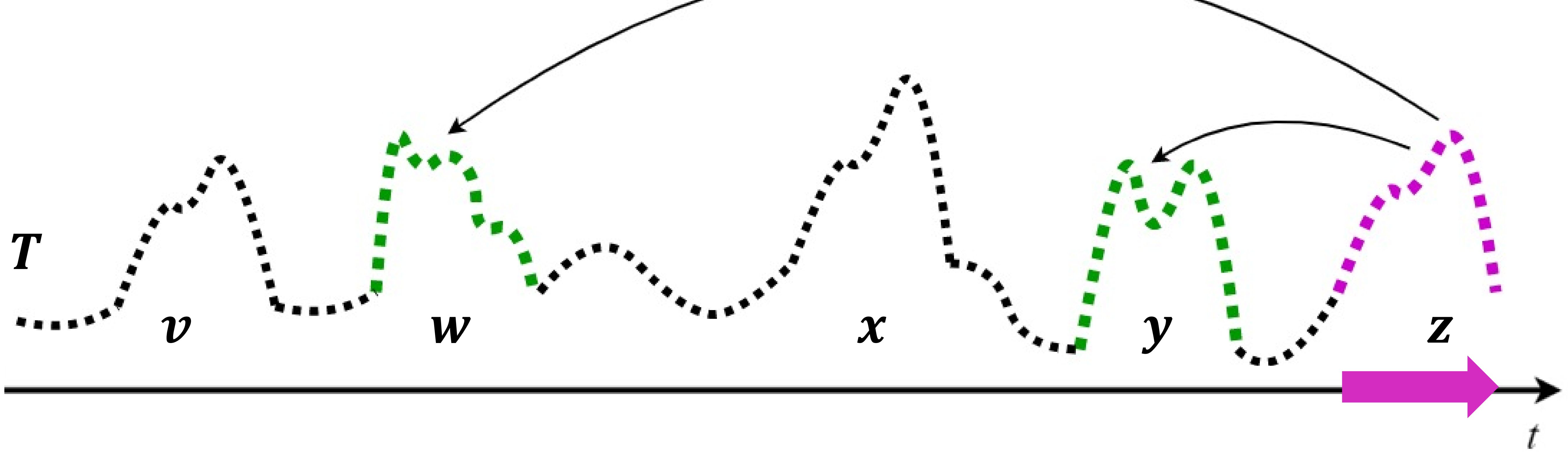
      $\mathcal{D} := \mathcal{D} \setminus \boldsymbol{d_i}$

$\mathcal{D} = \{\boldsymbol{w}, \boldsymbol{y}\}$

$\mathrm{dist}(\boldsymbol{z}, \boldsymbol{w}) \geq r$

$\mathrm{dist}(\boldsymbol{z}, \boldsymbol{y}) \geq r$

$\mathcal{D} = \{\boldsymbol{w}, \boldsymbol{y}\}$



PALMAD: автоматизированный подбор порога

Algorithm 3 PALMAD (IN T, $minL$, $maxL$, $topK$; OUT $\mathcal{D}$)

$\mathcal{D} \leftarrow \varnothing$; $r \leftarrow 2\sqrt{minL}$; $nnDist_{minL} \leftarrow -\infty$
$\{\bar{\mu}, \bar{\sigma}\} \leftarrow$ CalcMeanStd$(T, minL)$
while $nnDist_{minL} < 0$ **and** $|D_{minL}| < topK$ **do**
  $D_{minL} \leftarrow \mathrm{PD3}(T, \bar{\mu}, \bar{\sigma}, minL, r^2)$; $\mathcal{D} \leftarrow \mathcal{D} \cup D_{minL}$; $nnDist_{minL} \leftarrow \min\limits_{d \in D_{minL}} d.nnDist$
  $r \leftarrow 0.5 \cdot r$
for $i \leftarrow minL + 1$ **to** $minL + 4$ **do**
  $nnDist_i \leftarrow -\infty$
  $\{\bar{\mu}, \bar{\sigma}\} \leftarrow$ UpdateMeanStd$(T, \bar{\mu}, \bar{\sigma}, i)$
  while $nnDist_i < 0$ **and** $|D_i| < topK$ **do**
    $r \leftarrow 0.99 \cdot nnDist_{i-1}$
    $D_i \leftarrow \mathrm{PD3}(T, \bar{\mu}, \bar{\sigma}, i, r^2)$; $\mathcal{D} \leftarrow \mathcal{D} \cup D_i$; $nnDist_i \leftarrow \min\limits_{d \in D_i} d.nnDist$
    $r \leftarrow 0.99 \cdot r$
for $i \leftarrow minL + 5$ **to** $maxL$ **do**
  $\mu \leftarrow \mathrm{Mean}(\{nnDist_k\}_{k=i-1}^{i-5})$; $\sigma \leftarrow \mathrm{Std}(\{nnDist_k\}_{k=i-1}^{i-5})$; $r \leftarrow \mu - 2\sigma$
  $\{\bar{\mu}, \bar{\sigma}\} \leftarrow$ UpdateMeanStd$(T, \bar{\mu}, \bar{\sigma}, i)$
  $D_i \leftarrow \mathrm{PD3}(T, \bar{\mu}, \bar{\sigma}, i, r^2)$; $\mathcal{D} \leftarrow \mathcal{D} \cup D_i$; $nnDist_i \leftarrow \min\limits_{d \in D_i} d.nnDist$
  while $nnDist_i < 0$ **and** $|D_i| < topK$ **do**
    $D_i \leftarrow \mathrm{PD3}(T, \bar{\mu}, \bar{\sigma}, i, r^2)$; $\mathcal{D} \leftarrow \mathcal{D} \cup D_i$; $nnDist_i \leftarrow \min\limits_{d \in D_i} d.nnDist$
    $r \leftarrow r - \sigma$
return $\mathcal{D}$

Шаг 1. Поиск диссонансов минимальной длины $minL$
$r = 2\sqrt{minL}$

Шаг 2. Поиск диссонансов следующих четырех длин
$r = 0.99 \cdot nndist_{m-1}$

Шаг 3. Поиск диссонансов всех оставшихся длин
$r = \mu - 2\sigma$

PALMAD

Algorithm 3 PALMAD (IN T, $minL$, $maxL$, $topK$; OUT $\mathcal{D}$)

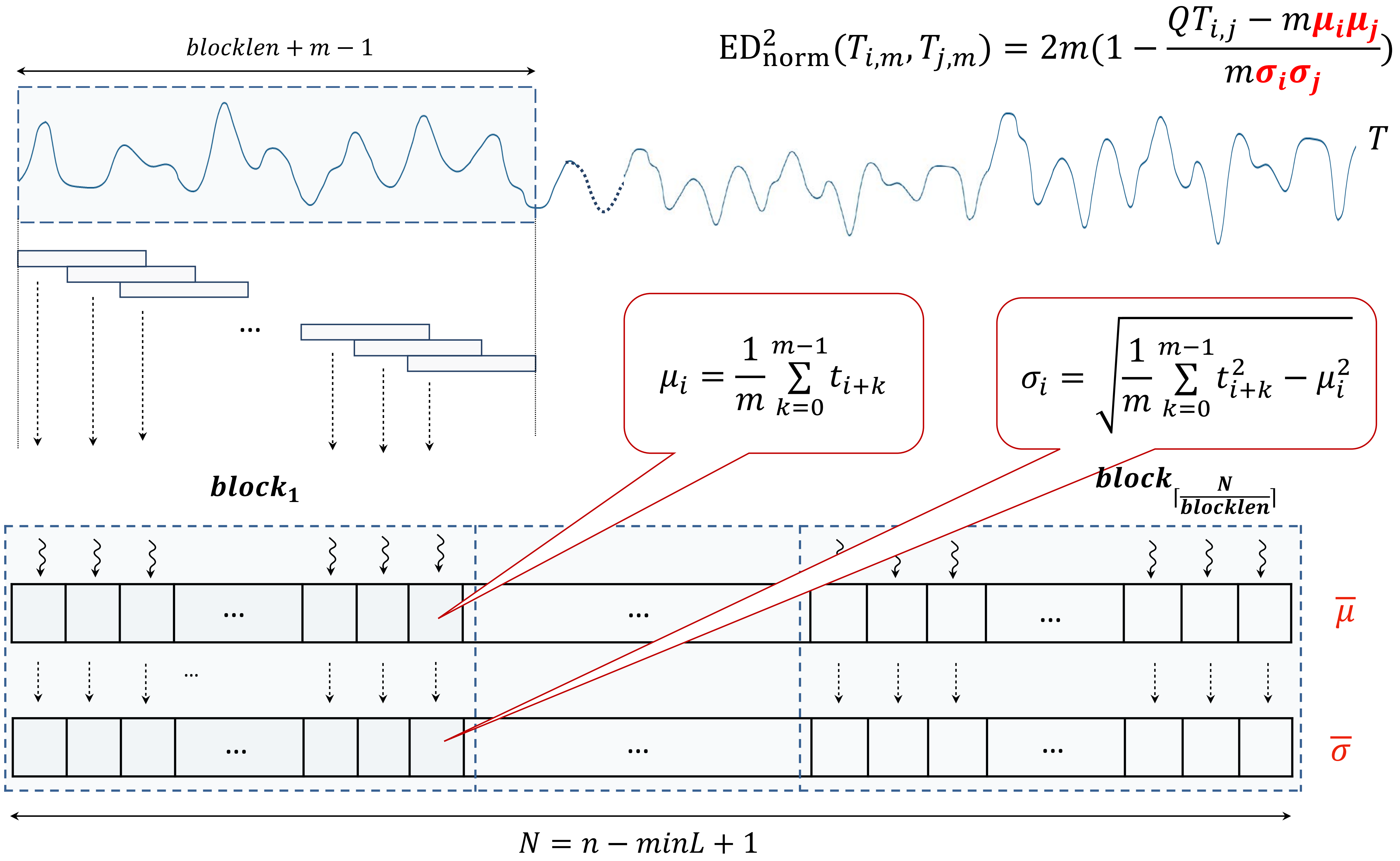
$\mathcal{D} \leftarrow \varnothing$; $r \leftarrow 2\sqrt{minL}$; $nnDist_{minL} \leftarrow -\infty$
$\{\bar{\mu}, \bar{\sigma}\} \leftarrow$ CalcMeanStd$(T, minL)$
while $nnDist_{minL} < 0$ **and** $|D_{minL}| < topK$ **do**
    $D_{minL} \leftarrow$ PD3$(T, \bar{\mu}, \bar{\sigma}, minL, r^2)$; $\mathcal{D} \leftarrow \mathcal{D} \cup D_{minL}$; $nnDist_{minL} \leftarrow \min\limits_{d \in D_{minL}} d.nnDist$
    $r \leftarrow 0.5 \cdot r$
for $i \leftarrow minL + 1$ **to** $minL + 4$ **do**
    $nnDist_i \leftarrow -\infty$
    $\{\bar{\mu}, \bar{\sigma}\} \leftarrow$ UpdateMeanStd$(T, \bar{\mu}, \bar{\sigma}, i)$
    while $nnDist_i < 0$ **and** $|D_i| < topK$ **do**
        $r \leftarrow 0.99 \cdot nnDist_{i-1}$
        $D_i \leftarrow$ PD3$(T, \bar{\mu}, \bar{\sigma}, i, r^2)$; $\mathcal{D} \leftarrow \mathcal{D} \cup D_i$; $nnDist_i \leftarrow \min\limits_{d \in D_i} d.nnDist$
        $r \leftarrow 0.99 \cdot r$
for $i \leftarrow minL + 5$ **to** $maxL$ **do**
    $\mu \leftarrow \mathrm{Mean}(\{nnDist_k\}_{k=i-1}^{i-5})$; $\sigma \leftarrow \mathrm{Std}(\{nnDist_k\}_{k=i-1}^{i-5})$; $r \leftarrow \mu - 2\sigma$
    $\{\bar{\mu}, \bar{\sigma}\} \leftarrow$ UpdateMeanStd$(T, \bar{\mu}, \bar{\sigma}, i)$
    $D_i \leftarrow$ PD3$(T, \bar{\mu}, \bar{\sigma}, i, r^2)$; $\mathcal{D} \leftarrow \mathcal{D} \cup D_i$; $nnDist_i \leftarrow \min\limits_{d \in D_i} d.nnDist$
    while $nnDist_i < 0$ **and** $|D_i| < topK$ **do**
        $D_i \leftarrow$ PD3$(T, \bar{\mu}, \bar{\sigma}, i, r^2)$; $\mathcal{D} \leftarrow \mathcal{D} \cup D_i$; $nnDist_i \leftarrow \min\limits_{d \in D_i} d.nnDist$
        $r \leftarrow r - \sigma$
return $\mathcal{D}$

– сокращение избыточных вычислений

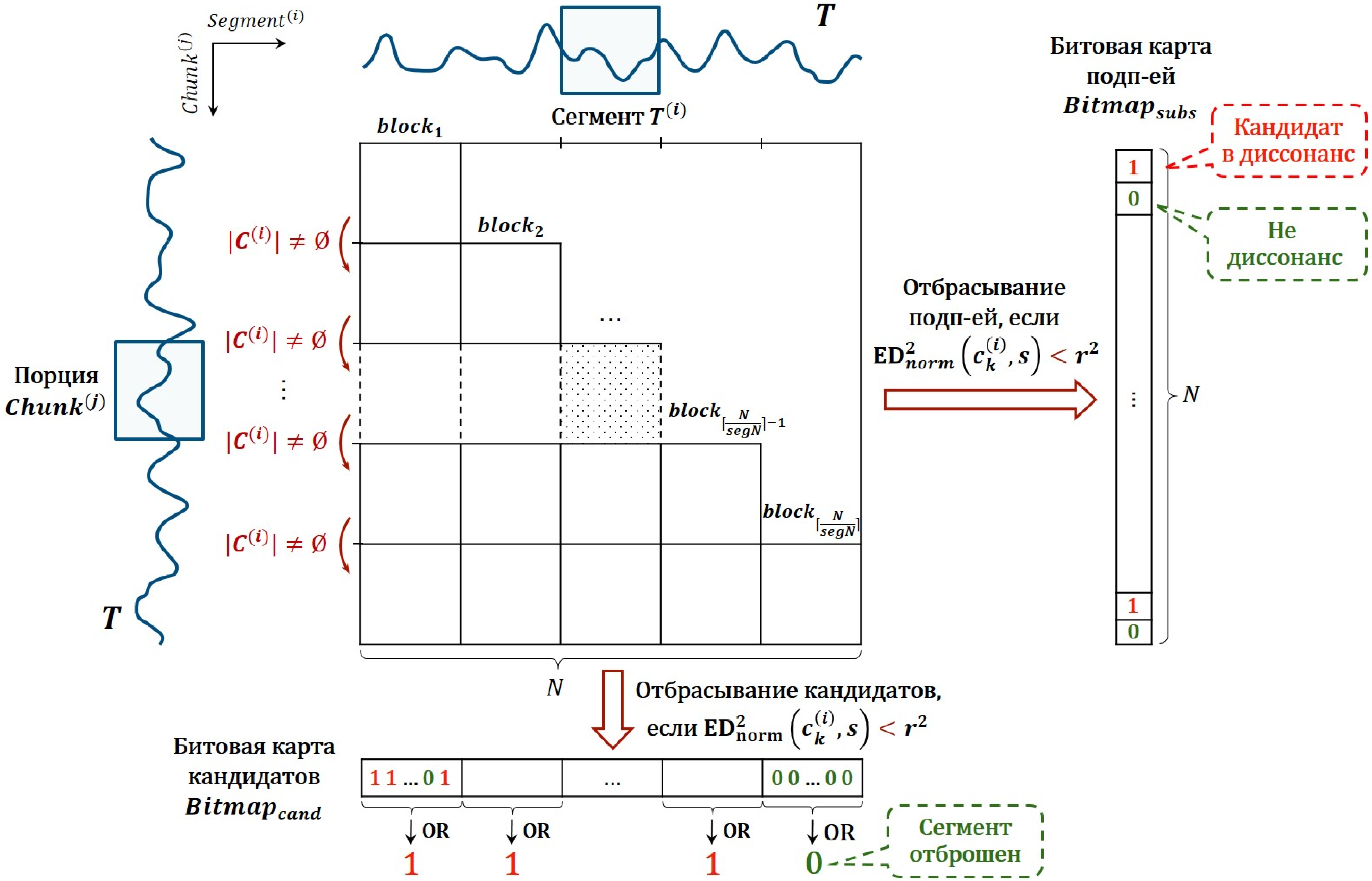
– использование PD3

– применение ED^2_{norm}

Предобработка: вычисление $\overline{\mu}$ и $\overline{\sigma}$

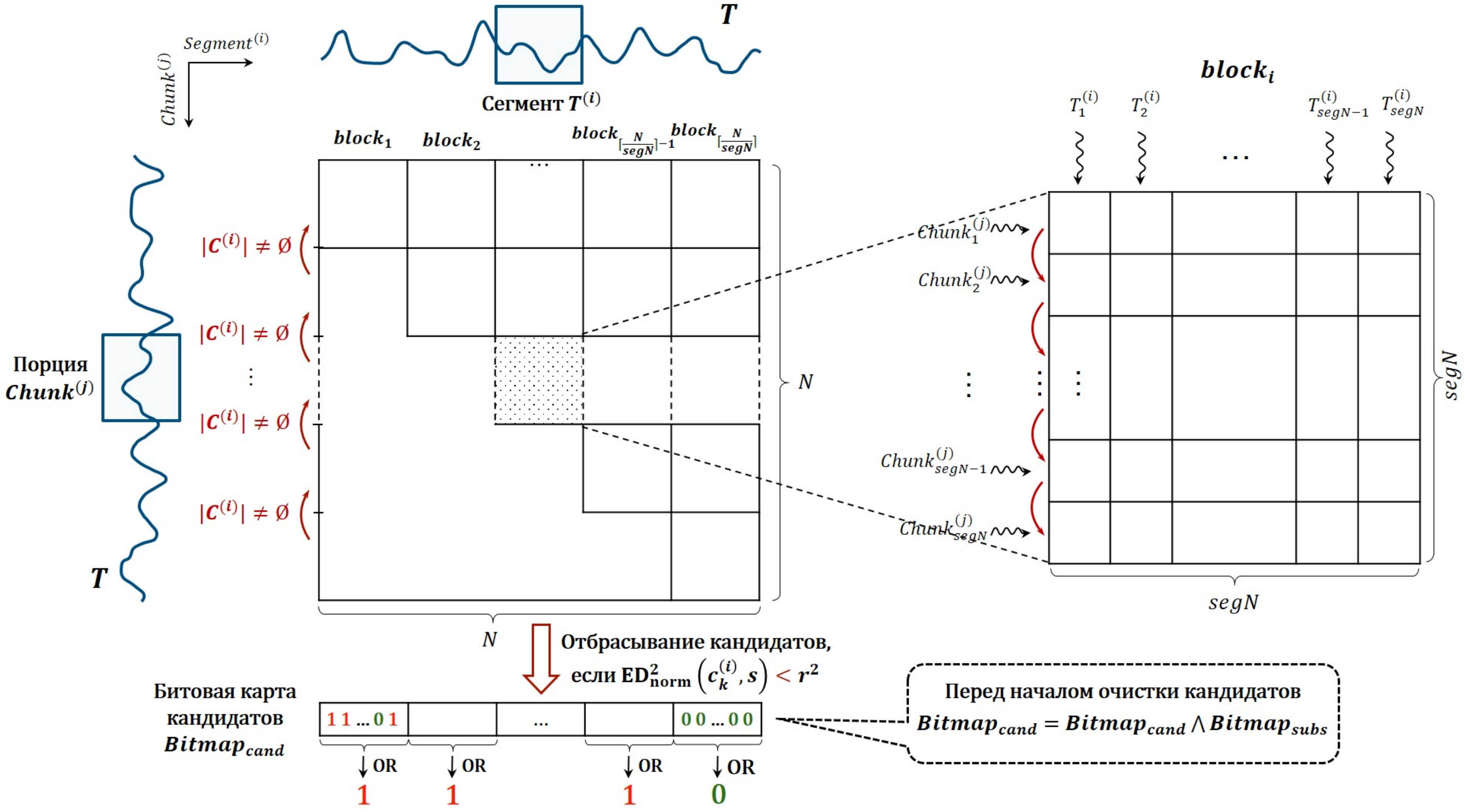


PD3: Отбор кандидатов, блочное распараллеливание



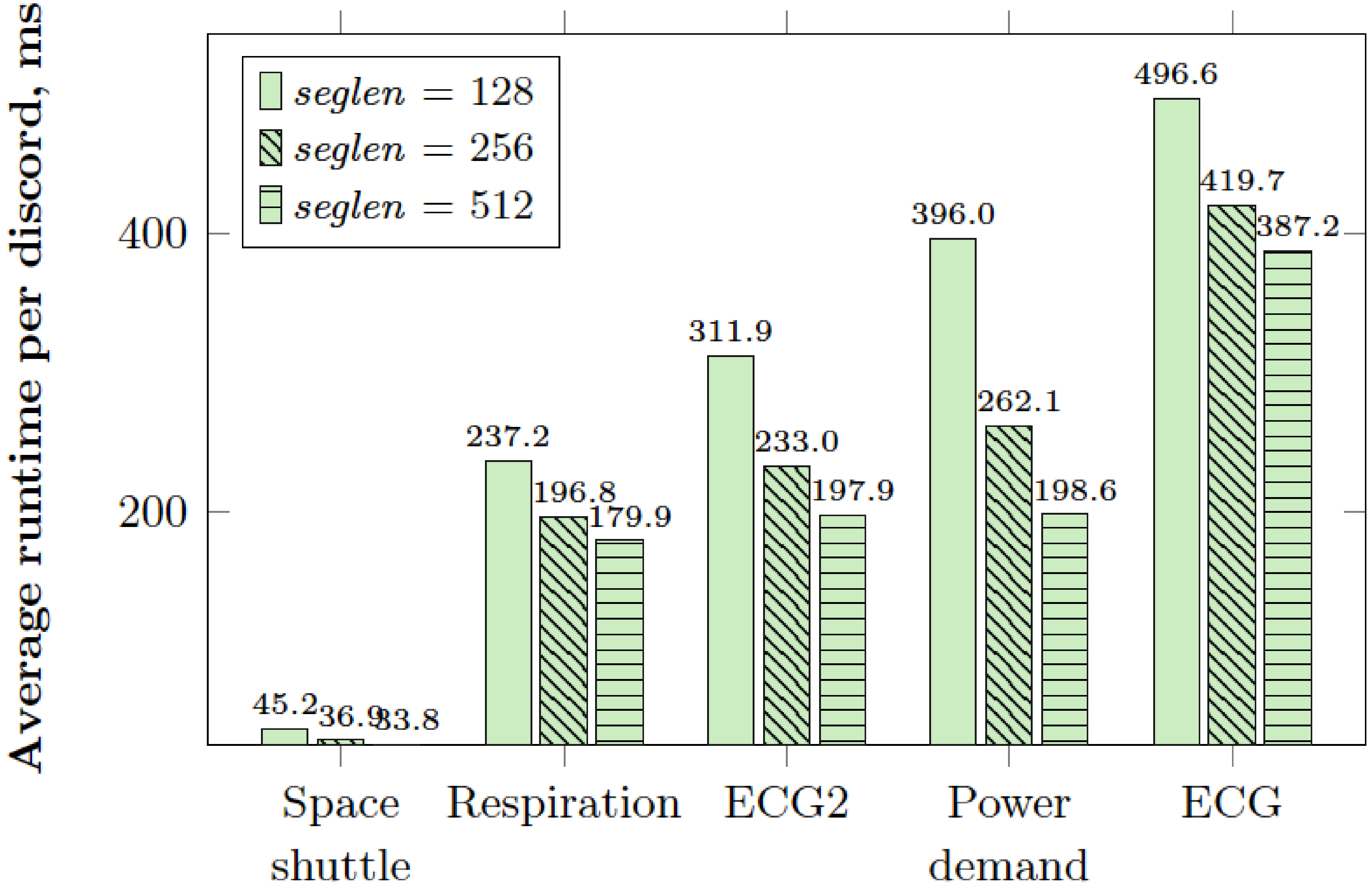
PD3: Очистка кандидатов

Блочное распараллеливание *Распараллеливание по нитям*

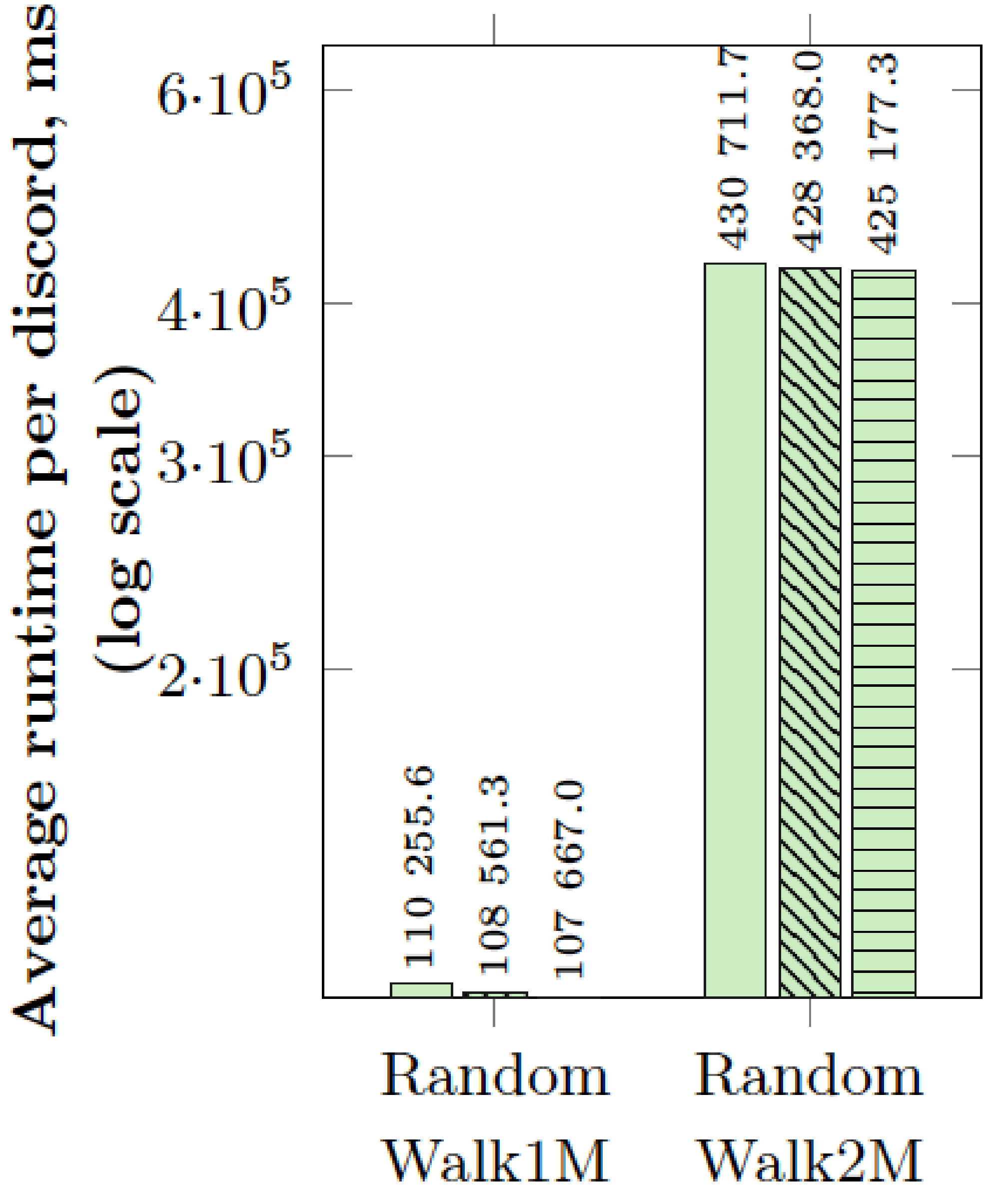


Производительность: влияние длины сегмента

Реальные временные ряды



Синтетические временные ряды



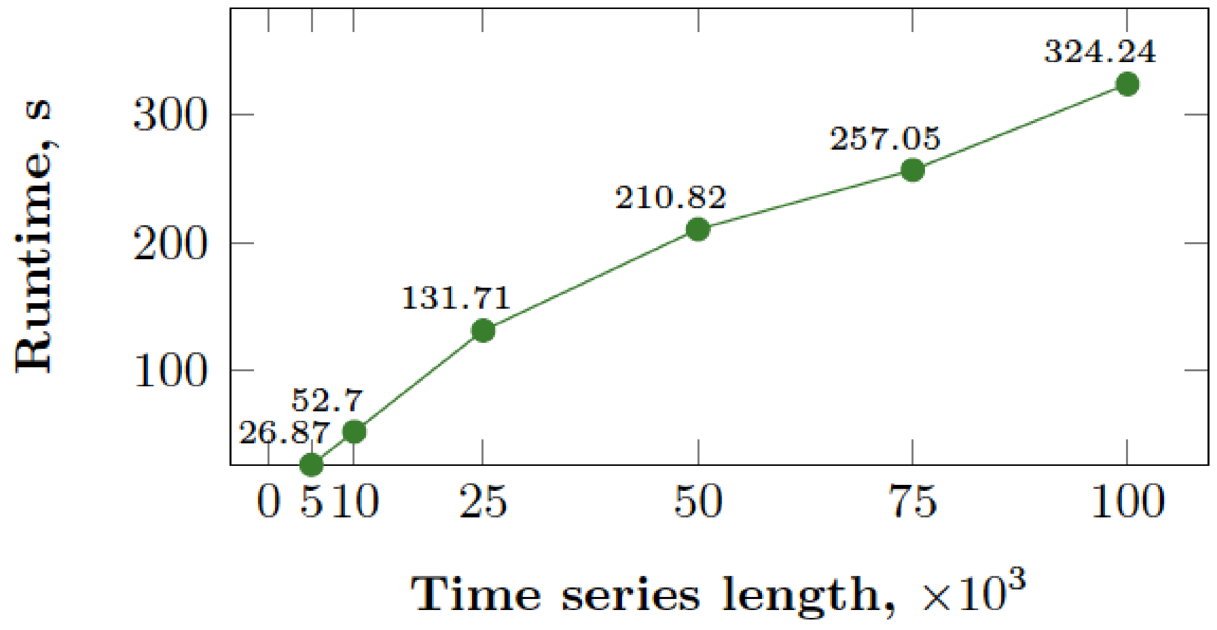
Производительность алгоритма пропорциональна длине сегмента

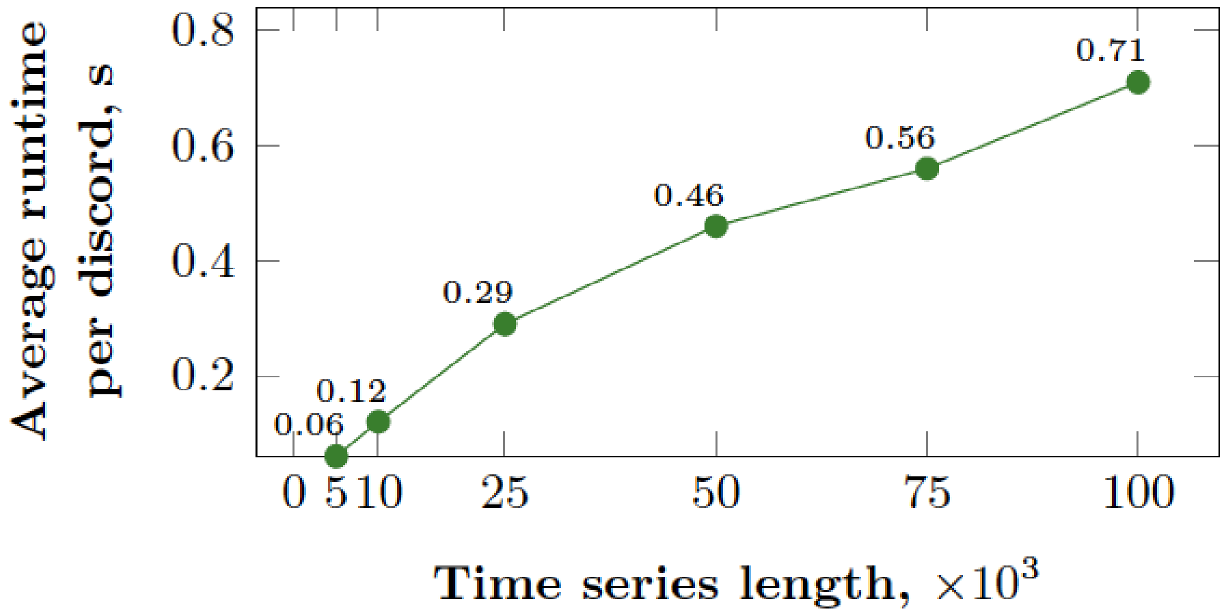
Масштабируемость: влияние длины ряда

Время на поиск **всех** диссонансов

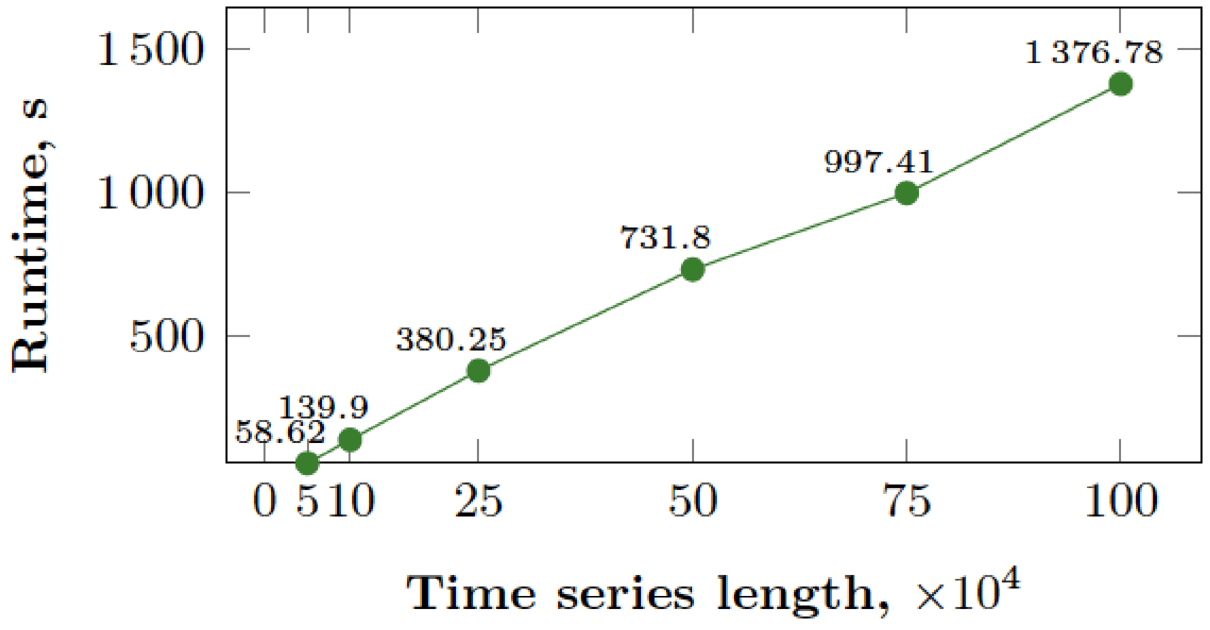
Среднее время на поиск **одного** диссонанса

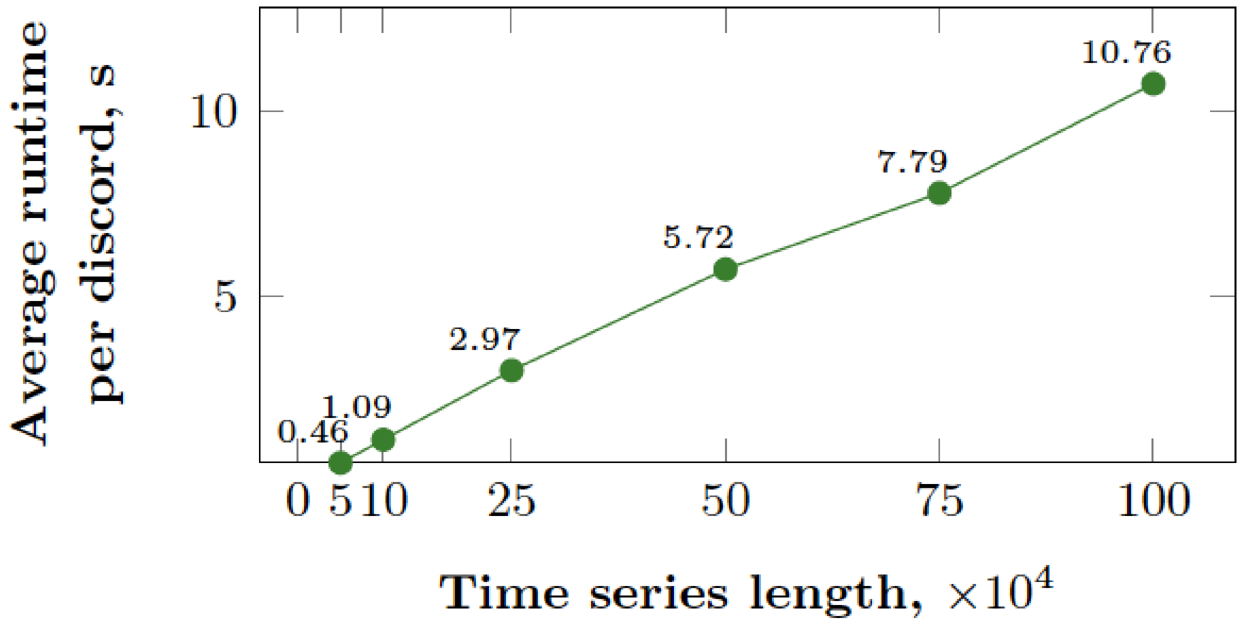
Ряд **Koski-ECG** $m \in [458, 916]$





Ряд **RandomWalk1M** $m \in [128, 256]$



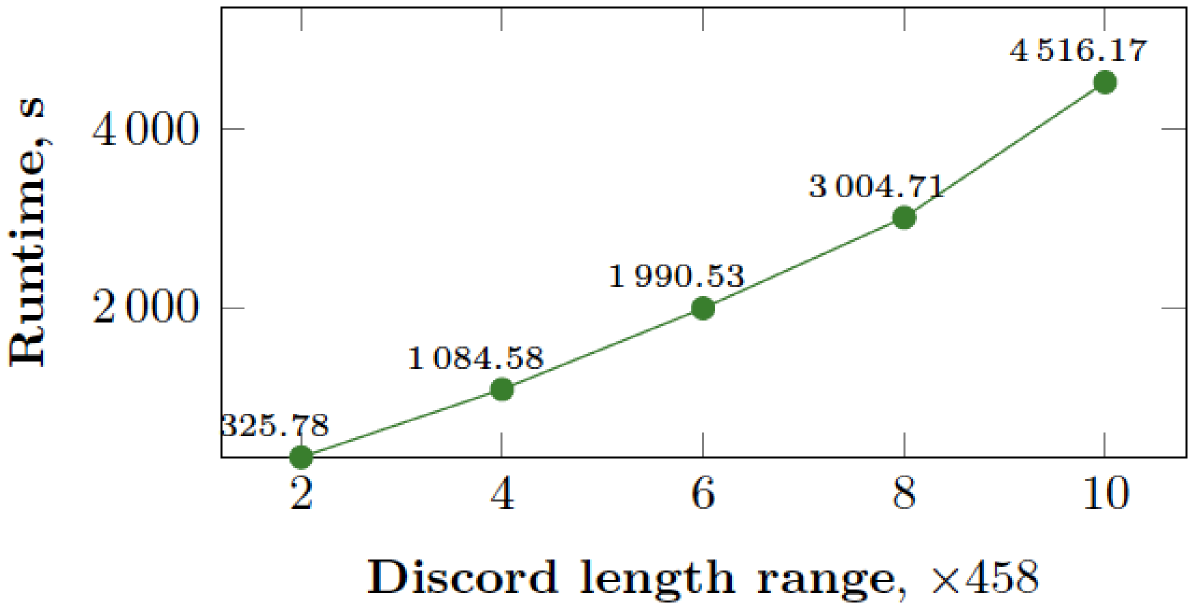


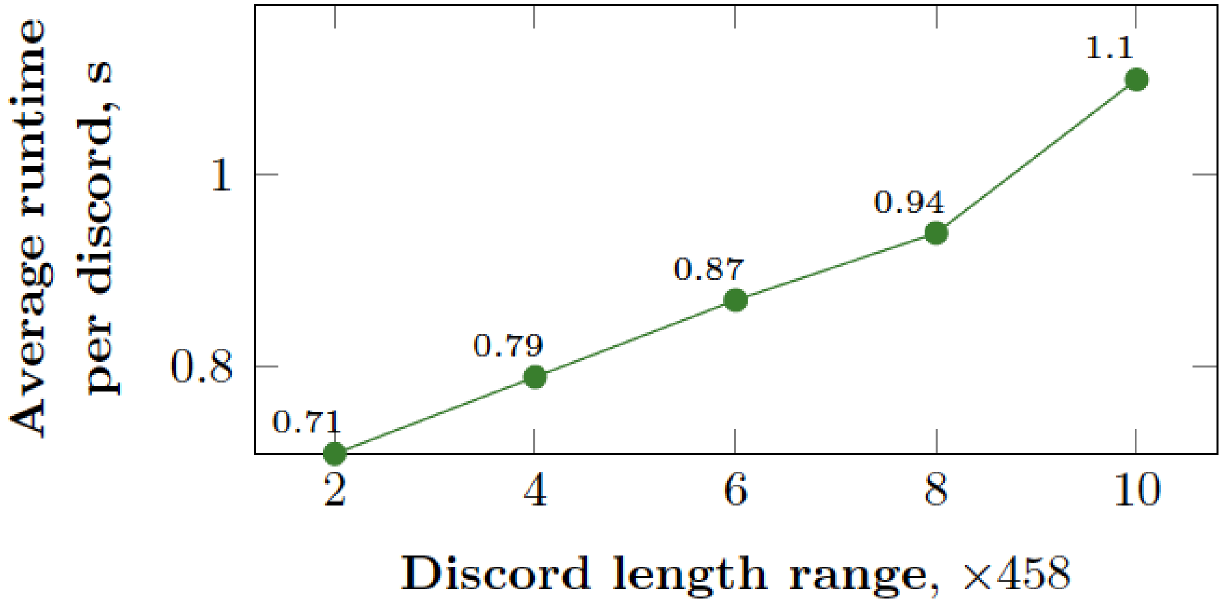
Масштабируемость: влияние диапазона длин диссонансов

Время на поиск **всех** диссонансов

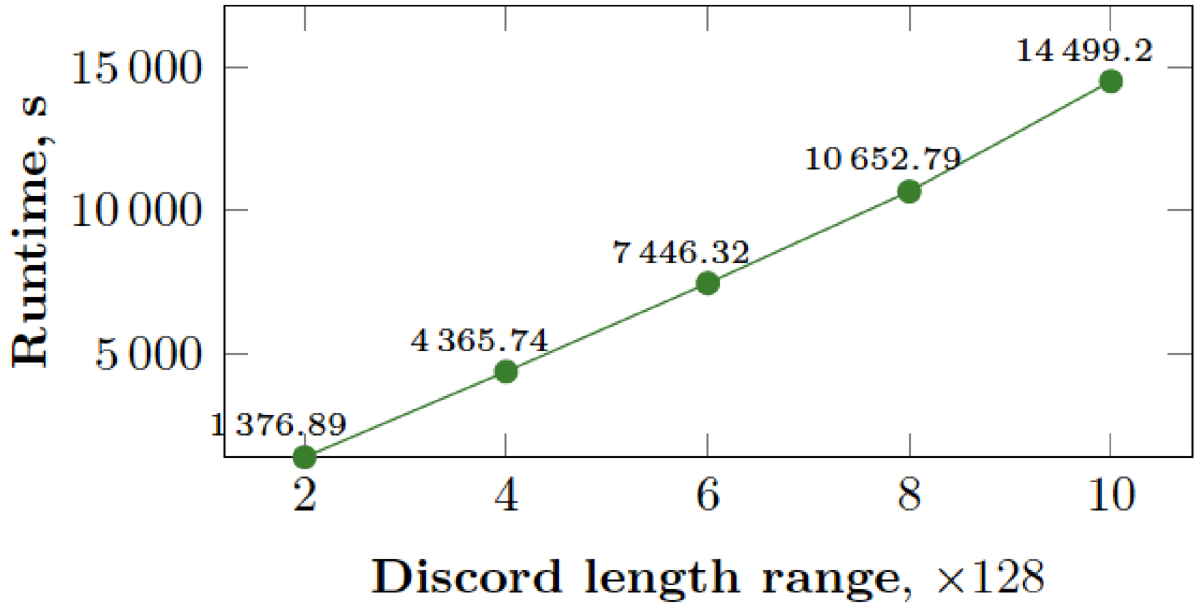
Среднее время на поиск **одного** диссонанса

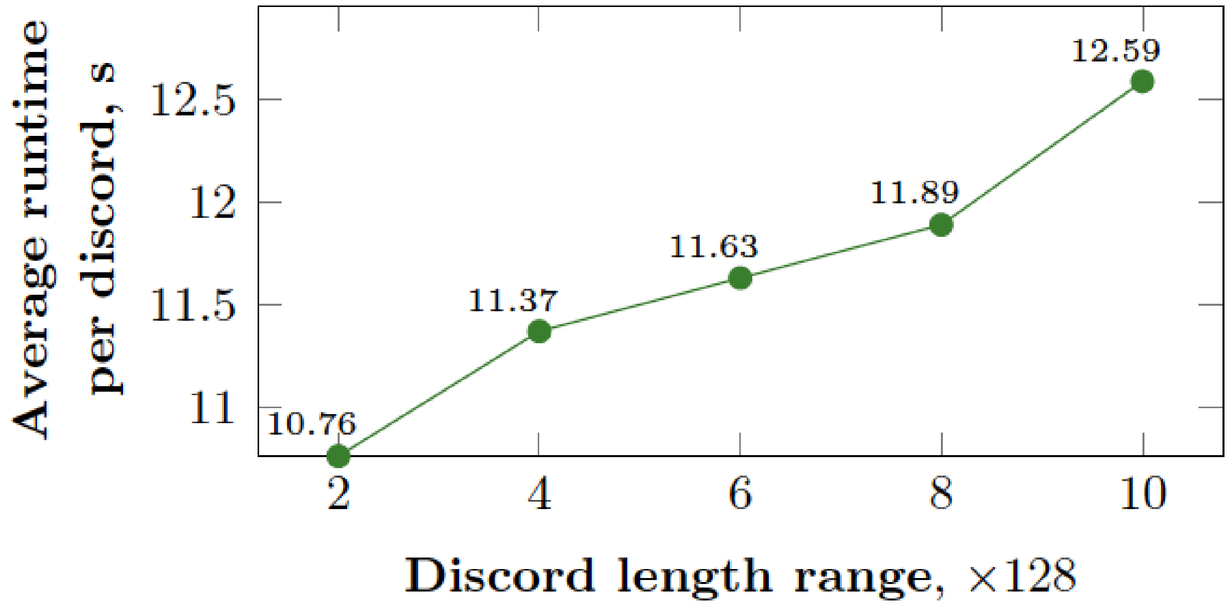
Ряд Koski-ECG





Ряд RandomWalk1M



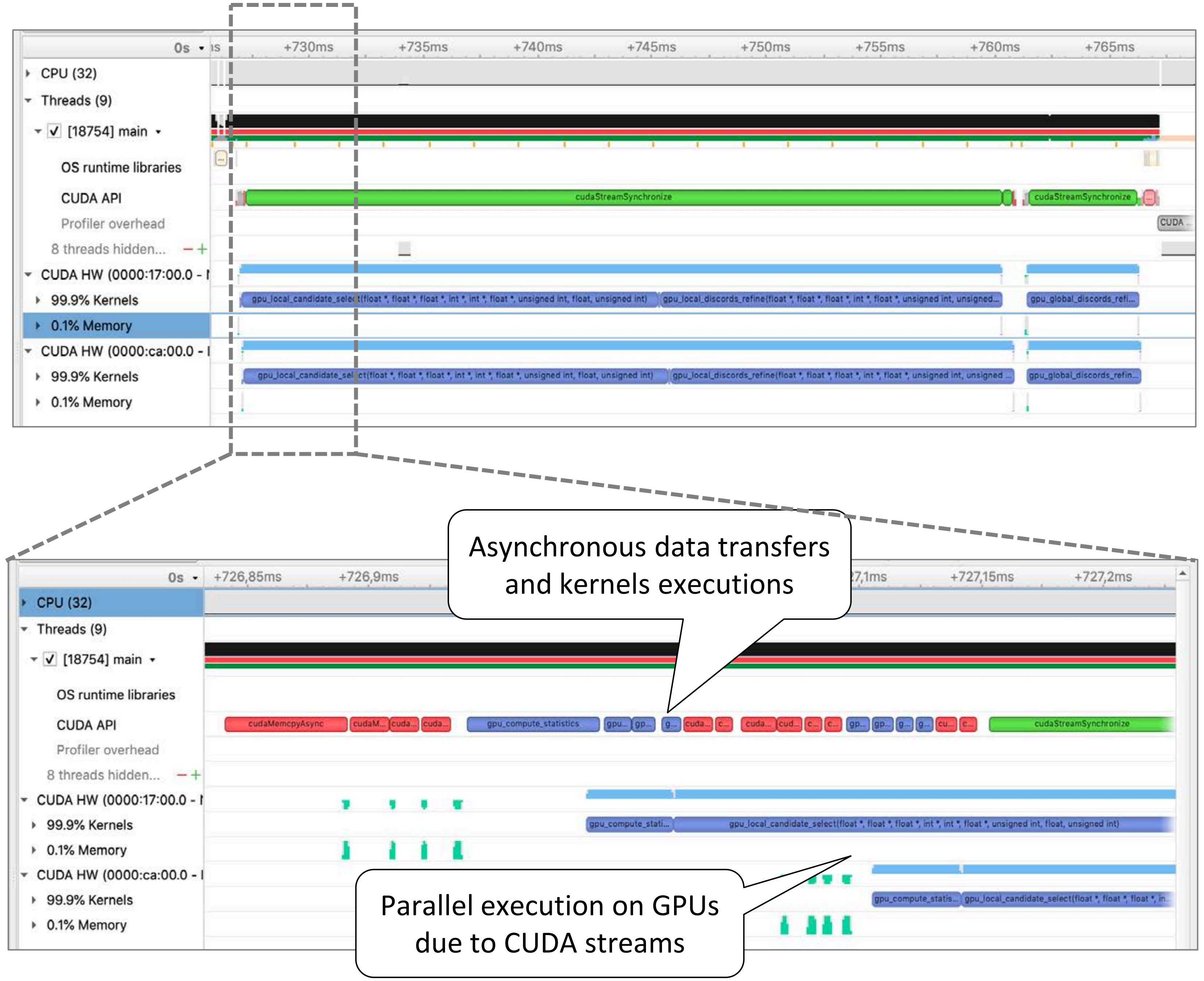


Efficient parallelization of PALMAD for a multi-GPU cluster node

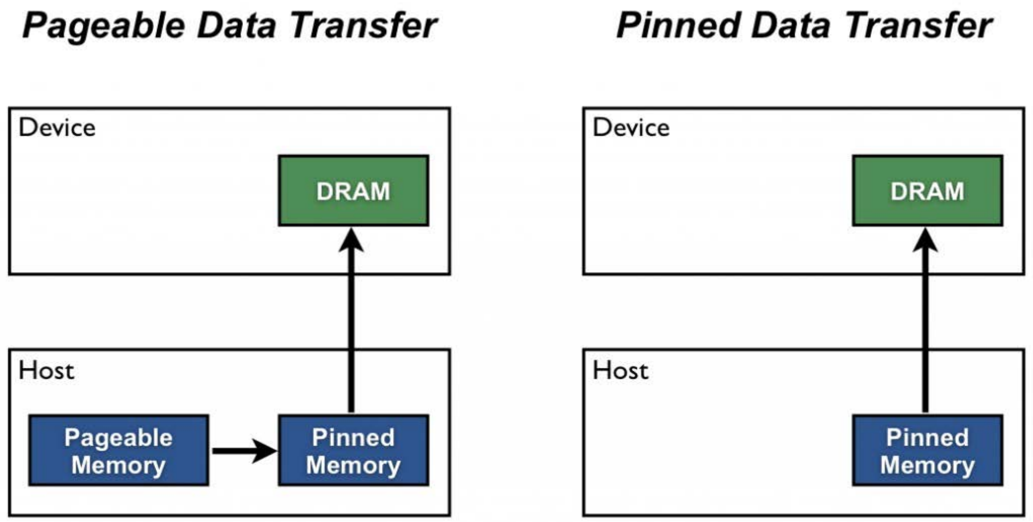
- Non-swappable host memory for data from GPU
- CUDA stream for each GPU

- Asynchronous CPU-GPU data transfers
- Asynchronous CUDA kernels

NVIDIA Nsight Systems




```
for (int i = 0; i < nGPUs; i++) {
    cudaSetDevice(i);
    cudaMalloc(…);
    cudaMallocHost(…);
    cudaStreamCreate(&streams[i]);
}
for (int i = 0; i < nGPUs; i++) {
    cudaSetDevice(i);
    cudaMemcpyAsync(…, cudaMemcpyHostToDevice, streams[i]);
    kernel<<<grid, block, 0, streams[i]>>>(…);
    cudaMemcpyAsync(…, cudaMemcpyDeviceToHost, streams[i])
}
cudaDeviceSynchronize();
```



Experiments: Setup

- **Hardware:** Neurocomputer SUSU, NVIDIA Ampere A100 PCIe (6 912 CUDA-cores, 9.7 TFLOPS)
- **Data:** MIT-BIH Arrhythmia Database[1)]
  - For generating the training dataset:

Time series	Training time series length, n_{train}	Test time series length, n_{test}	Snippet length, m	Subsequence length, l	Snippet number, K	Discord fraction, α
ECG_803_805	$5 \cdot 10^4$	$5 \cdot 10^4$	350	75	2	0.0002
ECG_803_806	$5 \cdot 10^4$	$5 \cdot 10^4$	350	75	2	0.0002

  - For neural network learning: generate 170 000 pairs (2 positive + 2 negative pairs per 1 subsequence)
- **Measure:** $Precision = \frac{TP}{TP+FP}$, TP – true anomalies detected, FP – false positive anomalies detected
- **Neural Network Hyperparameters:**
  - **Loss function:** contractive loss

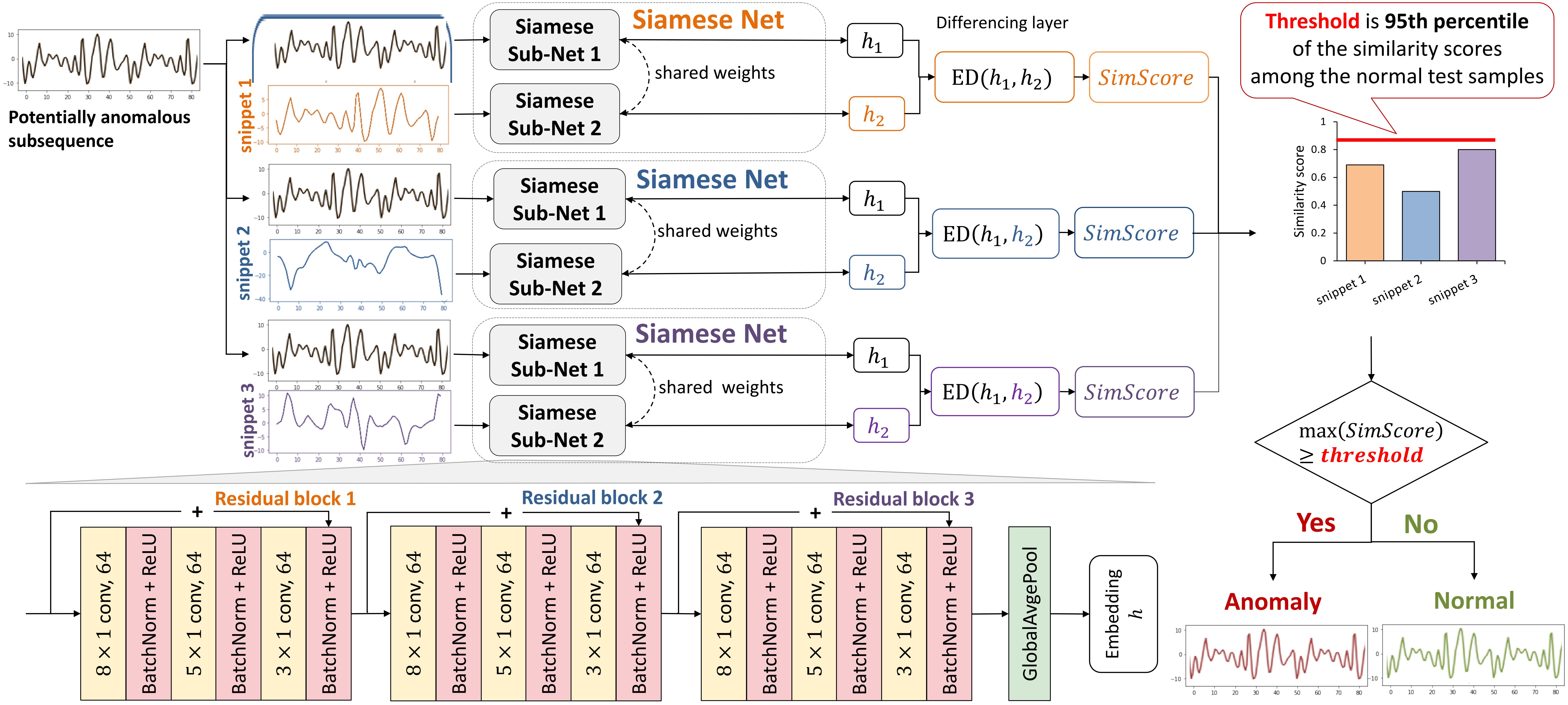
$$L\left(x_i, x_j, \delta_{x_i x_j}\right) = \delta_{x_i x_j} \mathrm{ED}^2(h_i, h_j) + \left(1 - \delta_{x_i x_j}\right) \max\left(0, \tau - \mathrm{ED}(h_i, h_j)\right)^2,$$

where τ – margin between samples of different classes

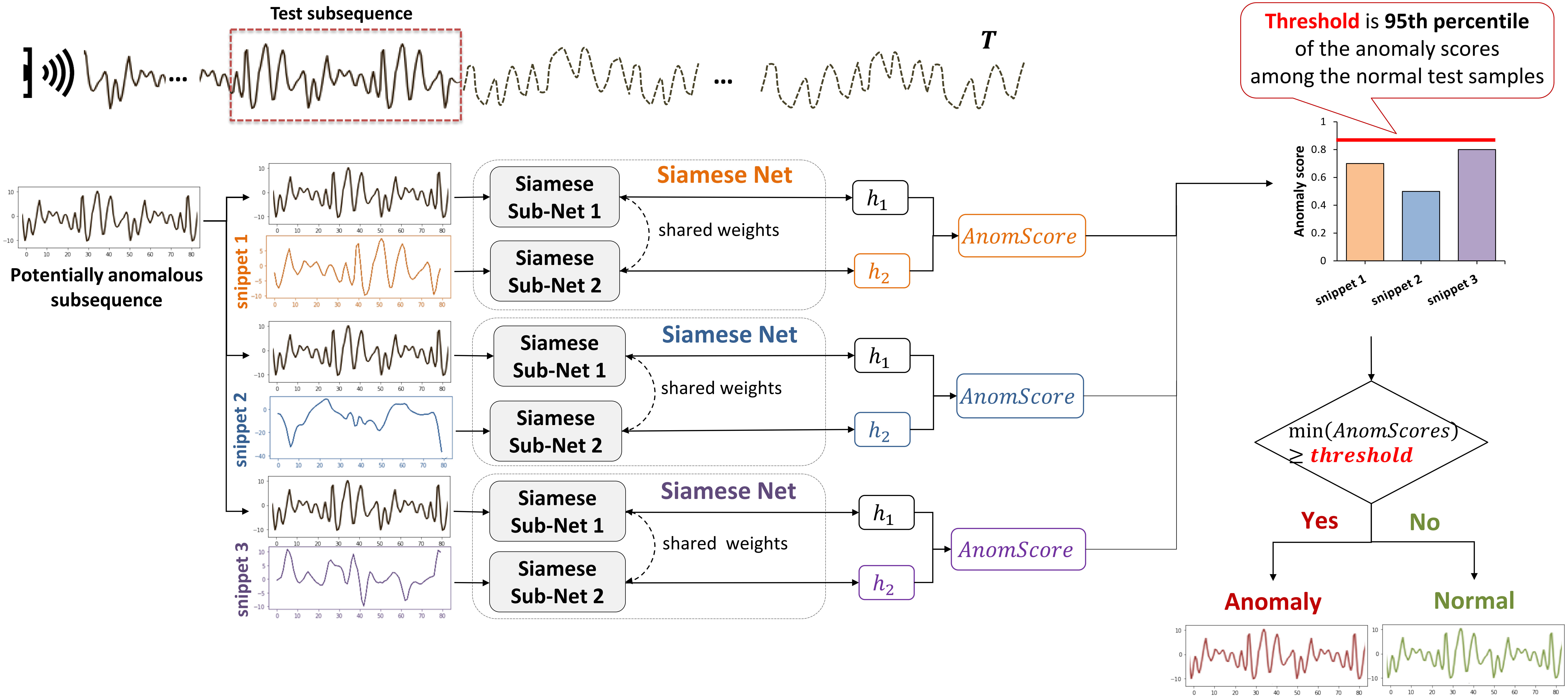
  - **Optimizer:** Adam
  - **# Epochs:** 40
  - **Batch size:** 128
  - **Learning rate:** $1 \cdot 10^{-3}$
  - **Embedding size:** 128

[1)] Goldberger A. L., et al. PhysioBank, PhysioToolkit, and PhysioNet components of a new research resource for complex physiologic signals. Circulation 101(23): 215-220. 2000.

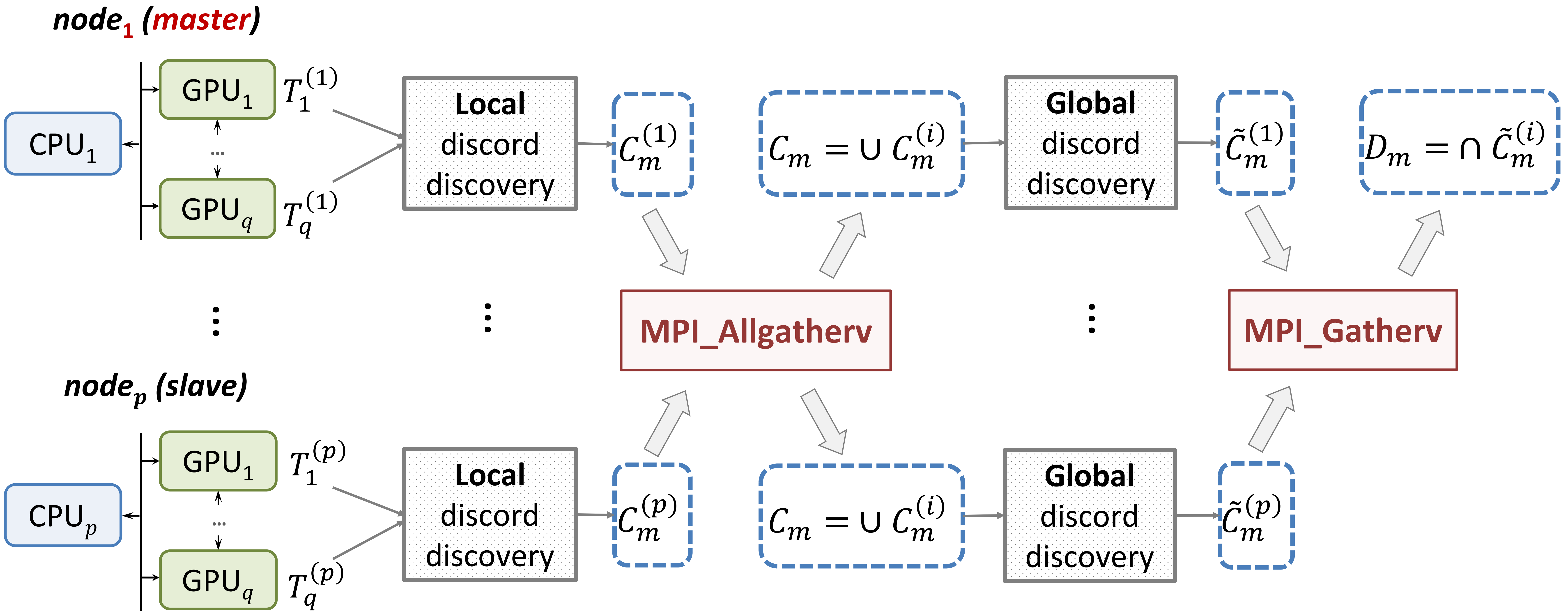
DiSSiD: Discord, Snippet, and Siamese Net-based Detector of anomalies



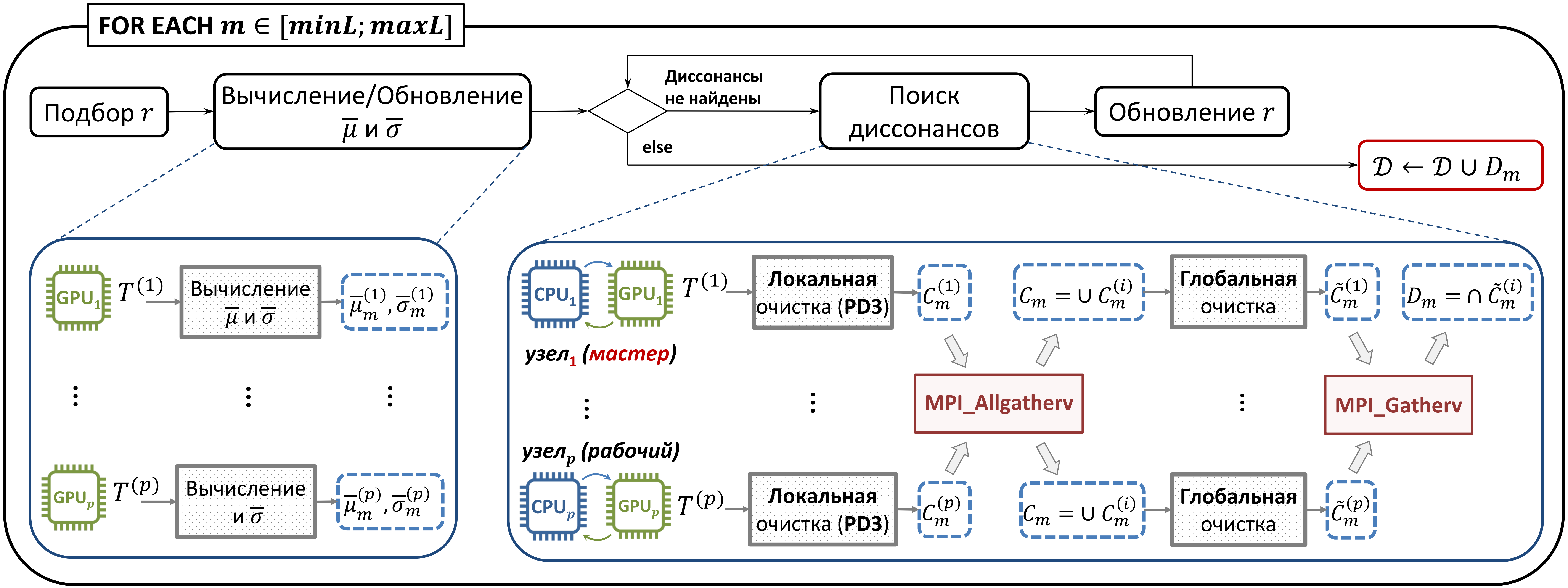
DiSSiD: Discord, Snippet, and Siamese Net-based Detector of anomalies



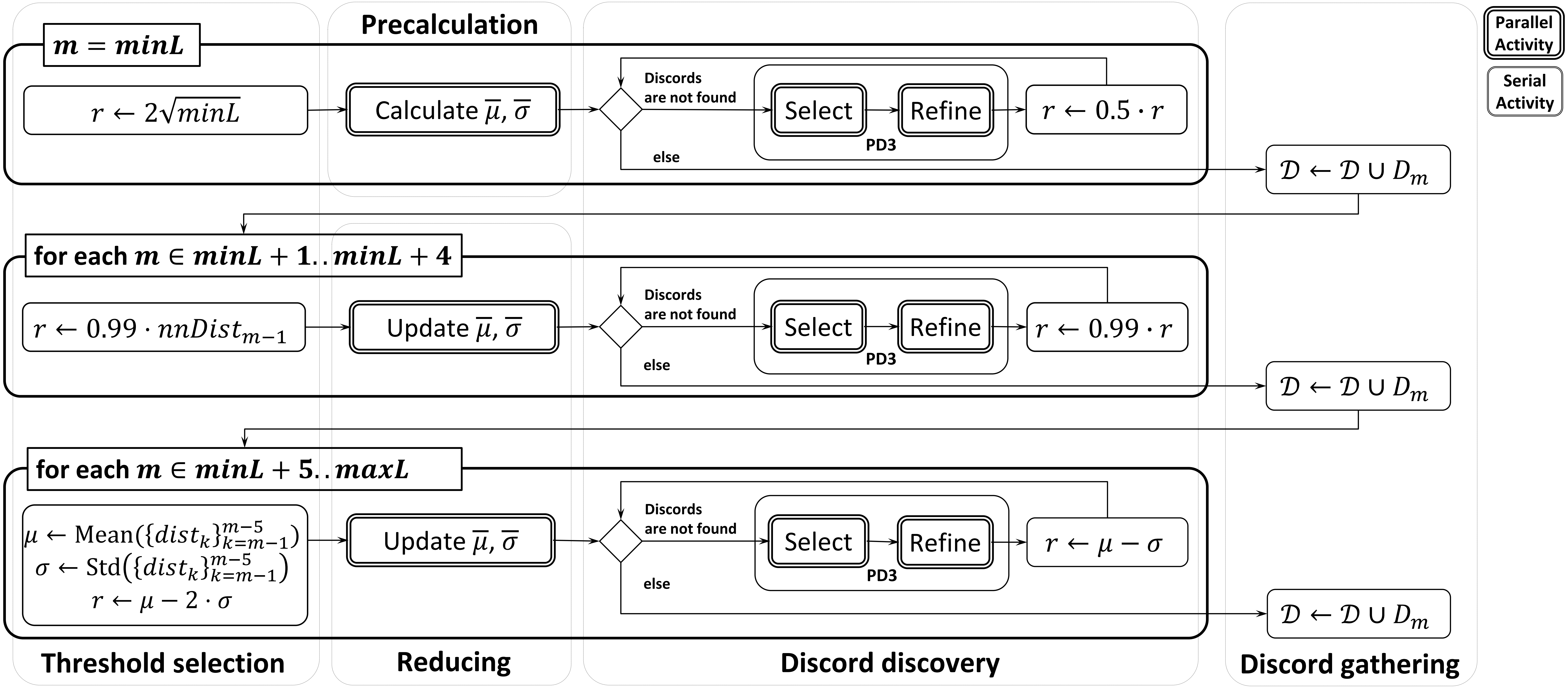
Distributed PALMAD: general scheme

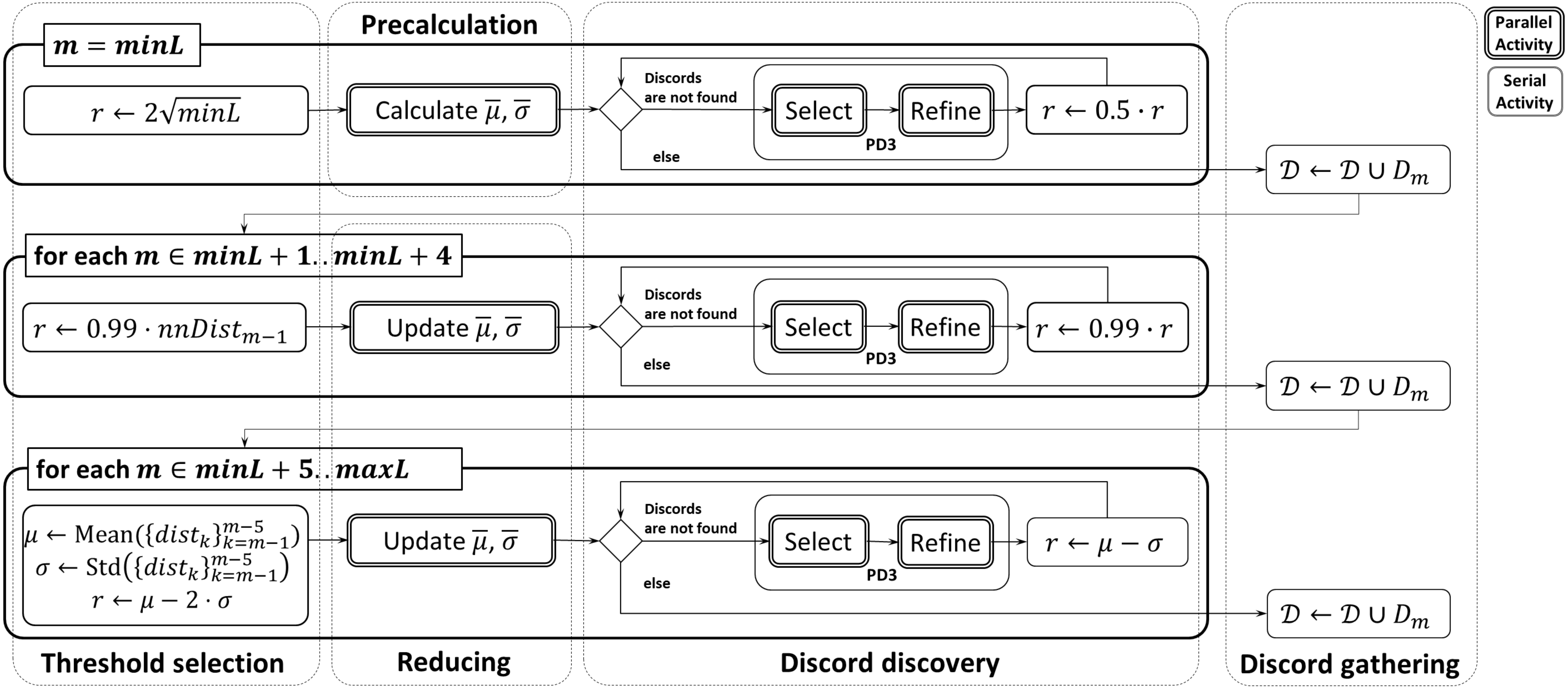


Distributed PALMAD: general scheme



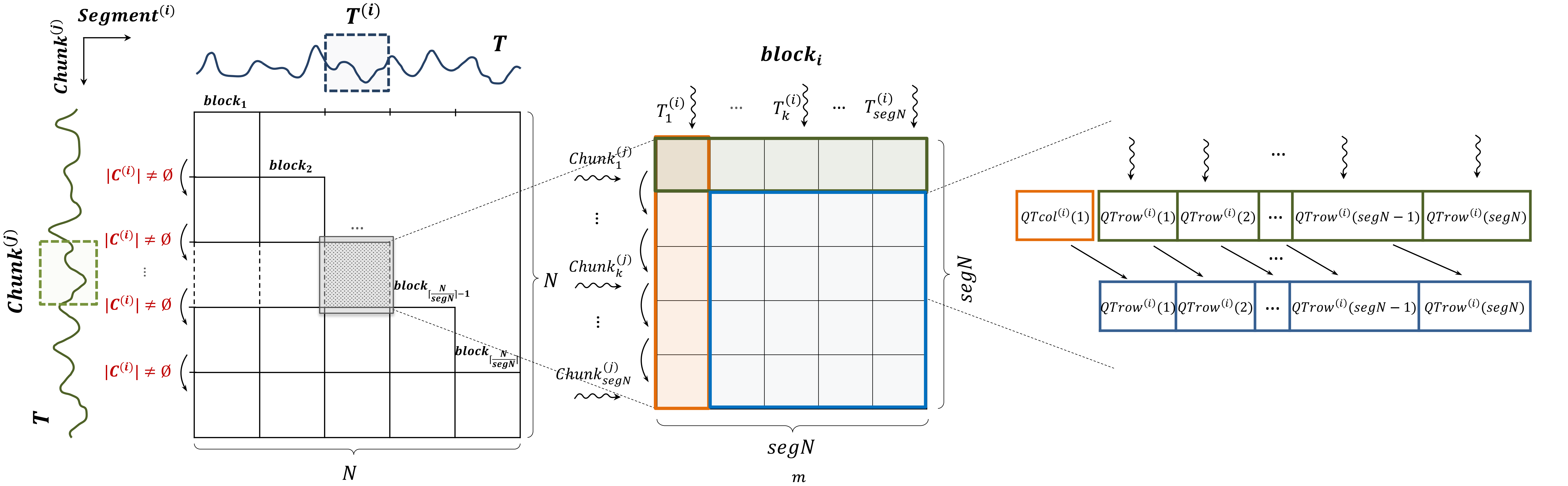
PALMAD (Parallel Arbitrary Length MERLIN-based Anomaly Discovery)[1)]





Precalculation
Parallel Activity
Serial Activity
$m = minL$
$r \leftarrow 2\sqrt{minL}$
Calculate $\overline{\mu}, \overline{\sigma}$
Discords are not found
Select
Refine
PD3
$r \leftarrow 0.5 \cdot r$
else
$\mathcal{D} \leftarrow \mathcal{D} \cup D_m$
for each $m \in minL+1..minL+4$
$r \leftarrow 0.99 \cdot nnDist_{m-1}$
Update $\overline{\mu}, \overline{\sigma}$
Discords are not found
Select
Refine
PD3
$r \leftarrow 0.99 \cdot r$
else
$\mathcal{D} \leftarrow \mathcal{D} \cup D_m$
for each $m \in minL+5..maxL$
$\mu \leftarrow \text{Mean}(\{dist_k\}_{k=m-1}^{m-5})$
$\sigma \leftarrow \text{Std}(\{dist_k\}_{k=m-1}^{m-5})$
$r \leftarrow \mu - 2 \cdot \sigma$
Update $\overline{\mu}, \overline{\sigma}$
Discords are not found
Select
Refine
PD3
$r \leftarrow \mu - \sigma$
else
$\mathcal{D} \leftarrow \mathcal{D} \cup D_m$
Threshold selection
Reducing
Discord discovery
Discord gathering

PD3: Parallel DRAG-based Discord Discovery



$$QTrow^{(i)}(tid) = \sum_{k=1}^{m} T_{tid}^{(i)}(k) \cdot Chunk_1^{(j)}(k)$$

$$QTcol^{(i)}(tid) = \sum_{k=1}^{m} T_1^{(i)}(k) \cdot Chunk_{tid}^{(j)}(k)$$

$$QTrow^{(i)}(tid) = QTrow^{(i)}(tid-1) - T_{tid-1}^{(i)}(1) \cdot Chunk_{tid-1}^{(j)}(1) + T_{tid}^{(i)}(m) \cdot Chunk_{tid}^{(j)}(m)$$