BOOSTING THE EFFICIENCY OF SMART WEARABLE DEVICES THROUGH BIOMETRIC SIGNAL COMPRESSION

Michele Rossi

rossi@dei.unipd.it

Dept. of Inf. Engineering, Univ. of Padova, Italy

SAMSUNG ADVANCED INSTITUTE OF TECHNOLOGY

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Roadmap

- Application scenarios
- Objectives / usage models
- Signal types / datasets / measurements
- Review of compression algorithms
- Subject-adaptive compression
  - SAM: a first design SOM-based design
  - SURF: an enhanced GNG-based design
- Conclusions

Application Scenarios

- Continuous monitoring of outpatients
- Chronic diseases such as
  - Hearth failures
  - Chronic respiratory failures
  - Hypertension

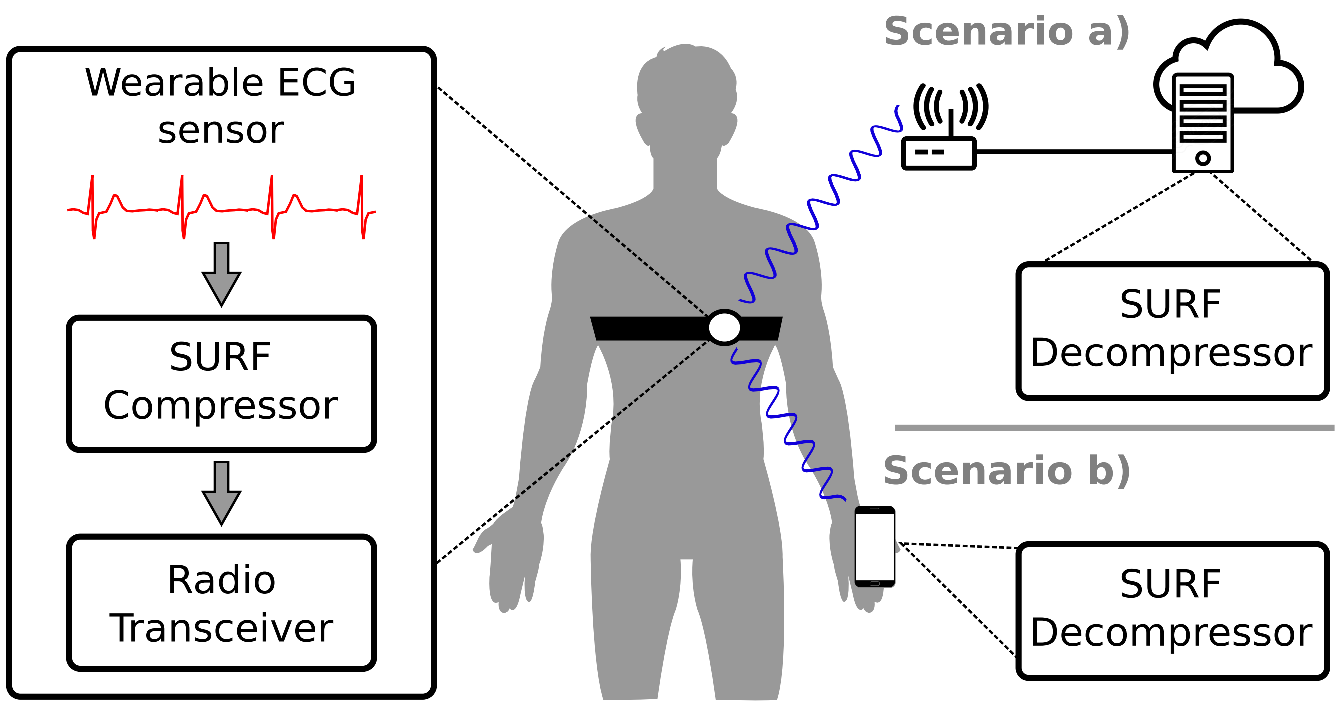
- Wearables
  - Unobtrusive monitoring from home
  - Rapid market expansion (e.g., Equivital, Zephyr Bioharness)
  - Lower cost for the health care system

Objectives

- Wireless wearable (IoT) devices are
  - Energy constrained
  - (memory constrained)
- Improve efficiency through signal processing
  - Data storage & representation compactness
  - Data transmission (wireless interface)
- Proposed solution
  - Lossy compression of physiological signals

Usage model

- TX biometric signal to
  - Personal device (e.g., Bluetooth)
  - Access point (e.g., Wi-Fi)

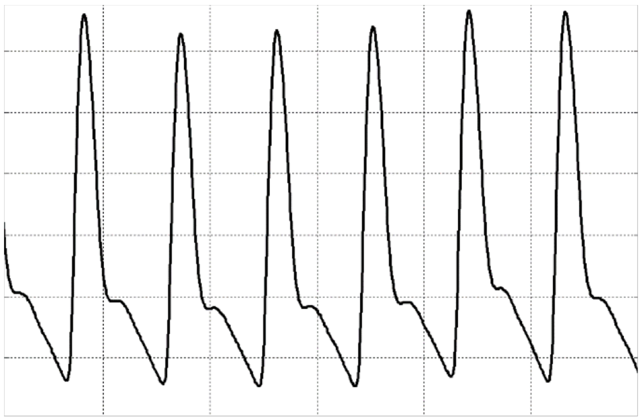


Targeted Signal Types

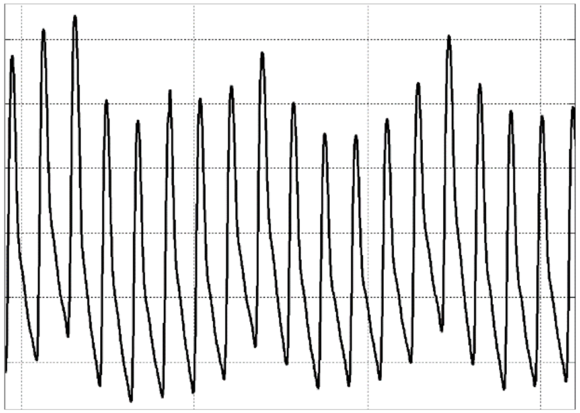
- Electrocardiogram (ECG)
- Photoplethysmography signal (PPG)
- Respiration signal (RESP)
- Arterial blood pressure (ABP)
- Note: they all are *quasi-periodic* signals

ECG

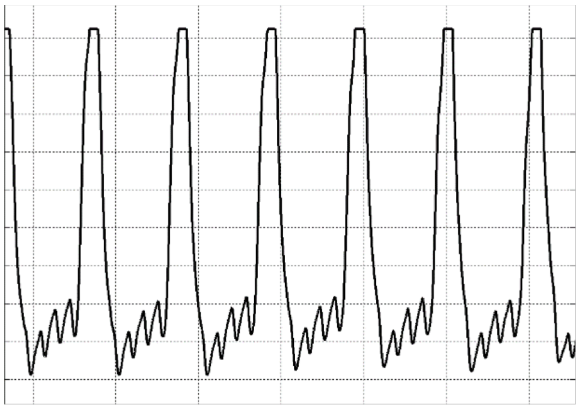
ABP



PPG



RESP



Biomedical datasets

ECG traces

- MIT-BIH arrhythmia database: 350 samples/s, 11 bit resolution Traces: 101, 112, 115, 117, 118, 201, 209, 212, 213, 219, 228, 231, 232

PPG and RESP traces

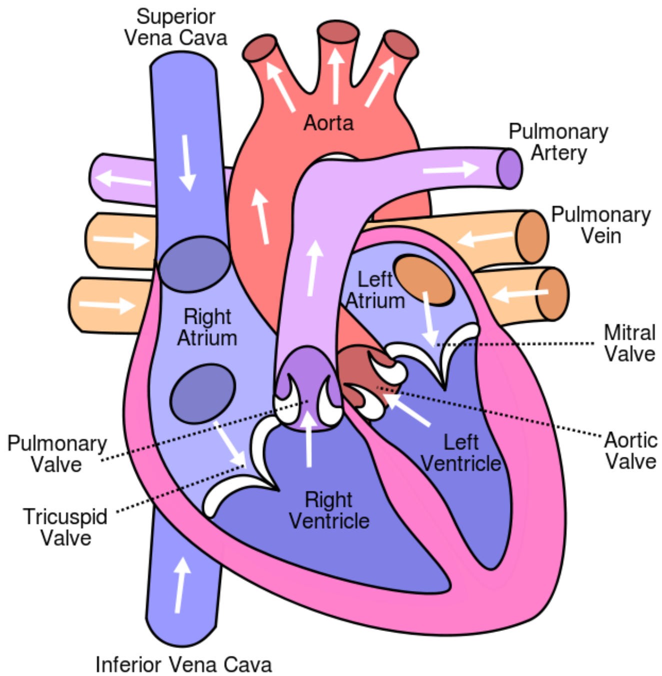
- Physionet MIMIC II database: 125 samples/s, 12 bit resolution

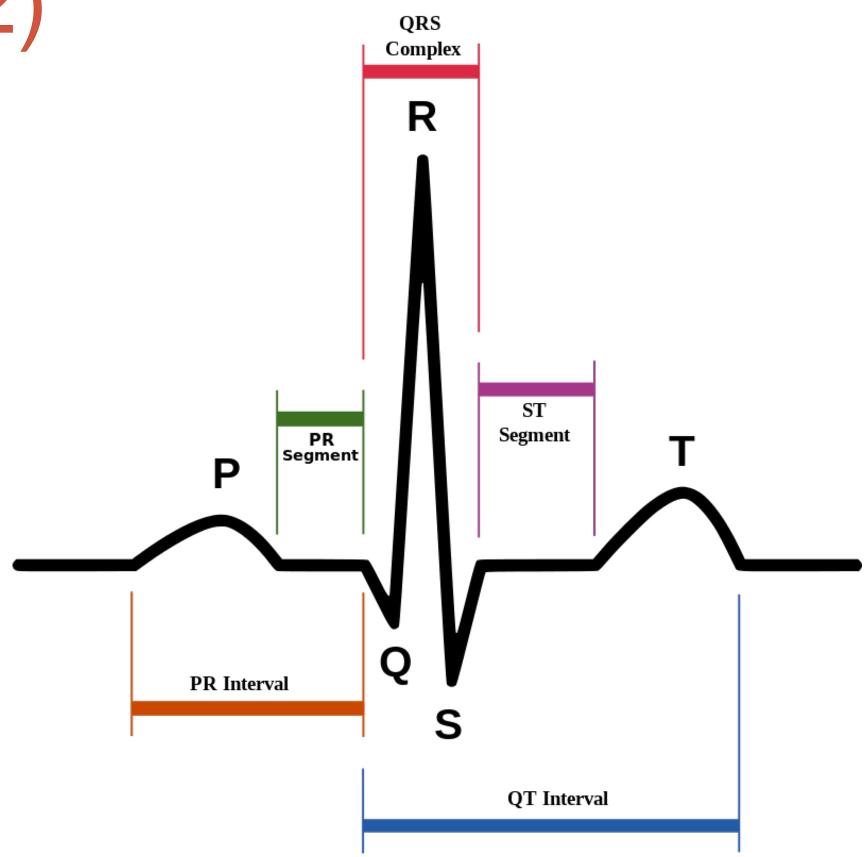
[S++11] M. Saeed and M. Villarroel and A.T. Reisner and G. Clifford and L. Lehman and G.B. Moody and T. Heldt and T.H. Kyaw and B.E. Moody and R.G. Mark, "Multiparameter Intelligent Monitoring in Intensive Care II (MIMIC-II): a public-access intensive care unit database," *Critical Care Medicine*, vol. 39, no. 5, May 2011.

Own collected ECG traces

- Zephyr Bioharness 3 (chestband)
- From 11 healthy individuals
- Continuously recorded form 8am to 6pm
  - Free-linving conditions
- Sample rate 250 samples/s
- Each sample takes 12 bits
- Artifacts prone (important)

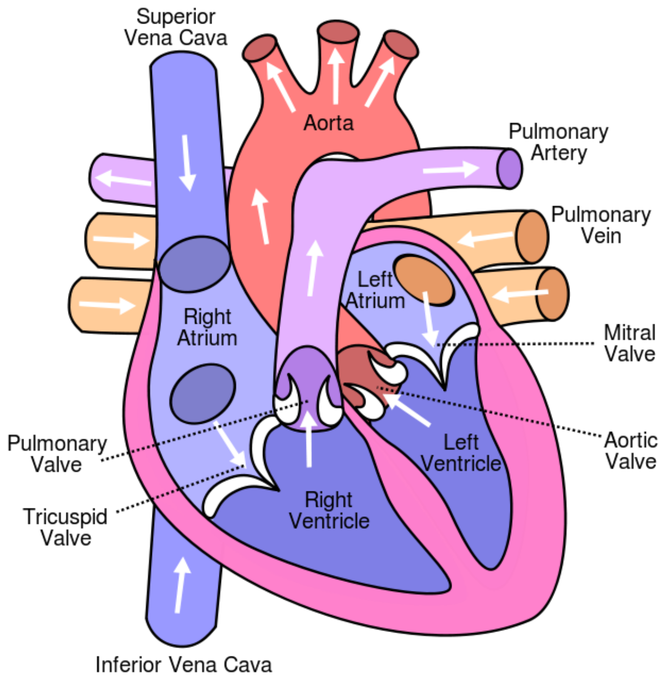
The heart muscle (1/2)

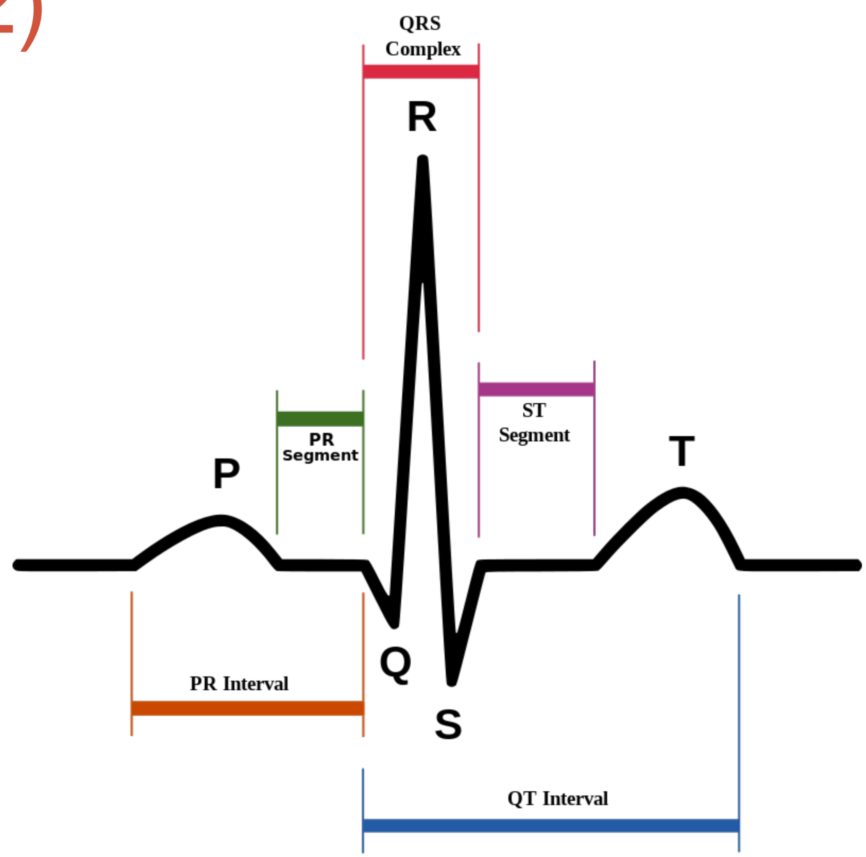




- Human heart: specialized *muscle* that pumps blood throughout the body through the blood vessels of the circulatory system. It has four chambers: two upper chambers (right and left atrium) and two lower ones (right and left ventricle). Heart activity follows a cyclic pattern. The normal *rhythmical heartbeat* (called *sinus rhythm*) is established by the sinoatrial node, the heart's natural pacemaker in the upper part of the right atrium.

The heart muscle (2/2)

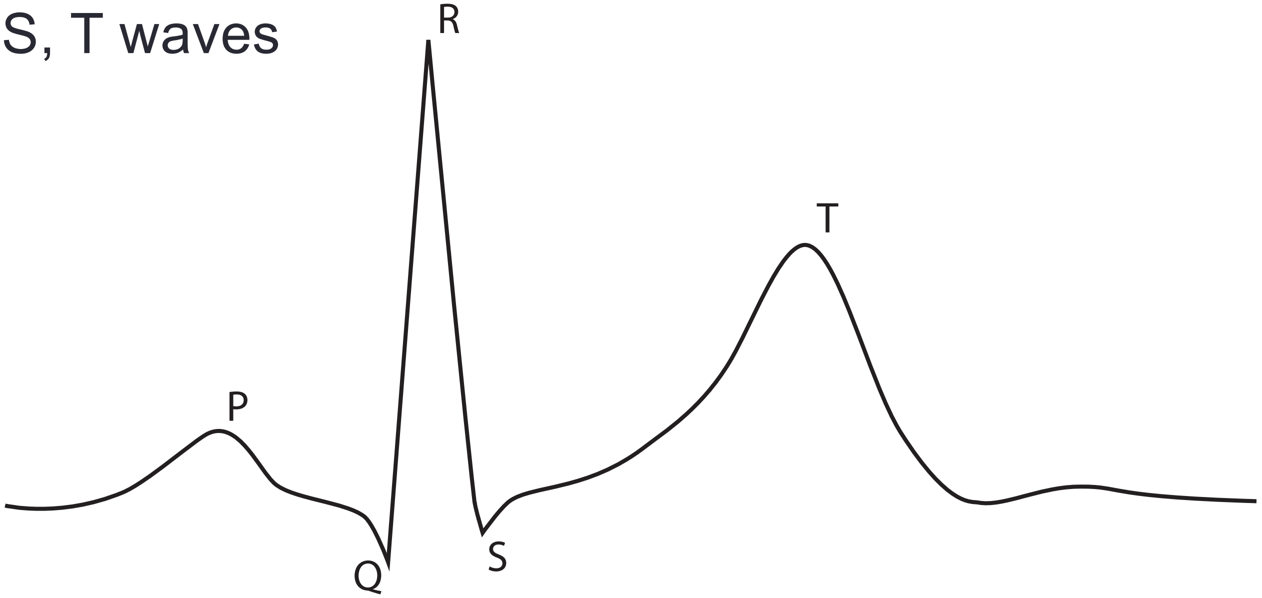




- During cardiac cycles: *de-oxygenated blood* transported by the veins passes from the right atrium to the right ventricle and then to the lungs, where carbon dioxide is released and oxygen is absorbed. At the same time, *oxygenated blood* (from previous cycle) enters the left atrium, passes through the left ventricle and is pumped through the *aorta* and *arteries* to the rest of the body.

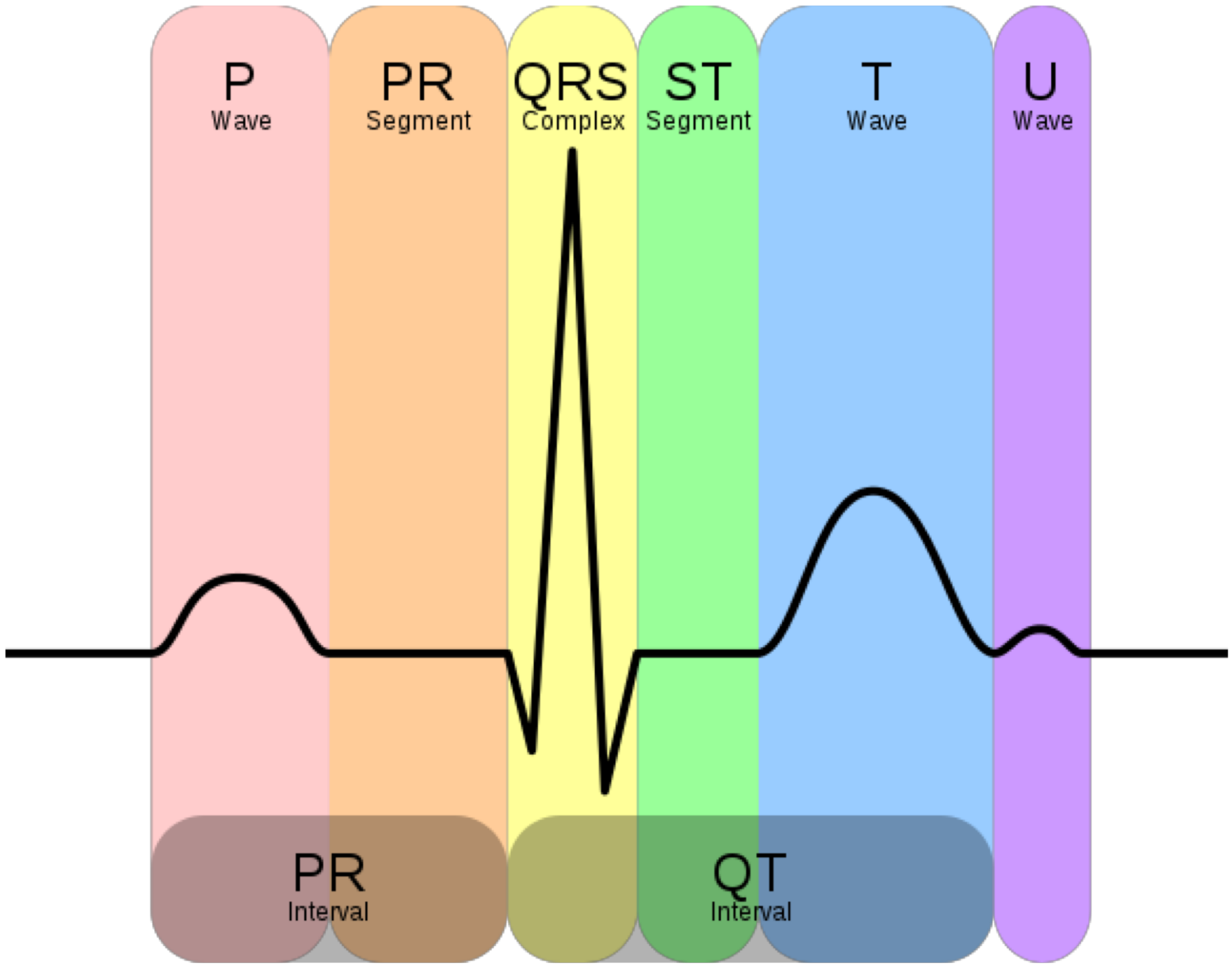
The ECG complex (1/4)

- Normal resting adult human heart rate
  - form 60 to 100 beats per minute (bpm)
- The ECG records heart electrical activity
  - Three complexes: P, QRS and T complexes (see figure)
    - QRS complex is particularly important: captures the ventricular depolarization leading to ventricular contraction
  - Five waves: P, Q, R, S, T waves



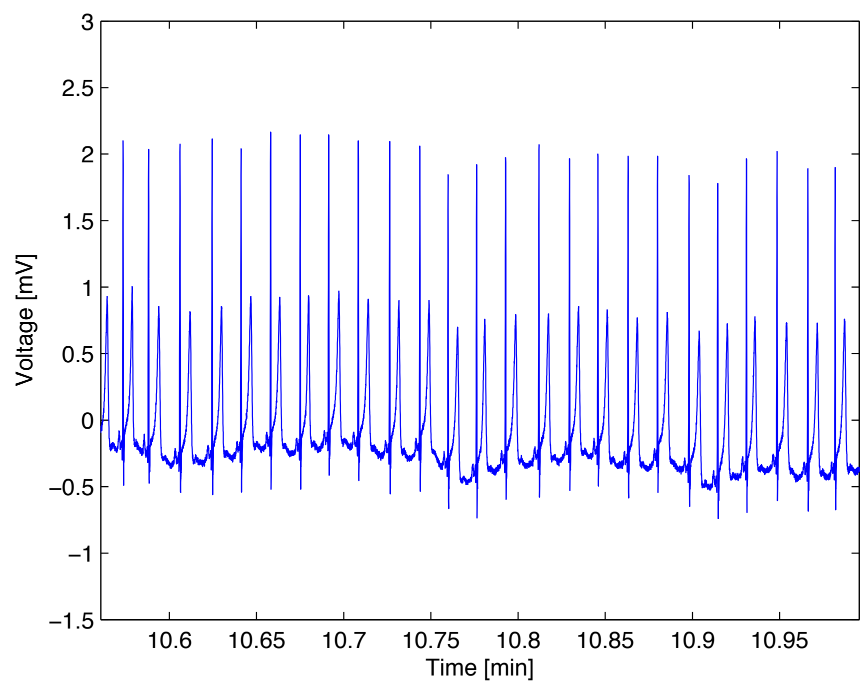
The ECG complex (2/4)

- Quantities of interest in a nutshell



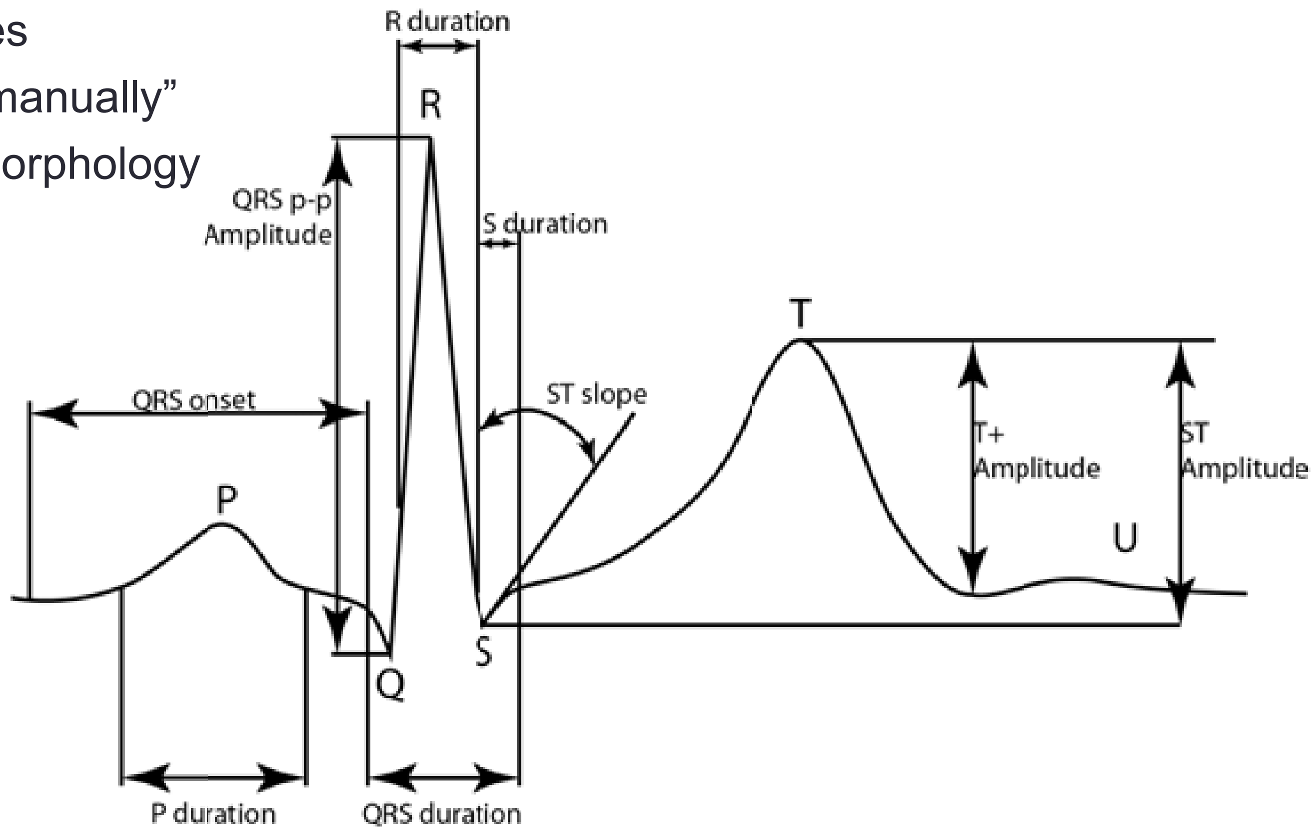
The ECG complex (3/4)

- Duration, amplitude and morphology of the waves
  - Are useful in diagnosing cardiovascular diseases
- R peaks
  - Are the most high frequency components
  - Are used to keep track of the heart rate (to assess rhythm regularity)
- ECG signal
  - Is quasi-periodic
  - Amenable to dictionary-based analysis
  - First step is extraction of ECG segments (segment between subsequent R peaks)



The ECG complex (4/4)

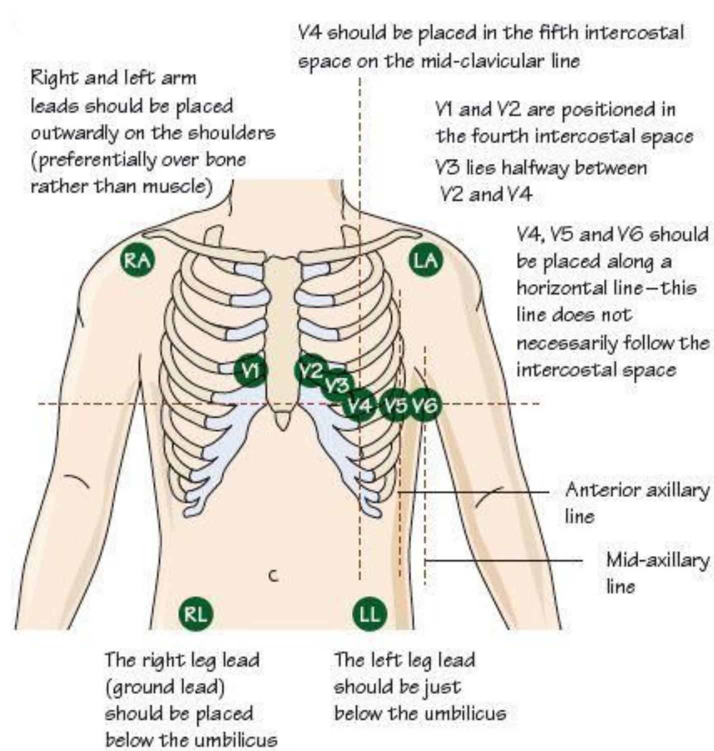
- Existing ECG analysis approaches
  - Identify measurable features
  - Features are determined “manually”
  - To characterize the ECG morphology



- [Biel01] L. Biel, O. Petterson, L. Philipson, P. Wide, “ECG Analysis: a New Approach in Human Identification,” *IEEE Transactions on Instrum. Meas.*, Vol. 50, No. 3, pp. 808-812, June 2001.

3-, 6-, 12-lead ECG

- Standard ECG measurement systems
- Signal recorded by several electrodes on the body surface
- Sensors are commonly defined to as “wet”
  - Require the use of *conductive gel* between the skin and the sensors
- In this study, we consider
  - Single-lead & dry sensors
  - Multi-lead is difficult with wearables
  - Electrodes have to be dry (no gel)
  - Signals are artifact-prone
- Dimensionality reduction of ECG
  - Much more challenging



ECG QRS detection (segment extraction)

- Many techniques are available
  - Standard and well studied problem
  - Pan-Tompkins [Pan85] is the de-facto standard algorithm
  - Often assumed very reliable & utilized
    - for comparison (ground truth) against new schemes
  - We use it: no details provided on QRS detection

- Reference Matlab implementation (2014)
  - https://it.mathworks.com/matlabcentral/fileexchange/45840-complete-pan-tompkins-implementation-ecg-qrs-detector

- [Pan85] Jiapu Pan, Willis J. Tompkins, “A Real-Time QRS Detection Algorithm,” *IEEE Transactions on Biomedical Engineering*, Vol. BME-32, No. 3, pp: 230-236, March 1985.

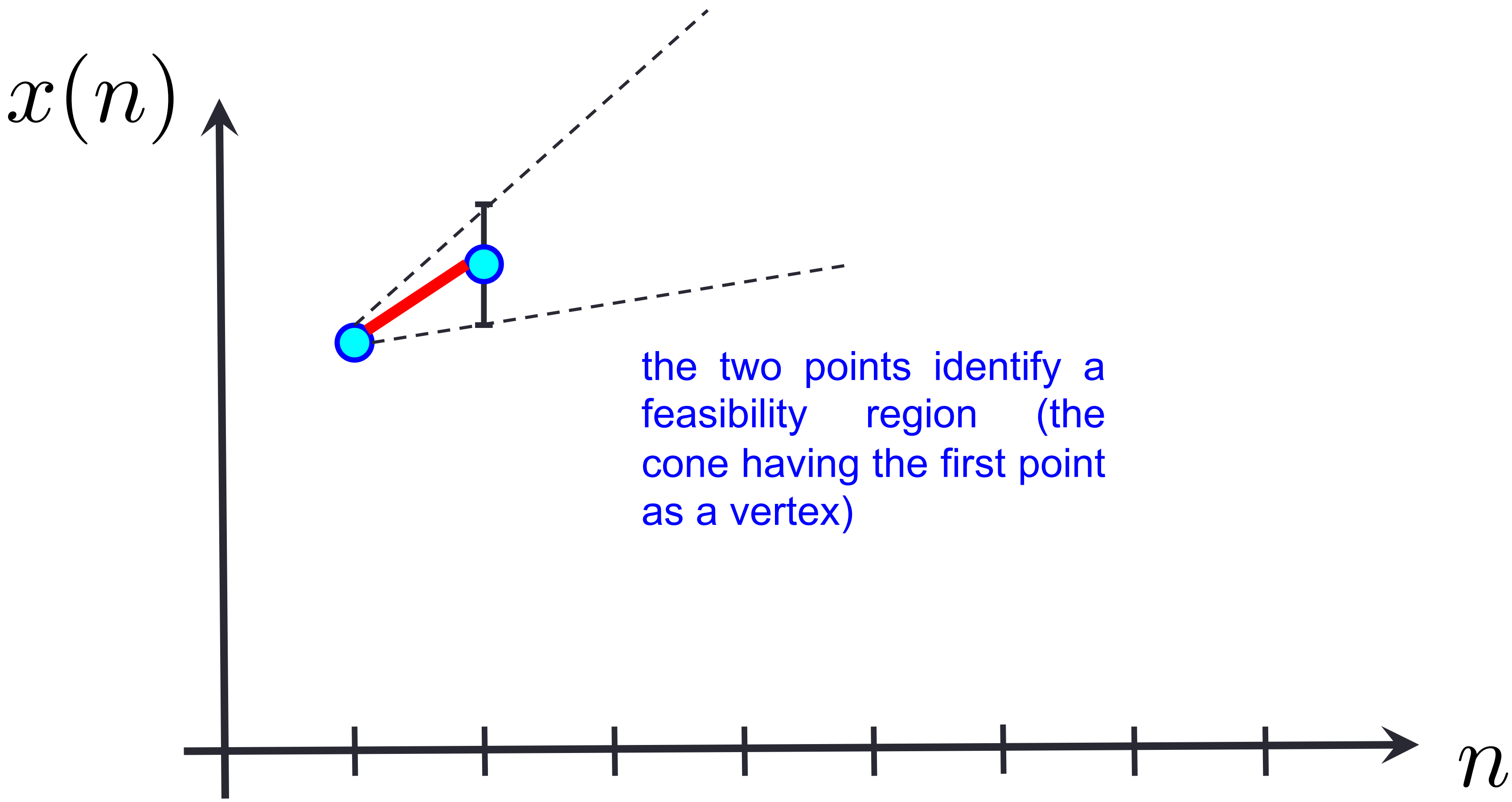
REVIEW OF ENERGY COMPRESSION ALGORITHMS

Existing compression approaches [ZMVR15]

- Transform based
  - Represent the signal in a suitable transform domain
  - Fast Fourier Transform (FFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Compressive Sensing (CS)
- Time domain processing
  - Discard some of the samples and apply linear approximations
  - AZTEC, CORTES and Lightweight Temporal Compression (LTC)
- Parametric models
  - Process the times series to obtain some kind of knowledge
  - Use that knowledge to predict the signal behavior
  - Neural networks, vector quantization (GSVQ), pattern matching

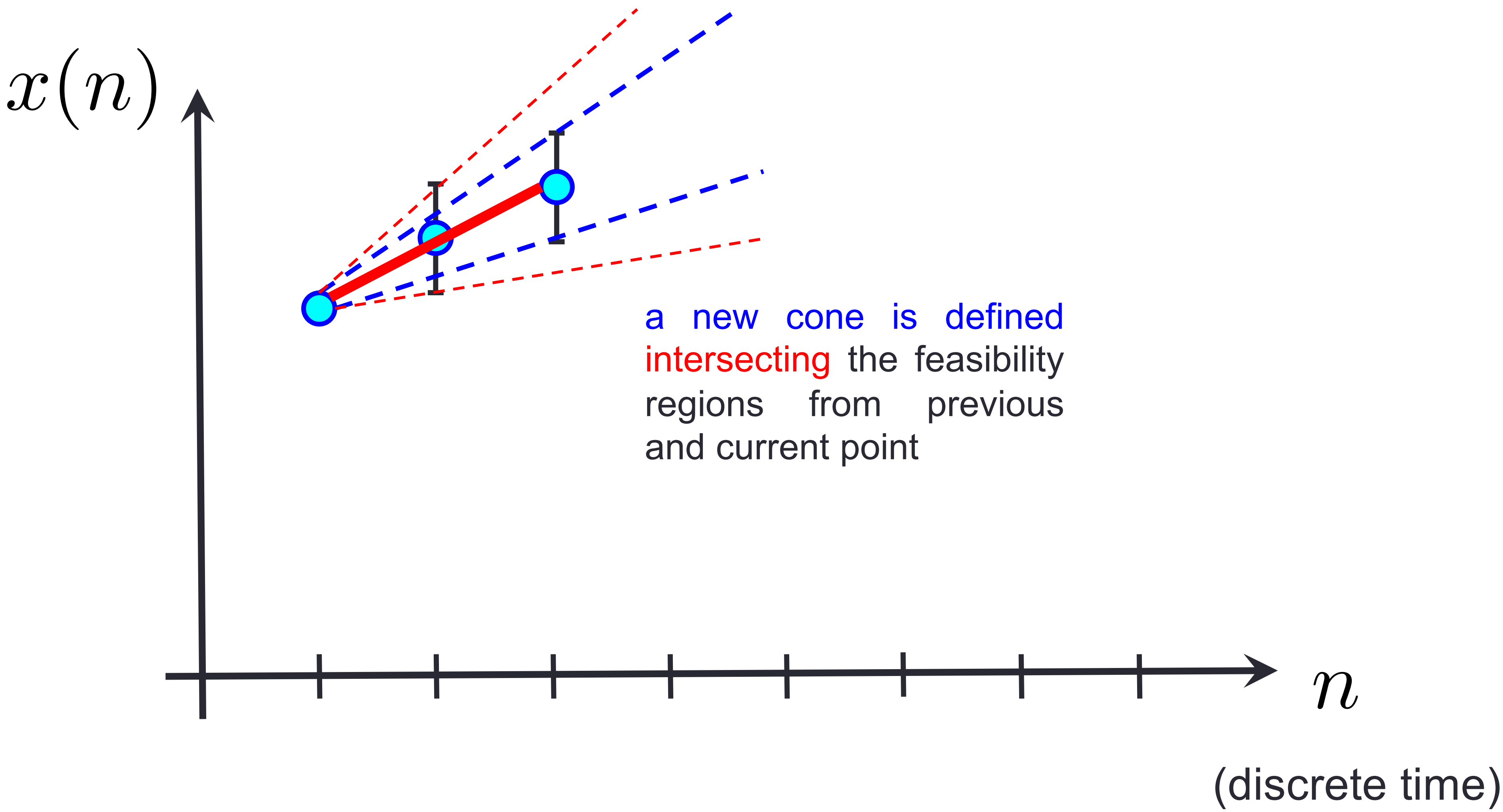
[ZMVR15] D. Zordan, B. Martinez, I. Vilajosana, M. Rossi, "On the Performance of Lossy Compression Schemes for Energy Constrained Sensor Networking," *ACM Transactions on Sensor Networks*, Vol. 11, No. 1, 2014.

LTC



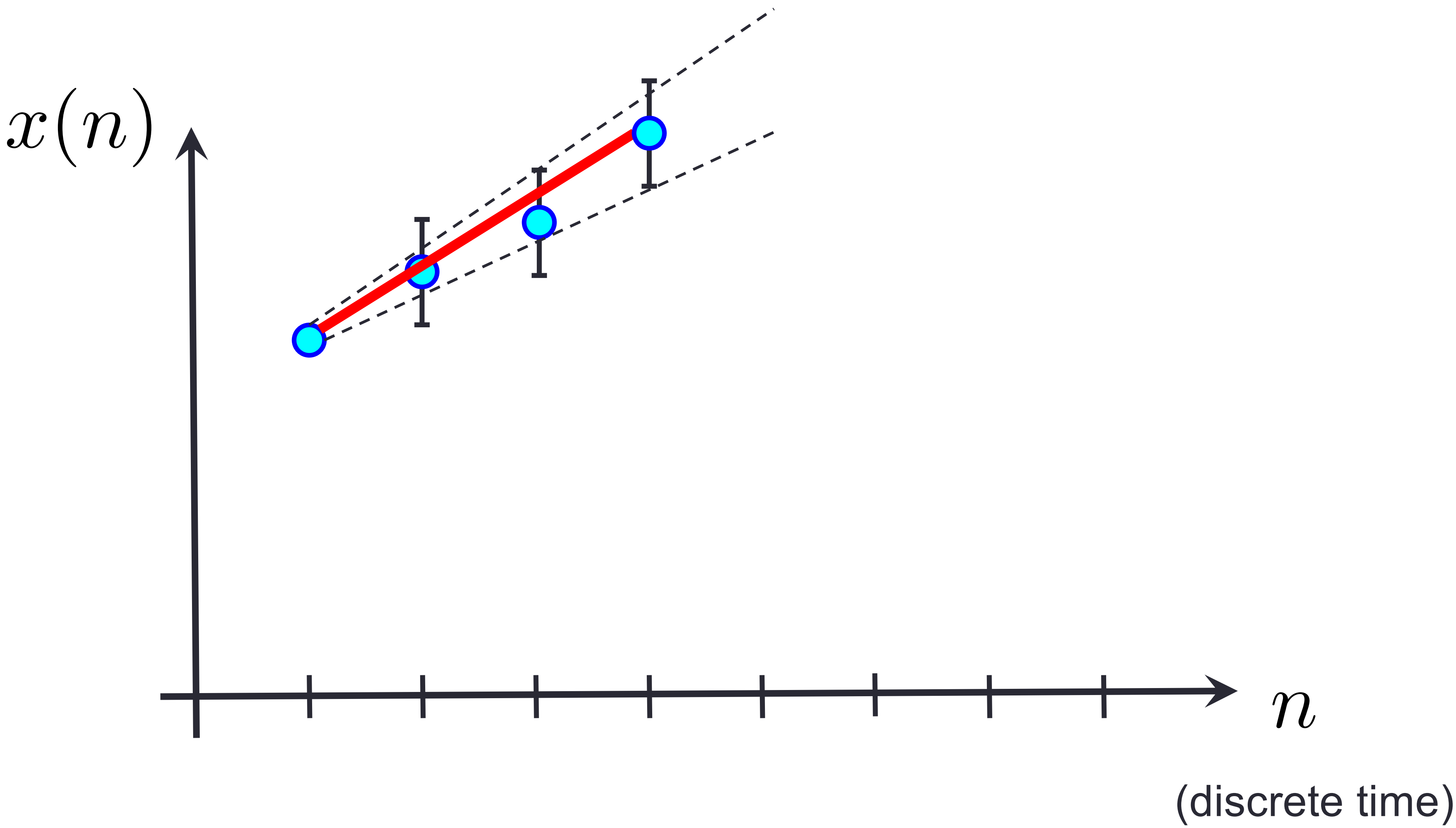
[S++04] T. Schoellhammer, B. Greenstein, M. Wimbrow E. Osterweil, and D. Estrin, “Lightweight temporal compres- sion of microclimate datasets,” in *Proceedings of the IEEE International Conference on Local Computer Net- works (LCN)*, Tampa, FL, US, Nov. 2004.

LTC

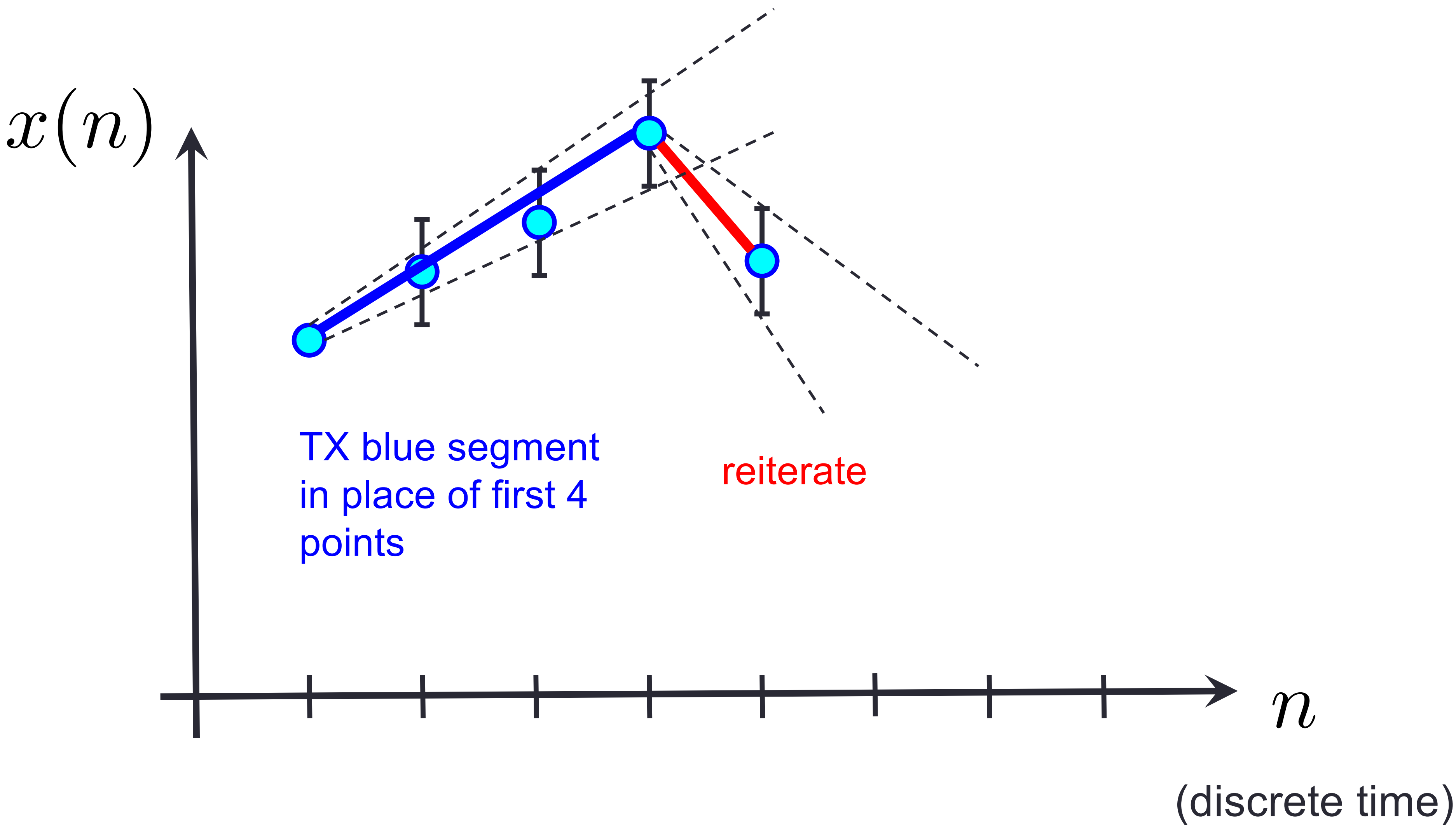


a new cone is defined intersecting the feasibility regions from previous and current point

LTC

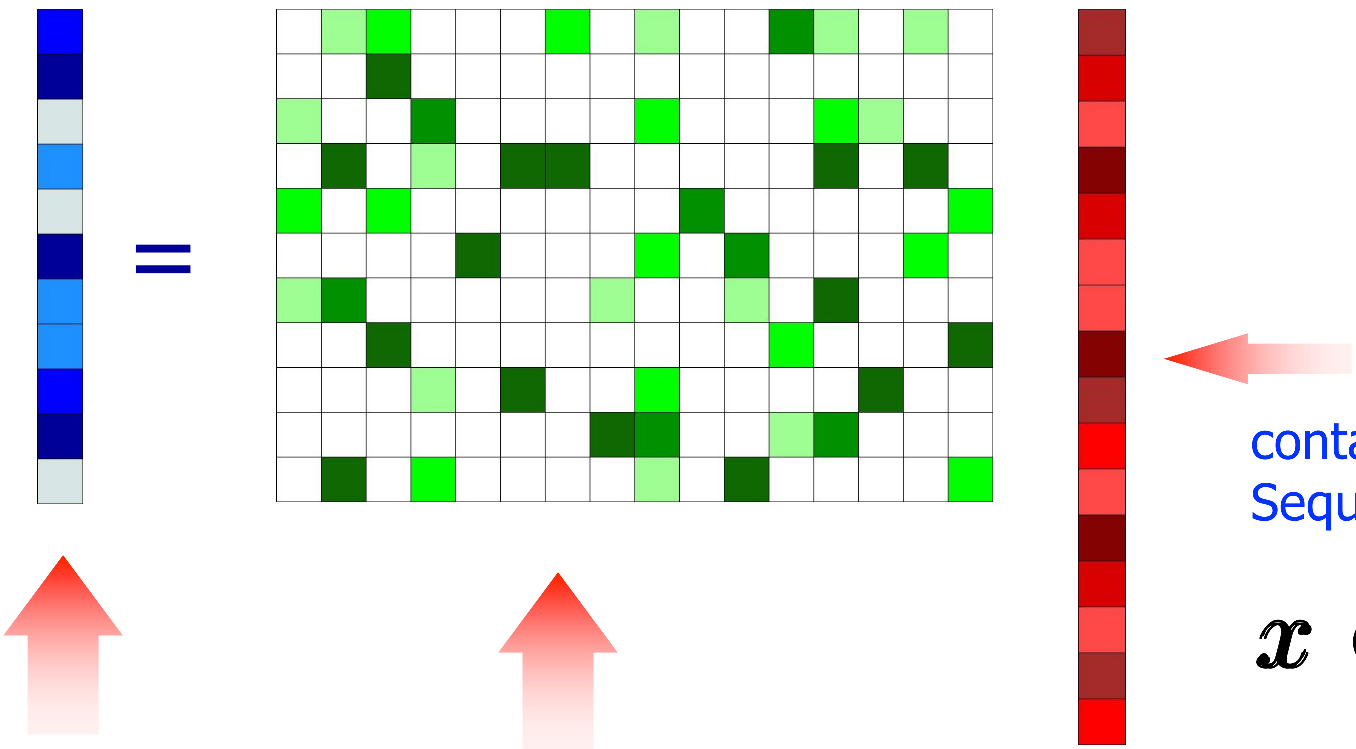


LTC



$x(n)$
TX blue segment
in place of first 4
points
reiterate
n
(discrete time)

Compressive Sensing (CS) in a nutshell



x

contains full data
Sequence (in space / time)

$$\boldsymbol{x} \in \mathbb{R}^N$$

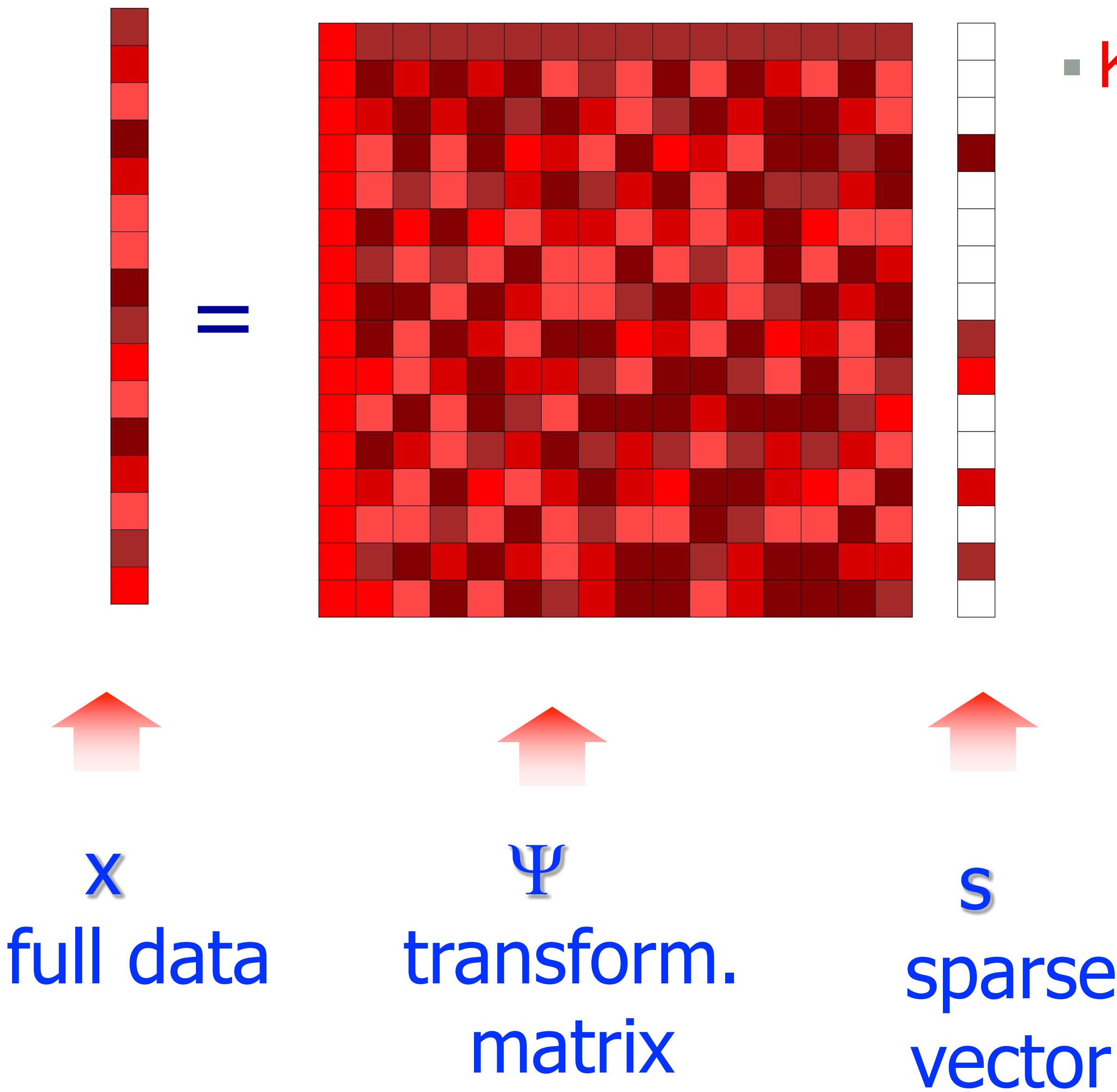
y

data at collection point
M dimensions (M<N)

Φ

measurement (or sampling) matrix – produces linear (random) combinations of input data vector

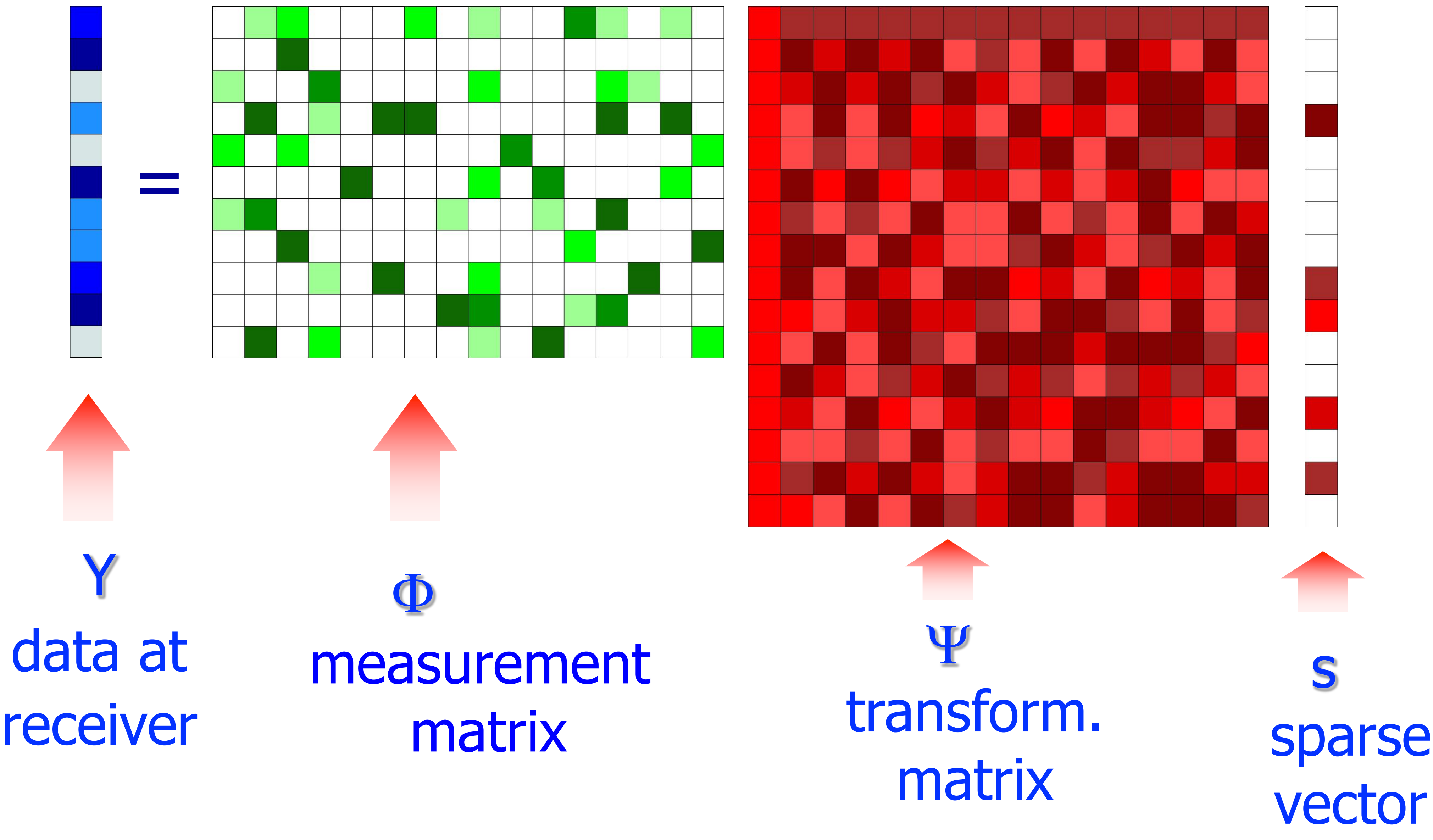
Sparse Signal



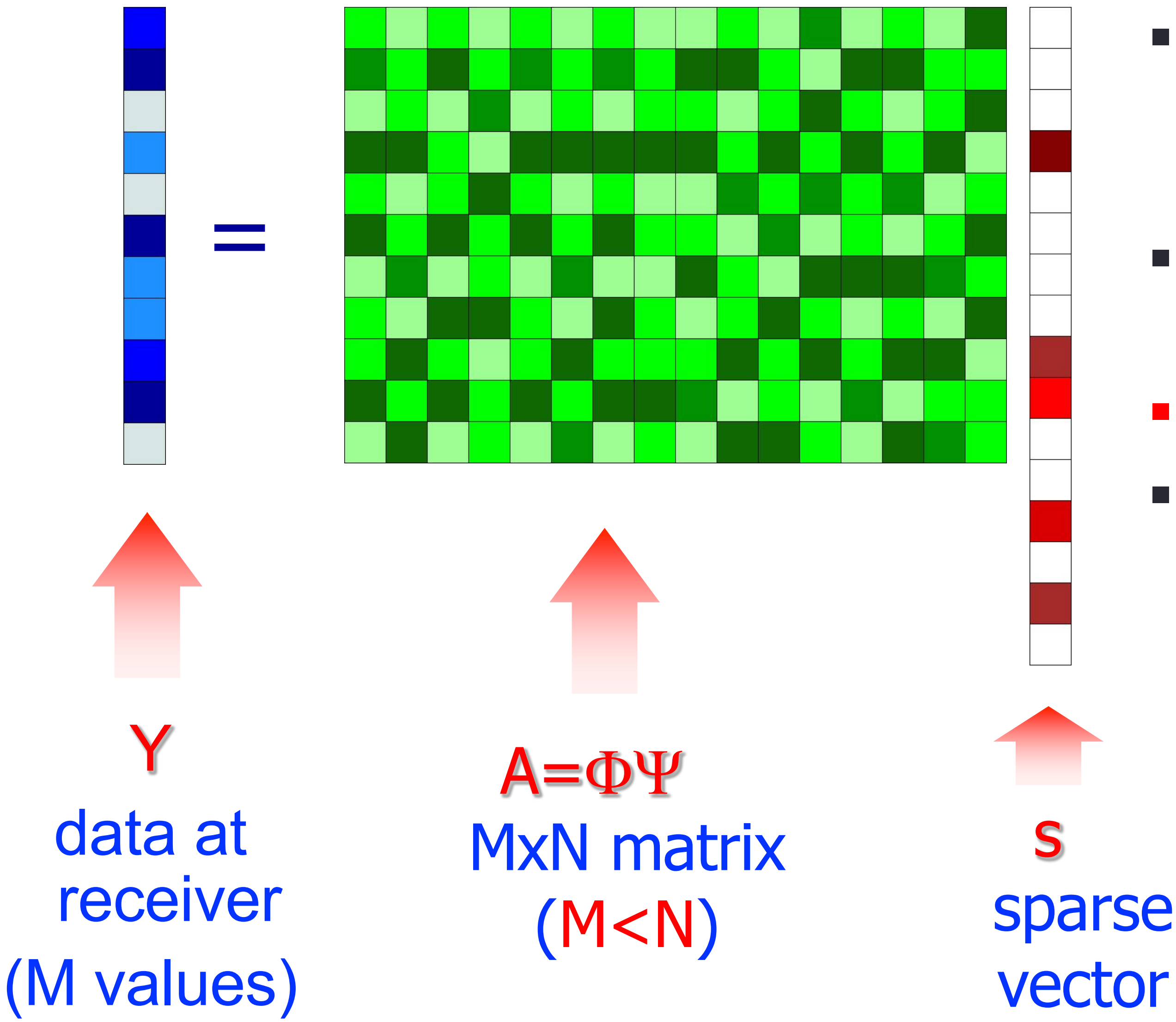
- Key assumption
  - There exists a domain under which the signal is sparse, Ψ is the basis that sparsifies the signal (known at the receiver)

$$\boldsymbol{x} = \boldsymbol{\Psi} \boldsymbol{s}$$

Replacing “x” with “Ψs”



We finally get



- Linear system, with M equations, N>M unknowns
- N-M deg. of freedom (ill-posed problem)
- Norm-1 minimization
- Solvers: NESTA, l1-magic, Subspace Pursuit, …

$$\boldsymbol{y} = \boldsymbol{\Phi\Psi s}$$

Compressive Sensing – some basic math (1/3)

- Input signal $\boldsymbol{x} \in \mathbb{R}^N$
- Measurement matrix
  - It is an MxN sampling matrix $\boldsymbol{\Phi}$
  - Example matrices:
    - a single one for each row (transform coding)
    - Random Gaussian entries
    - Random entries in {-1,+1}
- At the collection point, we measure
  - Vector **y** (M elements), that is a combination of the field data (vector **x**)
  - Vector **n** represents measurement noise

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x} + \boldsymbol{n}$$

- CS goal
  - Reconstruct data vector **x** from measured vector **y**
  - Seemingly impossible, as the system of equation is MxN
  - M measures, N unknown variables (ill-posed system of equations)

Compressive Sensing – some basic math (2/3)

- But
  - what is there is a transformation, under which the input vector is sparse?
  - Sparsity: only a small subset K<<N of elements are non-negligible
- Such transformation (NxN matrix) is called $\mathbf{\Psi}$
  - Transformation must be NxN and invertible (e.g., DCT, PCA)
  - Under this transformation, we get:

$$\boldsymbol{x} = \mathbf{\Psi}\boldsymbol{s}$$

- Putting it all together, we get:

$$\boldsymbol{y} = \mathbf{\Phi}\boldsymbol{x} + \boldsymbol{n} = \mathbf{\Phi}\mathbf{\Psi}\boldsymbol{s} + \boldsymbol{n}$$

Compressive Sensing – some basic math (3/3)

- From [Candes06]
  - If the signal s is sparse in the transformed domain
  - And if matrix product $\mathbf{\Phi\Psi}$ meets the *"Restricted Isometric Property"*
  - The previous ill-posed system can be inverted with high probability
  - Inversion occurs looking at the sparsest solution, i.e.,

$$\hat{\boldsymbol{s}} = \arg\min \|\boldsymbol{s}\|_0 \,, \text{ subject to: } \|\boldsymbol{y} - \mathbf{\Phi\Psi}\boldsymbol{s}\|_2 \leq \varepsilon$$

- Problem
  - **Norm-0 is non-convex!!!!** L0 minimization problems are NP hard, which means considered very hard to solve

- [Candes06] E. Candes, J. Romberg, T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

Compressive Sensing – RIP & sparsity

- Definition. s-sparse vector.
  - A vector **x** is defined as s-sparse if it has at most s non-zero entries

- Definition. Restricted Isometric Property (RIP)

$$\mathbf{\Phi}' \triangleq \mathbf{\Phi}\mathbf{\Psi}$$

  - For all integers s=1,2,..., define the *isometric constant* of a matrix as the smallest number δ_s in (0,1) such that

$$(1-\delta_s)\|\boldsymbol{x}\|_2^2 \leq \|\mathbf{\Phi}'\boldsymbol{x}\|_2^2 \leq (1+\delta_s)\|\boldsymbol{x}\|_2^2$$

holds for all sparse vectors s.

If RIP holds, then the matrix has sufficient structure to allow the perfect reconstruction of the sparse input vector (it is a *sufficient condition*).

Compressive Sensing – L0 vs L1 norms (1/3)

$$\|\boldsymbol{x}\|_p = \begin{cases} \left(\sum_{i=1}^{N} |x_i|^p\right)^{1/p} & p \in [1, +\infty) \\ \max_{i=1,2,\ldots,N} |x_i| & p = +\infty \end{cases}$$

- L0 norm – non-convex:

$$\|s\|_0 = \#(s_i | s_i \neq 0)$$

- L1 norm – convex:

$$\|\boldsymbol{s}\|_1 = \sum_{i=1}^{N} |s_i|$$

Compressive Sensing – L0 vs L1 norms (2/3)

- Best approximations of a 2D vector **x** using different norms
  - Typically we use norms to (i) approximate strength of a signal, (ii) the size of an error. Now, suppose we are given a 2D vector **x** and would like to approximate it with a given 1D space A
  - If we measure the approximation error using an Lp norm, our task is find a point

$$\hat{\boldsymbol{x}} \in \mathbb{R}^2, \quad \text{such that } \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_p \text{ is minimized}$$

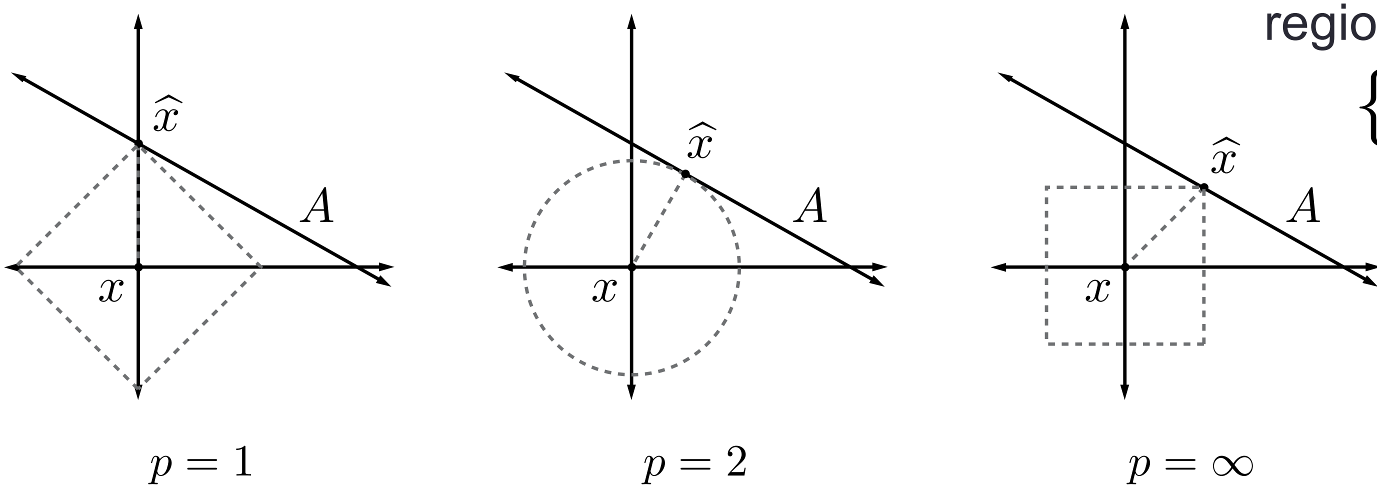
  - This will be the point that is closest to **x** in the corresponding Lp norm
  - The choice of the norm has an effect on the approximation error

Compressive Sensing – L0 vs L1 norms (3/3)

- Best approximations of a 2D vector **x** using different norms
  - Dashed geometries below corresponds to regions where the norm is constant. These regions are centered around point **x**
  - We build spheres centered on **x**, that represent regions where the norm is constant (equal to K). We grow spheres centered on **x** until they intersect with A (given)

Dashed lines:
regions with constant norm

$$\{\boldsymbol{x} : \|\boldsymbol{x}\|_p = K\}$$



As shown, the L1 norm (p=1) is the one that sparsifies the most the solution (the approximation).This holds in general for any number of dimensions

Using L1 norm

- We *relax* the L0 minimization problem
  - Turning it into an L1 minimization problem (*quadratic problem*)

$$\hat{\boldsymbol{s}} = \arg\min \|\boldsymbol{s}\|_1 \text{ , subject to: } \|\boldsymbol{y} - \boldsymbol{\Phi\Psi s}\|_2 \leq \varepsilon$$

  - CS theory shows that this problem can be solved
    - Efficiently (convexity)
    - Leading to the exact solution, or anyway to very good approximation to the original, full size **x**
    - Many suitable solvers available, e.g., L1-magic, Subspace-Pursuit, NESTA

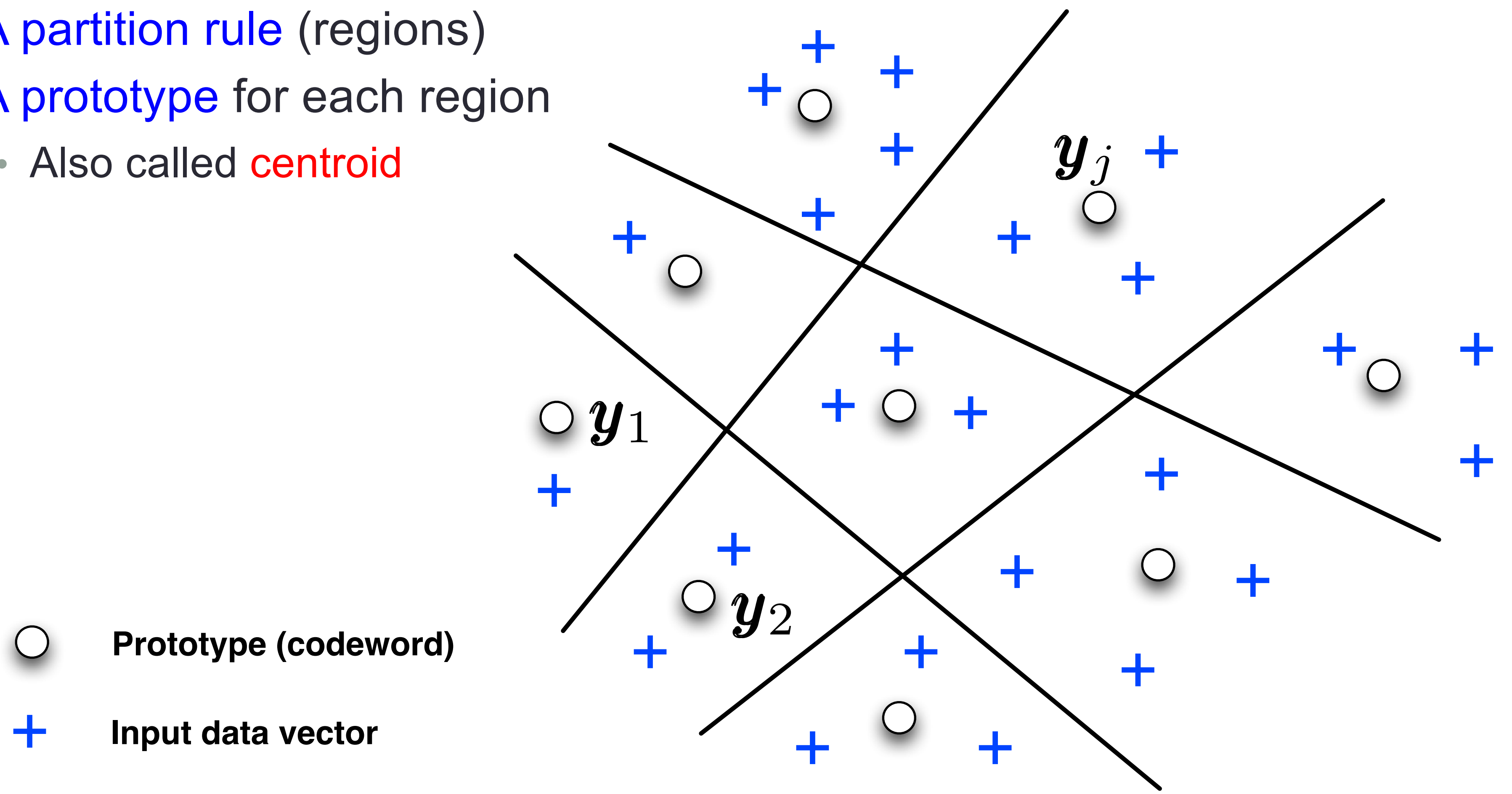
[Candes05] E. Candes and J. Romberg, "l1-MAGIC," Matlab routines for solving CS problems," [Online], 2005.

[Dai09] W. Dai and O. Milenkovic, "Subspace Pursuit for Compressive Sensing Signal Reconstruction," *IEEE Transactions on Information Theory*, 2009.

[Becker11] S. Becker, J. Bobin, E. J. Candes,"NESTA: A Fast and Accurate First-Order Method for Sparse Recovery," *SIAM Journal on Imaging Sciences*, 2011.

Vector Quantization (1/2)

- Given an input data distribution (multi-dimensional vectors)
- We need to find:
  - A partition rule (regions)
  - A prototype for each region
    - Also called centroid



Vector Quantization (2/2)

- Distortion measure
  - Most common measure: Euclidean distance
  - Average distortion is quantified as the mean square error

$$E[d(\boldsymbol{x}, \boldsymbol{y}_j)] = \sum_{j=1}^{\ell} \int_{I_j} \|\boldsymbol{a} - \boldsymbol{y}_j\| f(\boldsymbol{a}) d\boldsymbol{a} \quad (1)$$

Centroid

Input signal pdf

Region j

$\|\boldsymbol{x}\| \rightarrow \text{norm-2}$

- Optimal VQ
  - Find a set of centroids and a partition rule that minimize (1)
  - Nearest neighbor condition: the optimal partition is the one returning the minimum distortion (region j)

$$I_j = \{\boldsymbol{x} : d(\boldsymbol{x}, \boldsymbol{y}_j) \leq d(\boldsymbol{x}, \boldsymbol{y}_h), j \neq h\}$$

ECG segments

- First of all, ECG trace is segmented (from peak to peak)
- All segments are normalized to the same, fixed length m

$$\boldsymbol{x}_n = \left[x_{n1} \; x_{n2} \; \ldots \; x_{nm} \right]^T$$

- Offset:

$$e_{\boldsymbol{x}}(n) \triangleq \frac{\sum_{k=1}^{m} x_{nk}}{m}$$

- Gain:

$$g_{\boldsymbol{x}}(n) \triangleq \left(\frac{\sum_{k=1}^{m} x_{nk}^2}{m} \right)^{1/2}$$

- Normalization:

$$x_{nk} \leftarrow \frac{x_{nk} - e_{\boldsymbol{x}}(n)}{g_{\boldsymbol{x}}(n)}, \; k = 1, 2, \ldots, m$$

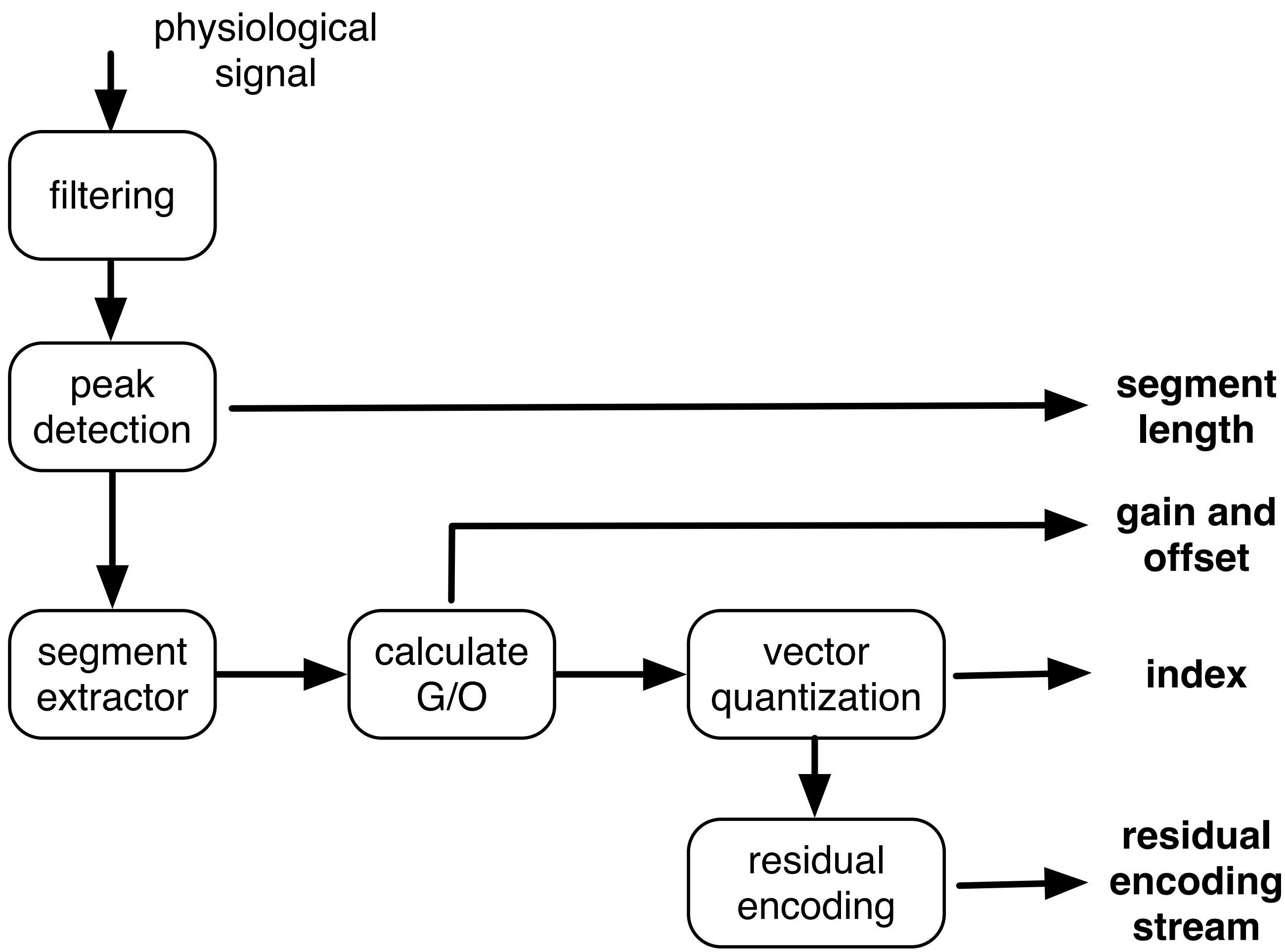
Gain Shape Vector Quantization (GSVQ) [ST05]

- Input time series is segmented (from peak to peak)
- Each segment is normalized (length, gain, offset)
- Codewords are found through through LBG [LBG80]
- Codebook is nearly-optimal but generated offline
  - Problem in non-stationary environments
  - A stream of residuals is sent to compensate for this

[ST05] C. C. Sun and S. C. Tai, “Beat-based ECG compression using gain-shape vector quantization,” *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 11, pp. 1882–1888, Nov. 2005.

[LBG80] Y. Linde, A. Buzo, and R. Gray, “An Algorithm for Vector Quantizer Design,” *IEEE Transactions on Communications*, vol. 28, no. 1, pp. 84–95, Jan. 1980.

GSVQ

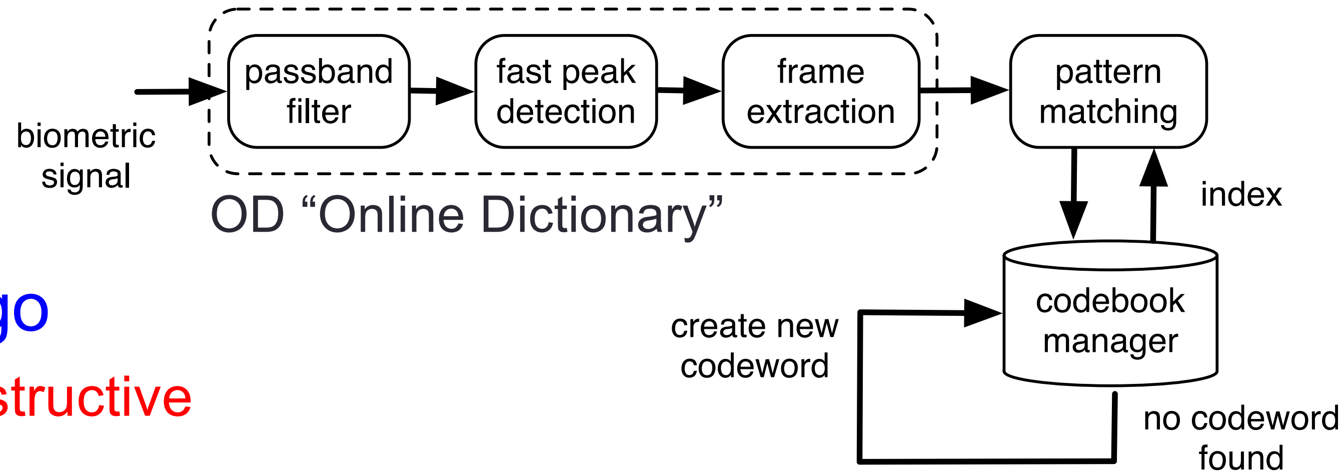


Peak detection

[AT83] M. L. Ahlstrom, W. J. Tompkins, “Automated High-Speed Analysis of Holter Tapes with Microcomputers,” *IEEE Transactions on Biomedical Engineering*, vol. 30, no. 10, pp. 651–657, Oct. 1983.

[E13] M. Elgendi, “Fast QRS Detection with an Optimized Knowledge-Based Method: Evaluation on 11 Standard ECG Databases,” *PLoS ONE*, vol. 8, no. 9, pp. 1–18, Sep. 2013.

OD



- Simple online algo
  - Impractical but instructive
- Idea
  - ECG segments are matched against a local dictionary at TX
  - Matching between segment and codeword occurs if
    - Their norm-2 distance remains within error tolerance ε
- Codeword management
  - **If matching occurs:** send codeword index
  - **If no matching:** add current segment to dictionary (new codeword)
- Observations
  - When a new codeword is added, it is also TX to the decompressor
  - Codewords are never removed from dictionary (memory problem)

Performance metrics (1/3)

E1 - Energy associated with compression

- Number of operations (e.g., divisions, additions, comparisons)
- Translate them into the corresponding number of *clock cycles*
- From clock cycles → energy consumption
- MCU: ARM Cortex M4

E2 - Energy associated with transmission / reception

- Consider the compressed data stream
- Compute the energy consumption associated with TX / RX
- Radio: Texas Instruments CC2541 (Bluetooth SoC)

⇒ total energy E1+E2

Performance metrics (2/3)

Representation accuracy

- Root Mean Square Error (RMSE)
- Expressed as % of the p2p signal's amplitude

Compression efficiency

$$CE = \frac{\#\text{bits in the original stream}}{\#\text{bits in the compressed stream}}$$

Performance metrics (3/3)

Representation accuracy

- Root Mean Square Error (RMSE)
- Expressed as % of the p2p signal's amplitude

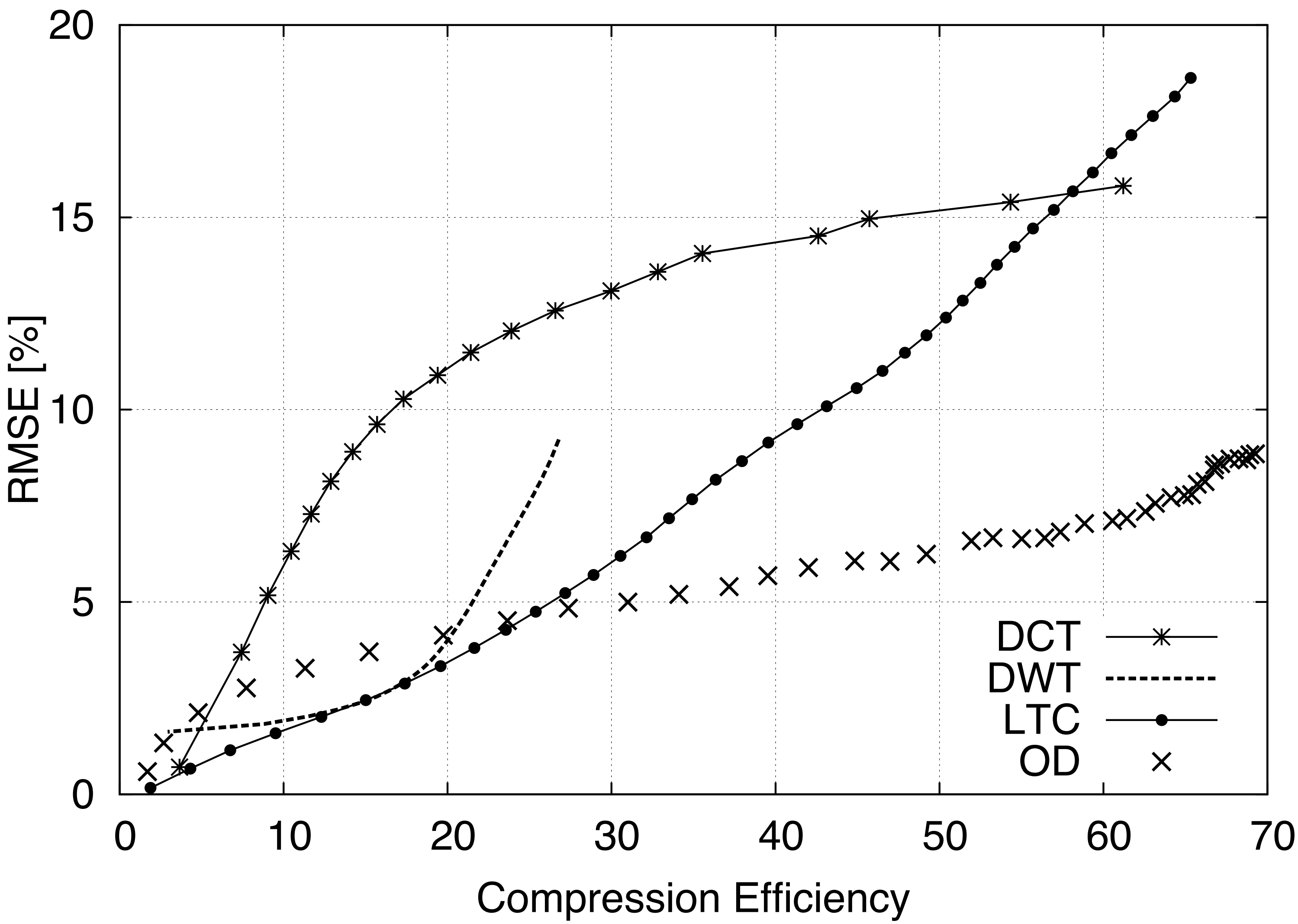
$$\mathrm{RMSE} = \frac{100}{\mathrm{p2p}} \left(\frac{\sum_{k=1}^{K} (x_k - \hat{x}_k)^2}{K} \right)^{1/2}$$

$$\begin{cases} K & \text{total no. of samples in ECG trace} \\ x_k & \text{ECG sample at time } k \\ \hat{x}_k & \text{compressed ECG sample at time } k \end{cases}$$

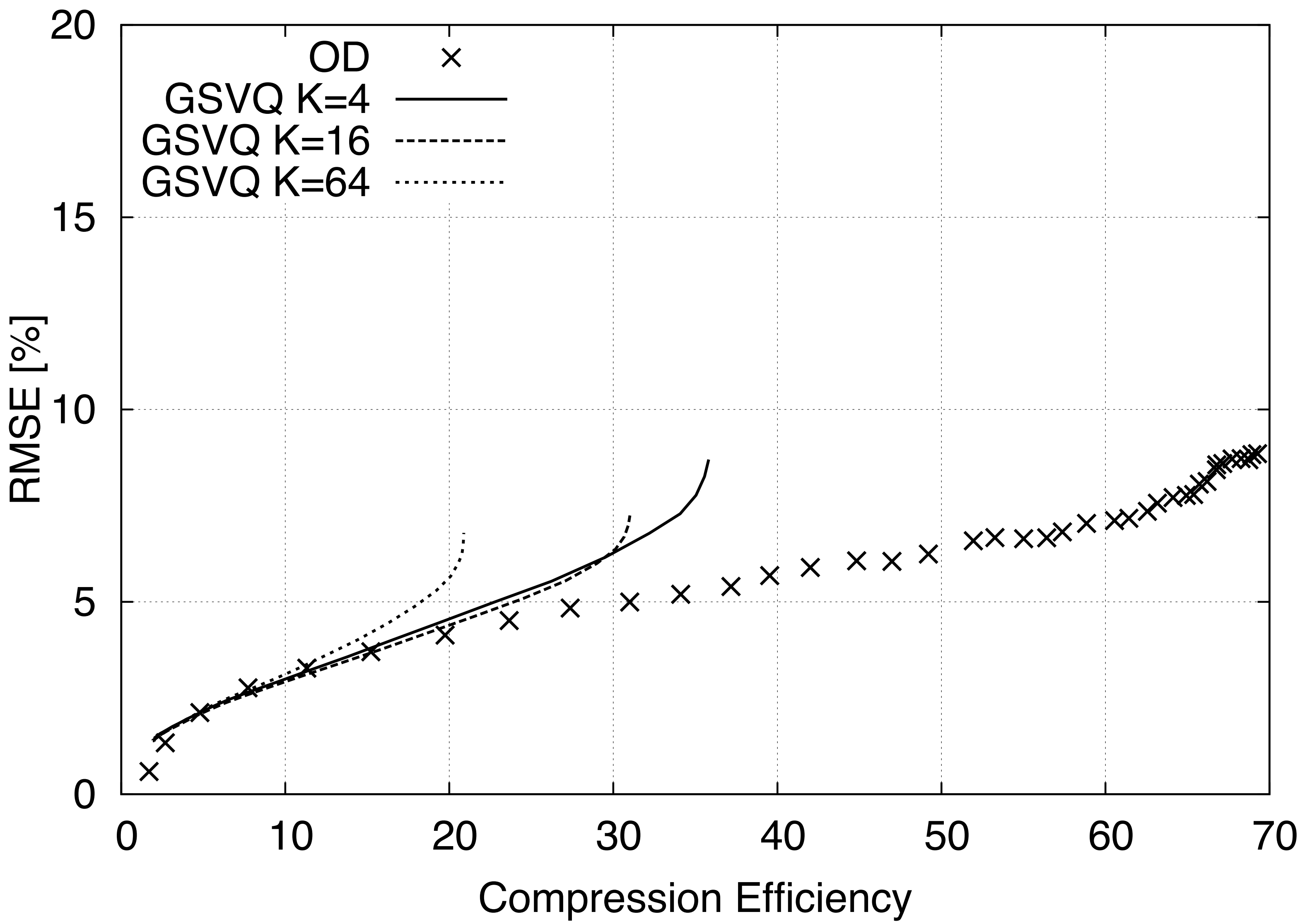
Results RMSE (1/3)

OD "online dictionary"

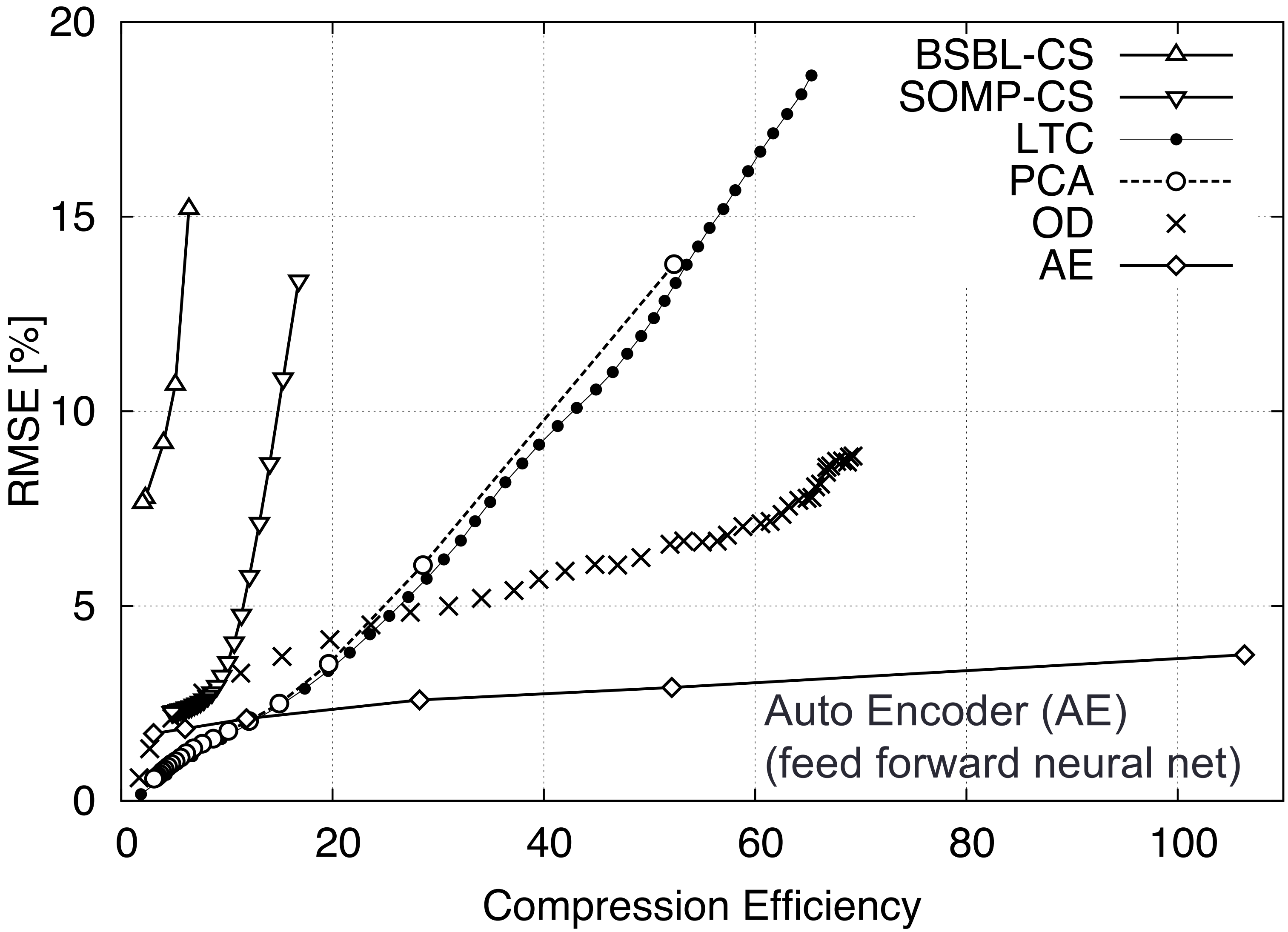
- heuristic & memory inefficient
- But its RMSE looks promising



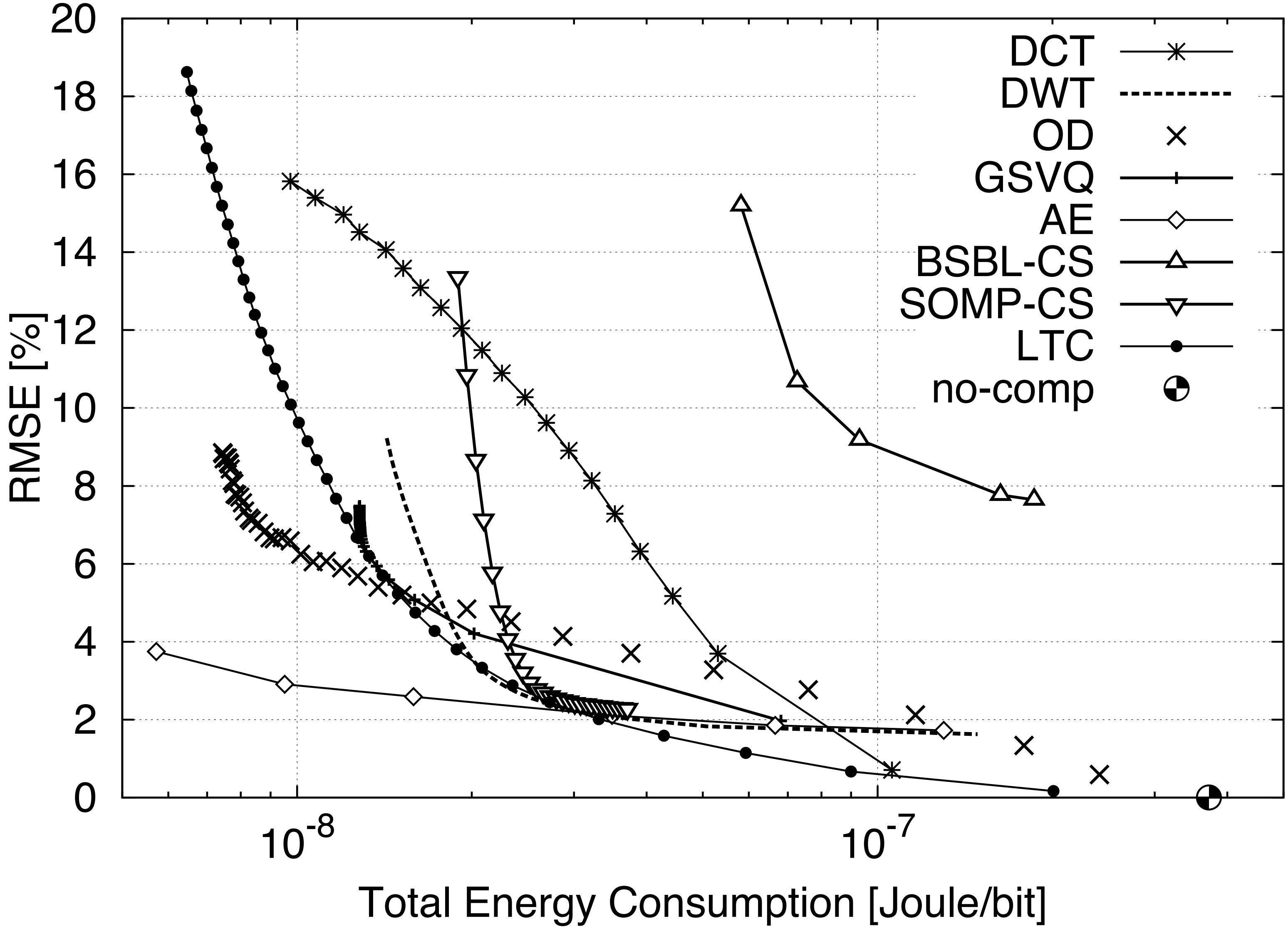
Results: RMSE *vs* CE (2/3)



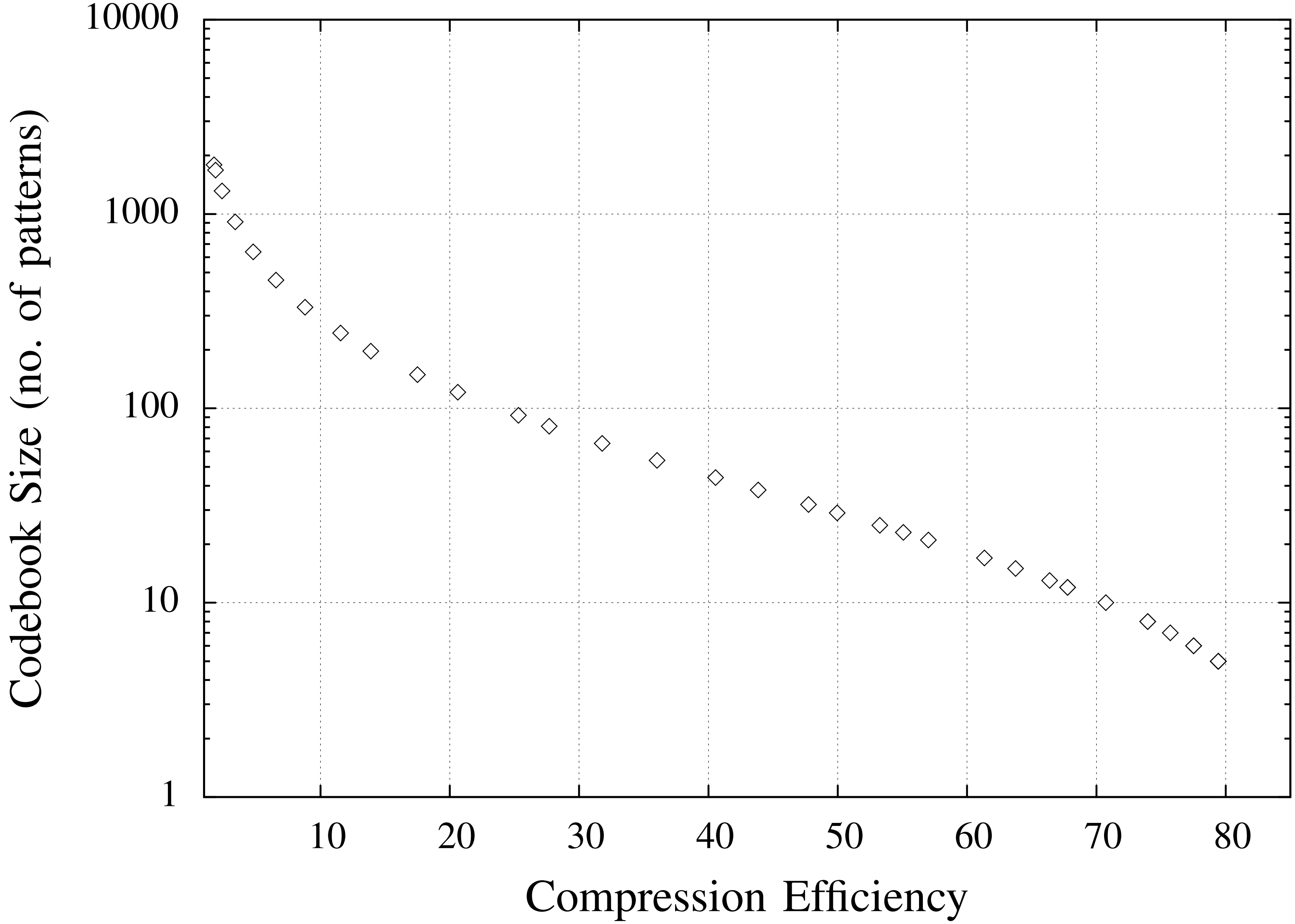
Results: RMSE *vs* CE (3/3)



Results: RMSE *vs* Energy



OD: memory use



SUBJECT ADAPTIVE COMPRESSION

Design principles

- We want the codebook (dictionary) to be
  - Compact (memory efficiency)
  - Built and maintained at runtime (adaptivity)
  - Provide accurate representations (accuracy)
  - Subject-adaptive (person centric)

- To that end we exploit neural network architectures
  - Self-Organizing Maps (SOM)
  - Time Adaptive Self-Organizing Maps (TASOM)
  - Growing Neural Gas (GNG) networks
  - Solve the VQ problem via training examples
  - Allow for compact representations (i.e., small # of prototypes)

Sparse coding (unsupervised)

- Learns set of bases to represent data efficiently $\{\boldsymbol{\phi}_i\}_{i=1}^{k}$
- Input vector $\boldsymbol{x} \in \mathbb{R}^m$
- Is represented as $\boldsymbol{x} = \sum_{i=1}^{k} a_i \boldsymbol{\phi}_i$
- Sparsity: a few coefficients are non-zero for each input vector
- Overcompleteness: k>m (better able to capture structure)
- Observation: sparsity is a highly non-linear model, as the dictionary can change from signal to signal

What we want

- Each ECG segment is a vector $\boldsymbol{x} \in \mathbb{R}^m$
- We aim at a dimensionality reduction technique where
  - Input vector is a linear combination of basis vectors
  - Only one coefficient is non-zero
  - Number of vectors in the basis k<<m
- This can be achieved through vector quantization
- But we additionally want it to be lightweight and online

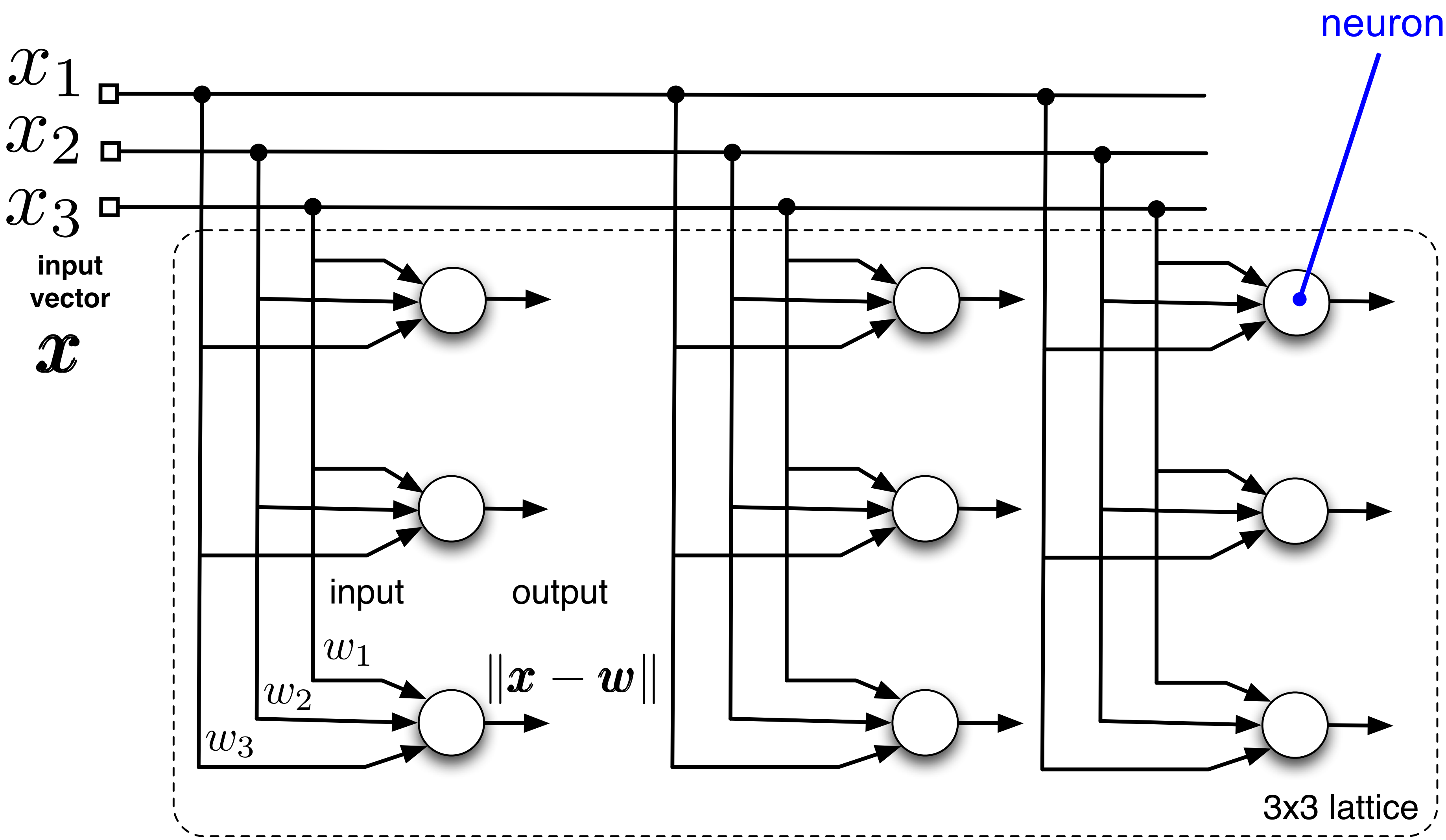
Self Organizing Maps [K82] (1/3)

- Uses unsupervised learning (unlabeled examples) to
  - Produce compact representations of the input space
  - These are encoded onto a MAP of weights
  - A vector of weights is assigned to each neuron in the map

- As learning evolves
  - Weights resemble more and more the input distribution
  - Each weight vector becomes a prototype for the input space

[K82] Teuvo Kohonen, "Self-Organized Formation of Topologically Correct Feature Maps," *Biological Cybernetics*, Vol. 43, No. 1, pp. 59-69, Jan. 1982.

Self Organizing Maps (2/3)



Self Organizing Maps (3/3)

- Competition
  - For each new training vector (sequence): $\boldsymbol{x}(n)$
  - The neurons compete among themselves
  - A winning neuron (fittest) is selected as

$$i(\boldsymbol{x}) = \underset{j}{\operatorname{argmin}} \left\| \boldsymbol{x} - \boldsymbol{w}_j(n) \right\|, \; j \text{ is a neuron of the MAP}$$

- Cooperation
  - The winning neuron identifies a neighborhood (function $h_{ij}(n)$)
  - The weights of the neurons in this neighborhood are also updated

- Synpatic-weight adaptation (*stochastic approx. like*)

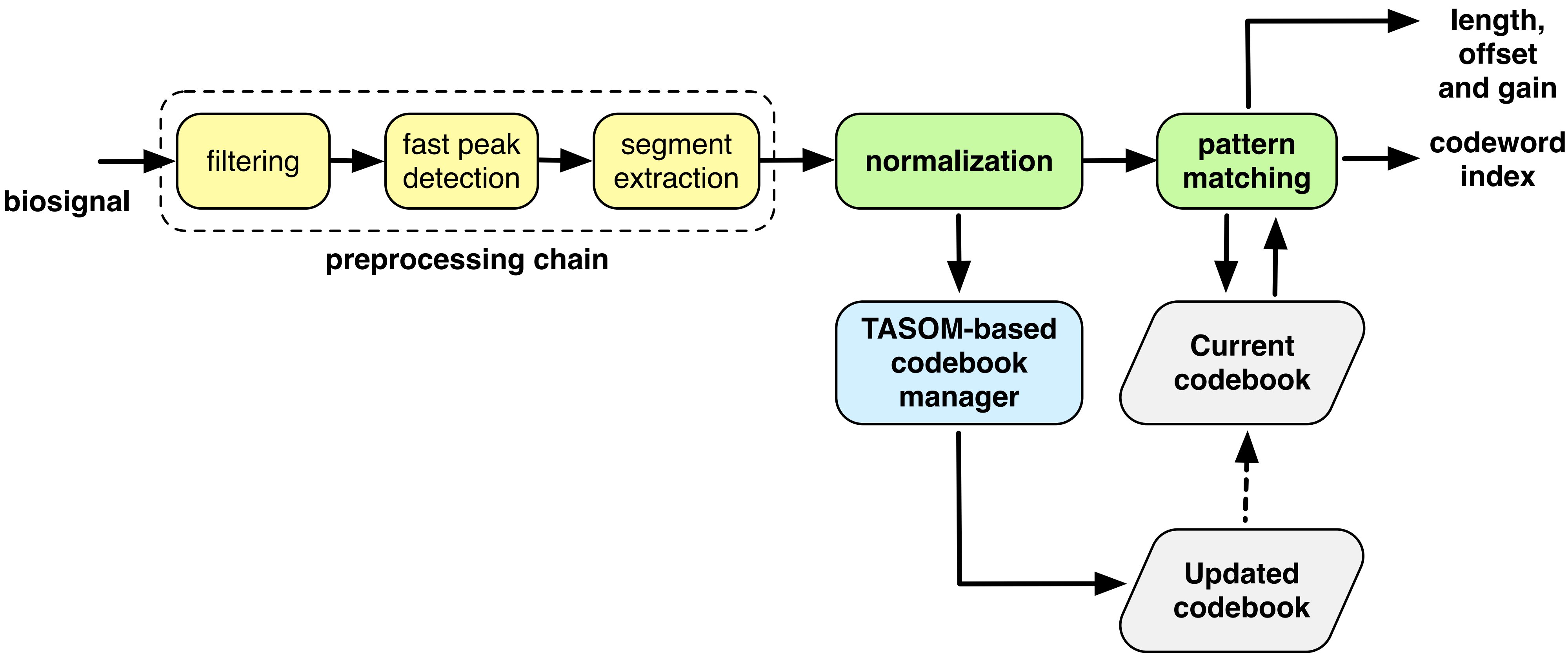
$$\boldsymbol{w}_j(n+1) = \boldsymbol{w}_j(n) + \eta(n) h_{ij}(n)(\boldsymbol{x}(n) - \boldsymbol{w}_j(n))$$

TASOM [HS00]

- The problem of SOMs is that
  - Once training is complete the MAP is no longer updated
  - This is because *the learning rate parameter shrinks to zero*
- TASOM solves the problem by
  - Allowing the rate parameter to also increase
  - The SOM map keeps on learning as the input distribution evolves

[HS00] H. Shah-Hosseini and R. Safabakhsh, "TASOM: the time adaptive self-organizing map," in *Proceedings of the International Conference on Information Technology: Coding and Computing*, Las Vegas, NV, US, Mar. 2000, pp. 422–427.

SAM Diagram



biosignal
filtering
fast peak detection
segment extraction
preprocessing chain
normalization
pattern matching
length, offset and gain
codeword index
TASOM-based codebook manager
Current codebook
Updated codebook

Codebook Manager

- Maintains two codebooks C1 and C2
- C1 updated at each new acquisition
  - Elect winning neuron for the current input
  - Adapt synaptic weight of winning neuron and neighborhood (TASOM)
  - C1 tracks time varying statistics $i(\boldsymbol{x})$
- C2 is updated only when

$$d(n)/\|x(\boldsymbol{n})\| > \varepsilon$$

where: $d(n) = \|\boldsymbol{x}(n) - w_{i(\boldsymbol{x})}\|$

- If this occurs
  - Replace C2 with C1
  - Send the new dictionary to the decompressor (receiver)

TASOM-based compression algorithm

Algorithm 1 TASOM-Based Compressor

1) Map $\boldsymbol{x}(n)$ onto the index of the best matching codeword in $\mathcal{C}^c(n)$, i.e., map $\boldsymbol{x}(n)$ onto the index $i_{\boldsymbol{x}}(n)$ such that

$$i_{\boldsymbol{x}}(n) = \arg\min_j \|\boldsymbol{x}(n) - \boldsymbol{c}_j^c(n)\|, \quad j = 1, \ldots, L. \quad (9)$$

Use SOM dictionary to get best matching codeword

2) Let $d(n) = \|\boldsymbol{x}(n) - \boldsymbol{c}_i^c(n)\|$ be the distance between the current segment and the associated codeword, where we use index i as a shorthand notation for $i_{\boldsymbol{x}}(n)$. Use $\boldsymbol{x}(n)$ as the new input for the current iteration of the TASOM learning algorithm and obtain the new synaptic-weight vectors $\boldsymbol{w}_j(n)$, $j = 1, \ldots, L$.

3) Update $\mathcal{C}^u(n)$ by using the weights obtained in step 2, i.e., setting $\boldsymbol{c}_j^u(n) \leftarrow \boldsymbol{w}_j(n)$ for $j = 1, \ldots, L$.

Update SOM dictionary with new ECG segment

4) Let $\varepsilon > 0$ be a tuning parameter. If $d(n)/\|\boldsymbol{x}(n)\| > \varepsilon$, then update $\mathcal{C}^c(n)$ by replacing it with $\mathcal{C}^u(n)$, i.e., $\mathcal{C}^c(n) \leftarrow \mathcal{C}^u(n)$ and, using (9), re-map $\boldsymbol{x}(n)$ onto the index $i_{\boldsymbol{x}}(n)$ of the best matching codeword in the new dictionary $\mathcal{C}^c(n)$.

If d(n) larger than threshold, adopt new dictionary and use it in place of current dictionary

5) Send to the receiver the segment's original length $r_{\boldsymbol{x}}(n)$, its offset $e_{\boldsymbol{x}}(n)$, gain $g_{\boldsymbol{x}}(n)$, and the codeword index $i_{\boldsymbol{x}}(n)$. If $\mathcal{C}^c(n)$ has been modified in step 4, then also send $\mathcal{C}^u(n)$ (that in this case is equal to the new $\mathcal{C}^c(n)$).

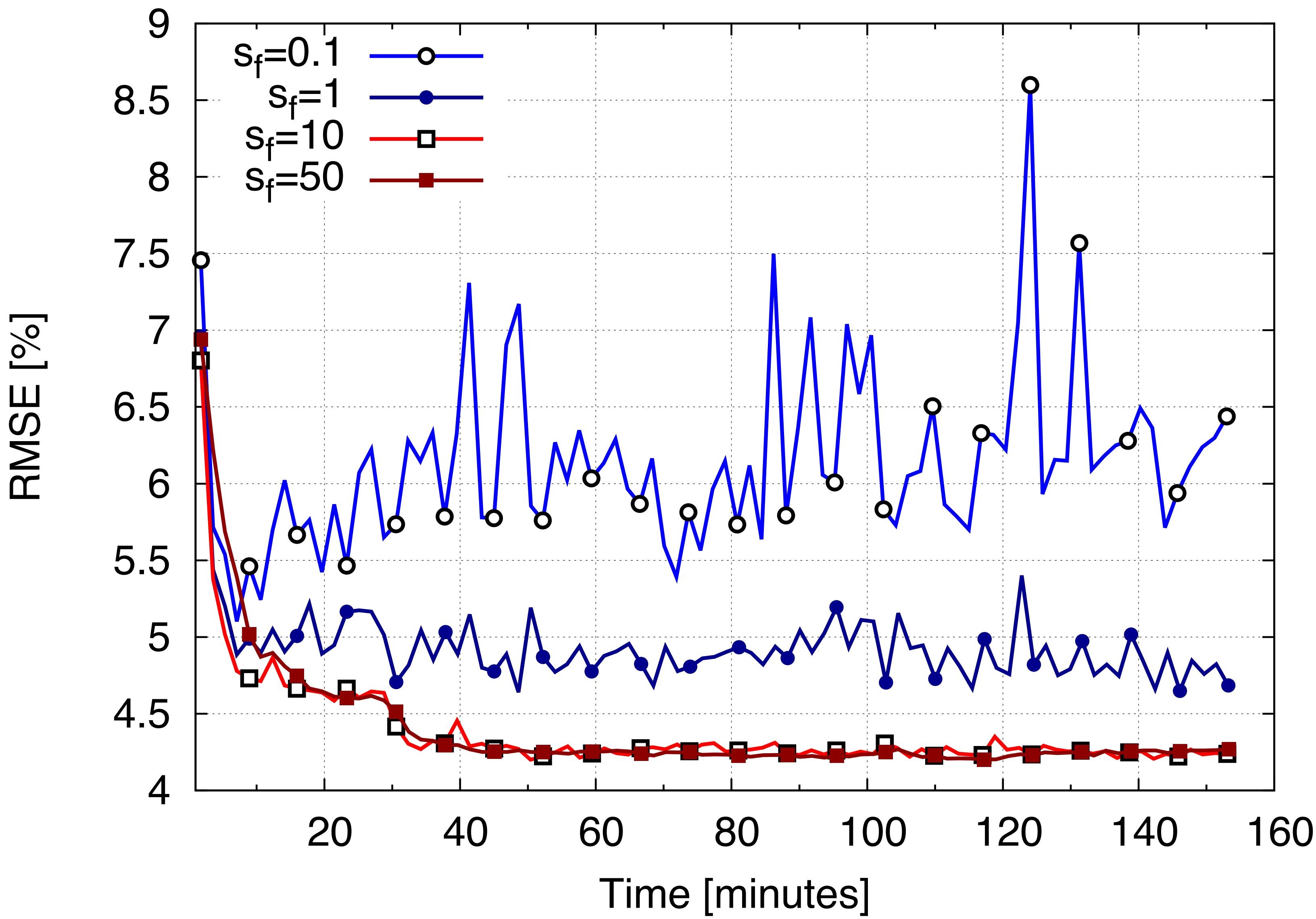
Send codeword index to the receiver in place of full segment

Decompressor at receiver side

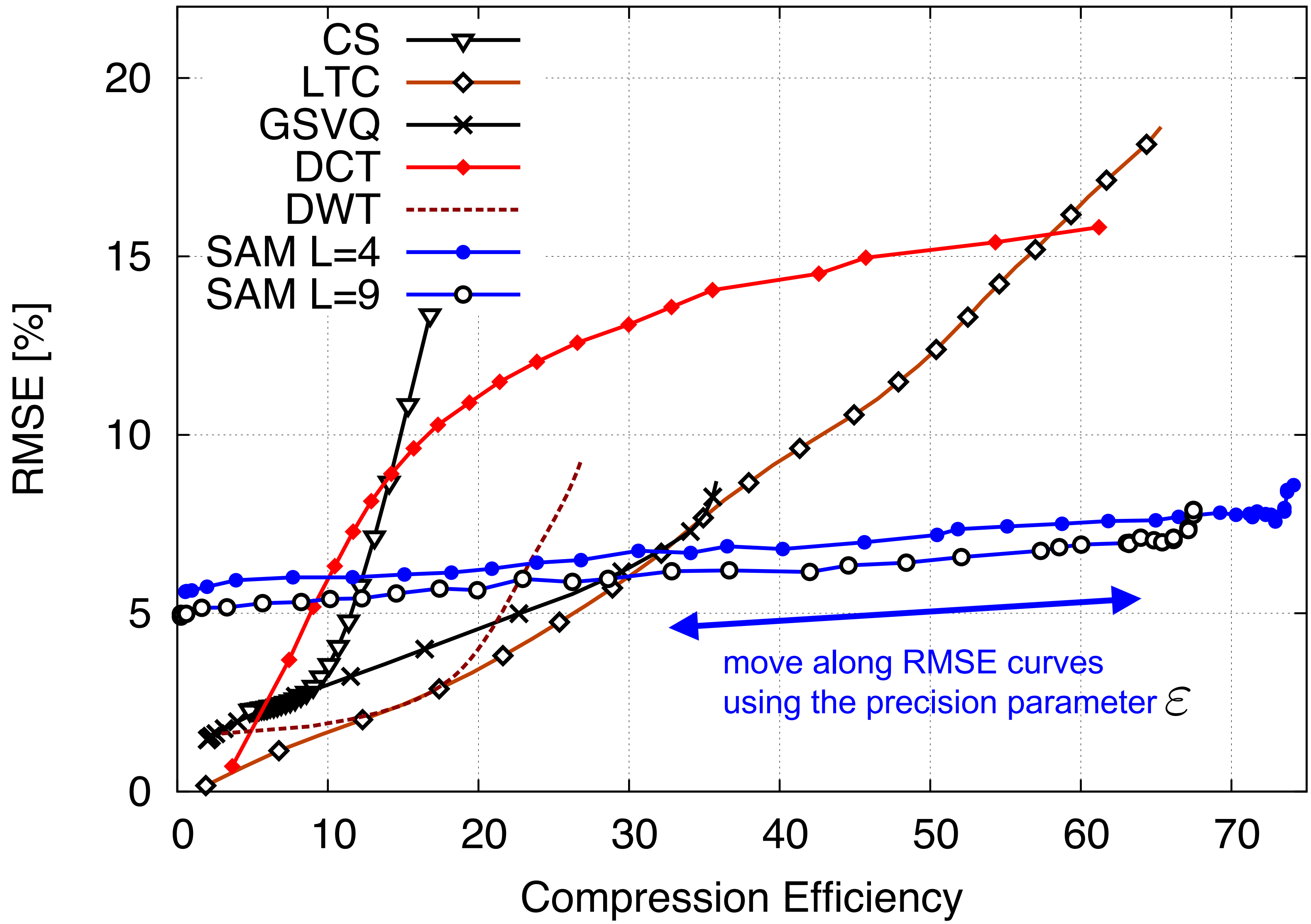
- ECG segment reconstruction at decompressor
  - Pick codeword with RXed index from the local dictionary
  - Perform renormalization of such codeword with respect to
    - offset
    - gain
  - Stretch the codeword according to the actual segment length
    - The ECG segment duration in the dictionary is fixed: it depends on sampling rate of measurement device and number of samples per codeword (m)
    - Example: rate = 250 samples/s → Δ=4 ms, m = 280 samples/codeword
    - With these pars, the codeword duration is DC = 280 * 4 ms = 1.12 s
    - If the duration of the current segment is DS = 1.3 s, stretching implies multiplying the inter-sample time Δ of the selected codeword:

$$4\text{ms} \times (\text{DS}/\text{DC}) = 4\text{ms} \times (1.3/1.12) = 4.643 \text{ ms}$$

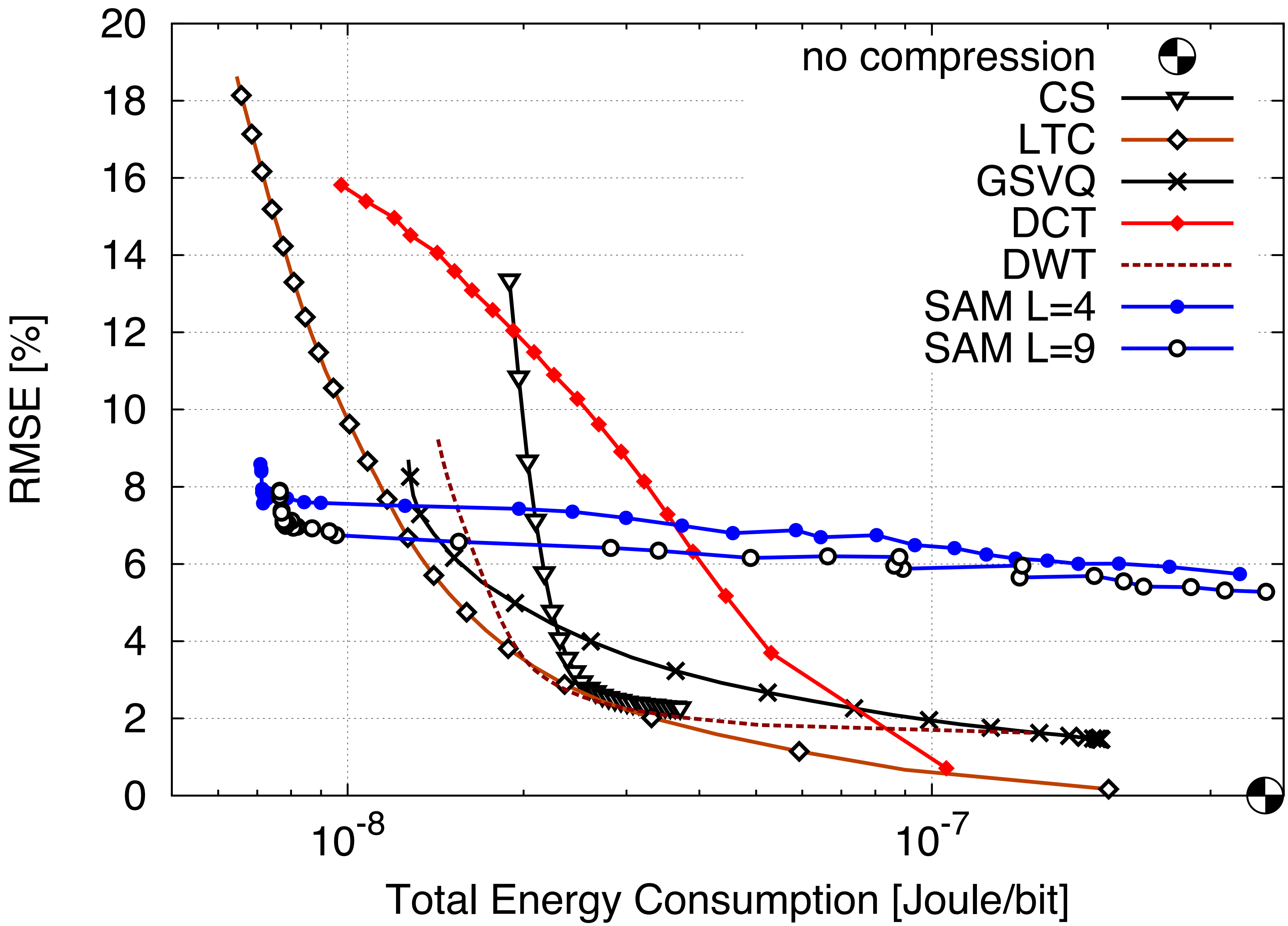
Learning & Generalization



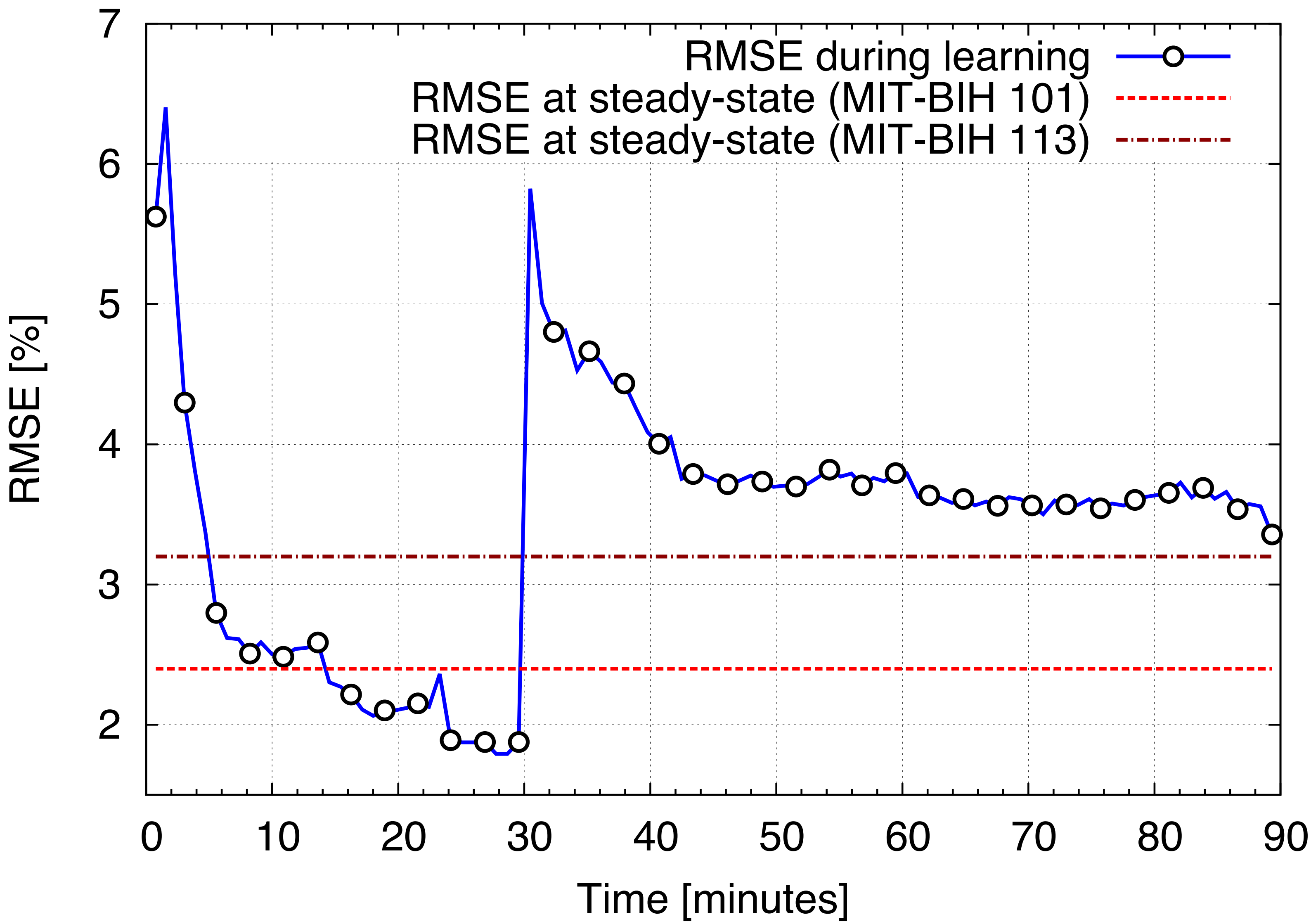
Compression Efficiency



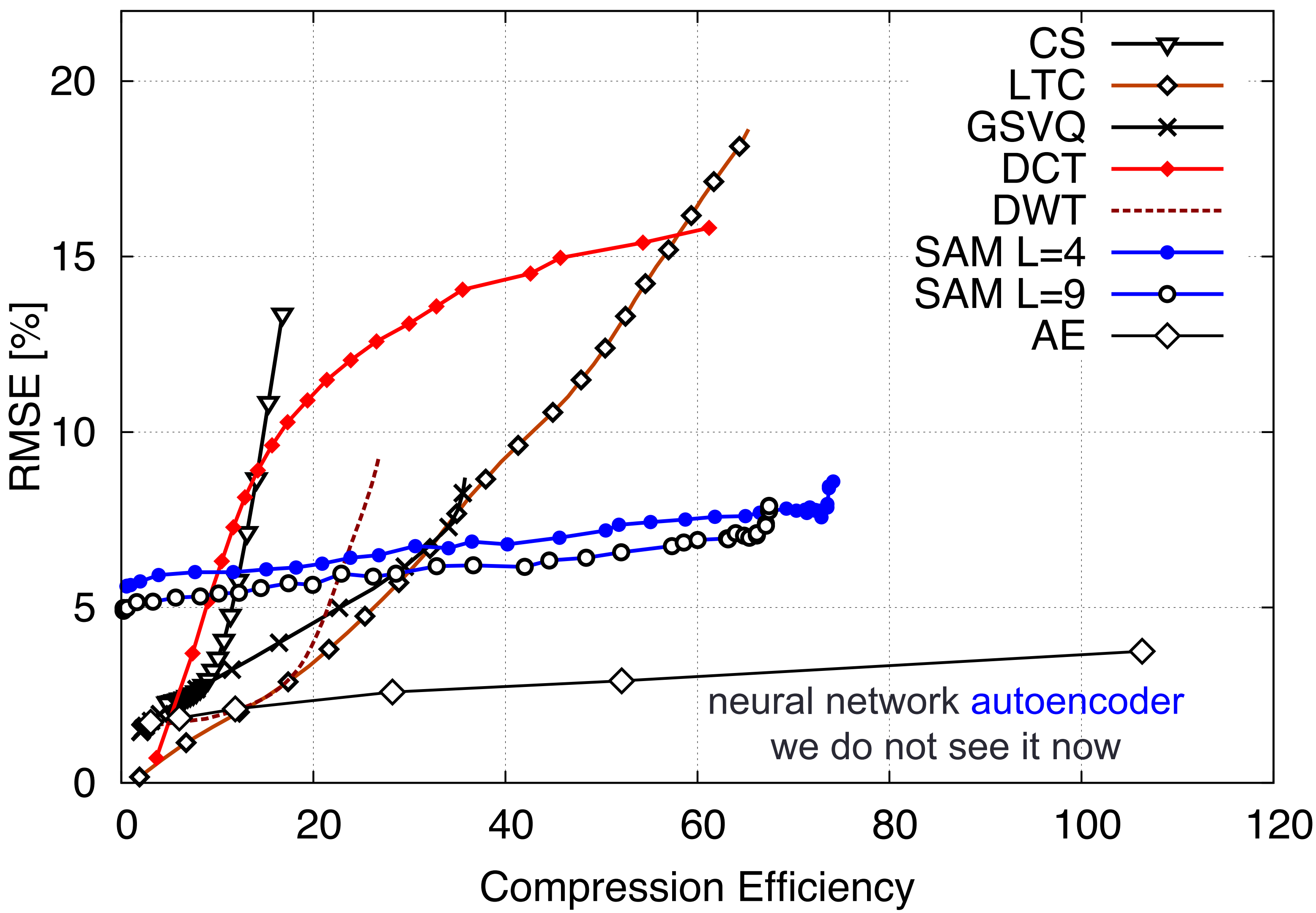
Energy Consumption



Learning in Non-Steady Environments

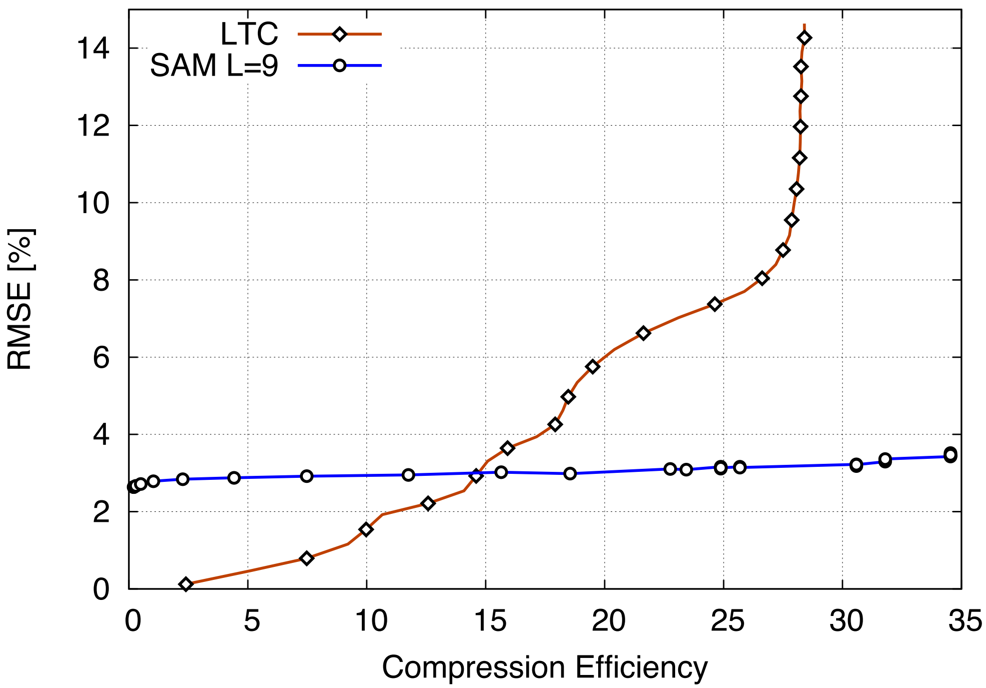


Compression Efficiency (all algos)

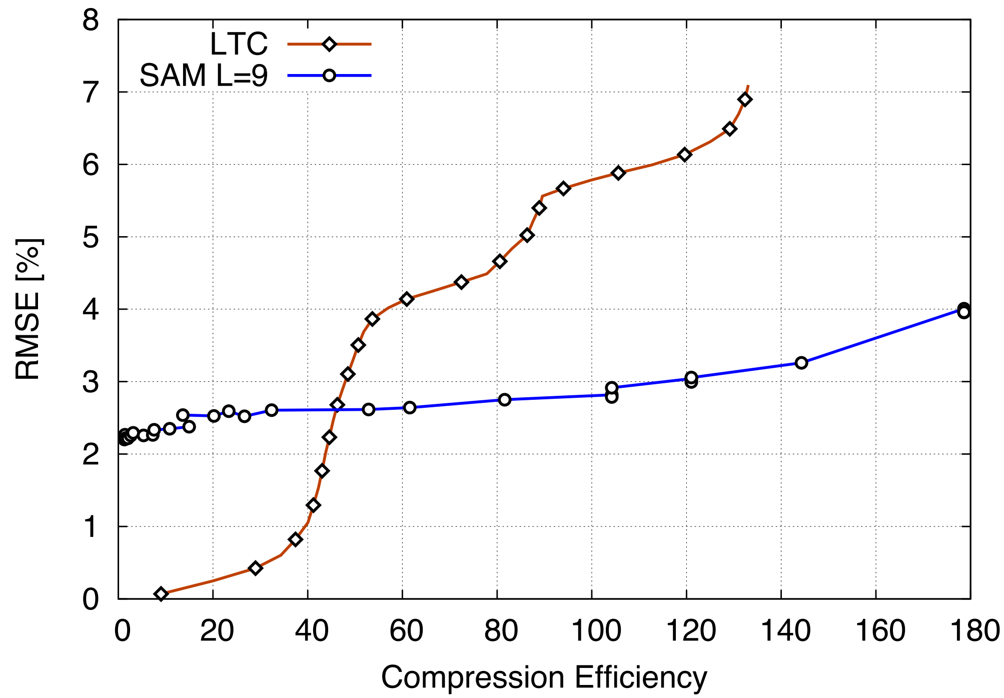


PPG and RESP

PPG



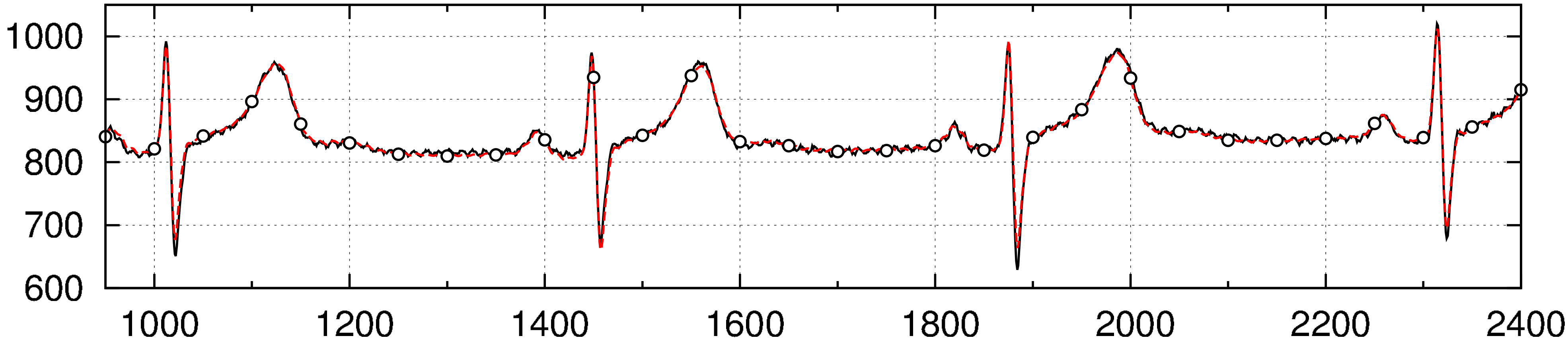
RESP



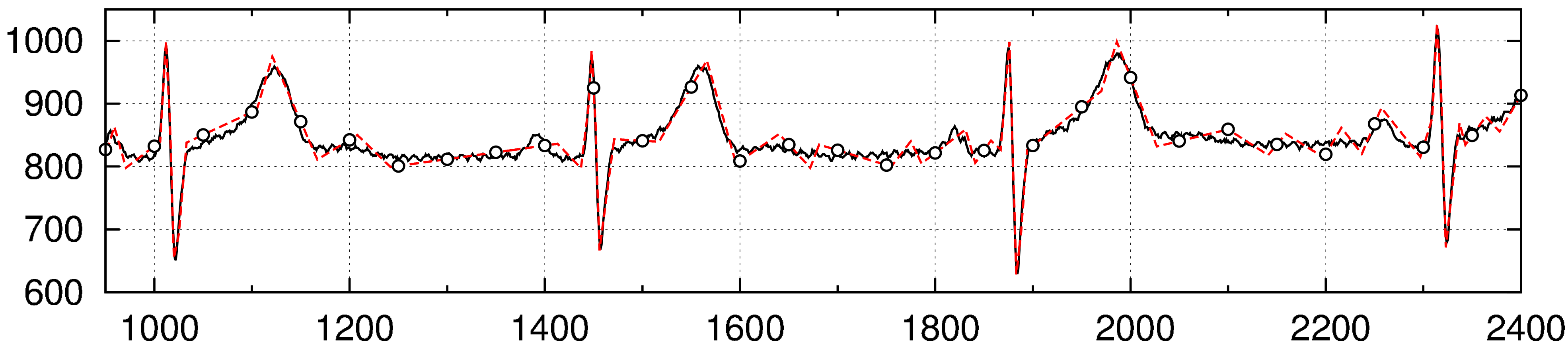
Temporal Compression Example

no artifacts

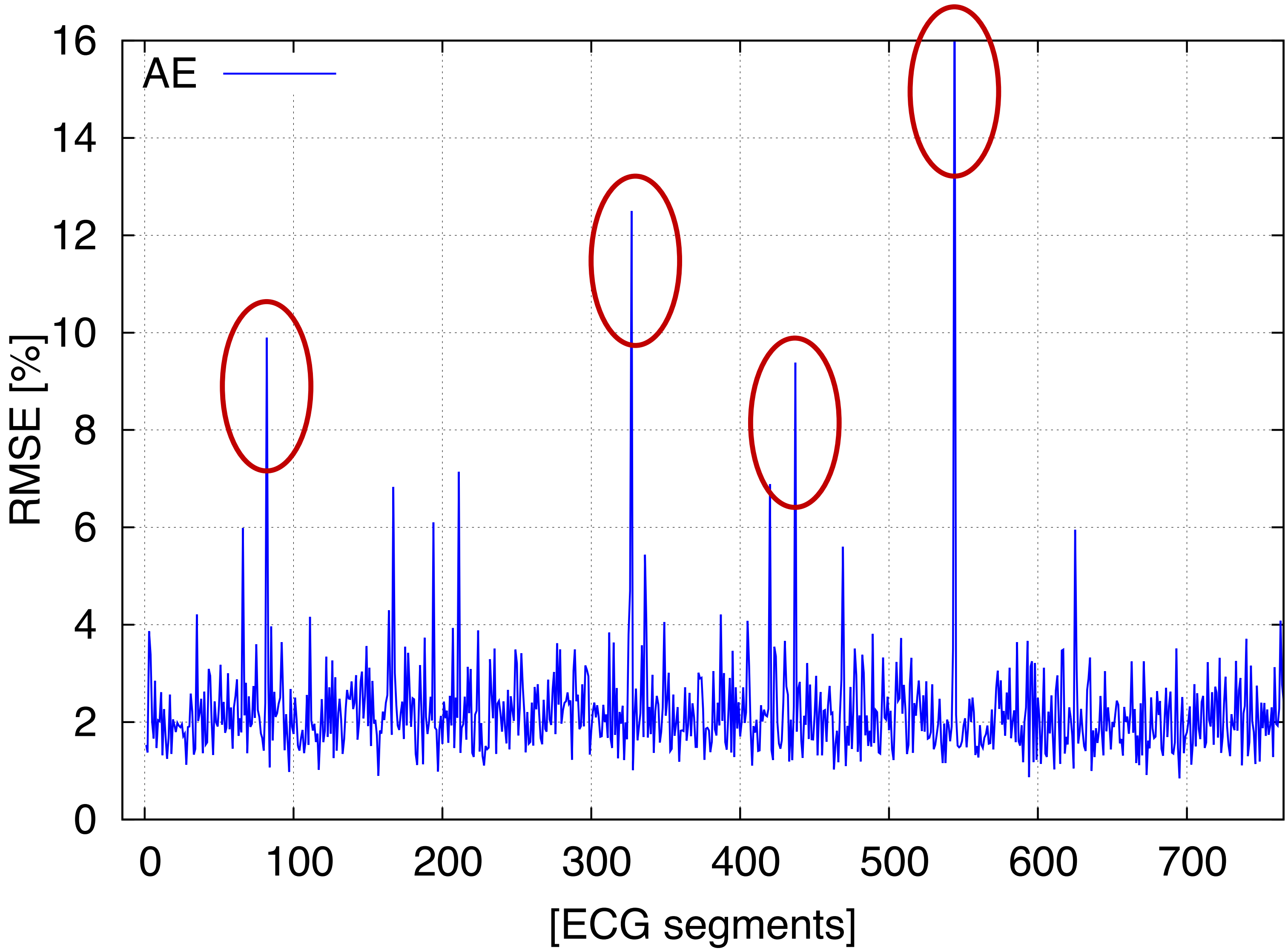
SAM: RMSE 6.71%, Compression efficiency is 61



LTC: RMSE 6.71%, Compression efficiency is 31



Artifacts: where TASOM (and AE) fails



Steps for an enhanced design - objectives

O1) specializing the dictionary to new signal areas

The number of neurons in the TASOM map remains fixed as time evolves and this entails that some *further refinement of the dictionary,* whenever the signal statistics undergoes major changes and new behaviors arise, may not always be possible.

At times, additional neurons may be beneficial to specialize the dictionary upon the occurrence of new patterns, while at the same time preserving what previously learned. In this case, we do not want previous neurons to be involved in the refinement as we are exploring a new area in the input signal space, and we do not want to do this at the cost of getting lower accuracies in the portion of space that we have already (and successfully) approximated.

Steps for an enhanced design - objectives

O2) reducing overhead (maintain and TX the dictionary)

This is achieved through two techniques:

1) Feature space: representing codewords working within a suitable feature space, where a number of features much smaller than the size m of each ECG segment suffices for its accurate representation

2) Selective dictionary update: in the TASOM-based approach, a dictionary is entirely replaced whenever any of its codewords is no longer capable of approximating ECG segments belonging to a given signal's area within a preset accuracy. Instead, in the new design codewords are selectively replaced by new ones that better approximate the portion of signal space that they are responsible for

Steps for an enhanced design - objectives

O3) coping with artifacts

Dictionary-based approaches are particularly sensitive to artifacts as no existing codeword can adequately approximate them. Dictionary updates, attempting to bring the codewords closer to the new segments (i.e., the artifacts) are likely to result in a degraded representation accuracy for the recurring segments. However, these noisy segments must be accurately represented, as these may indicate anomalous behavior that has *clinical relevance*. The new compressor: **(1)** sends features in place of full codewords whenever none of the current codewords provide a satisfactory match and **(2)** concurrently starts an assessment phase for the new pattern: there, a new neuron (codeword) is temporarily added to a local dictionary, which is only maintained at the source and is used for the evaluation of new (or anomalous) patterns. The addition of such codeword to the main dictionary only occurs if further segments are found to match it, which means that the new segment has become recurrent

SURF: an Enhanced Design

- A new design using Growing Neural Gas (GNG) networks [Fr94]
  - Neurons still form a map but
    - they can be added and removed (specialize dictionary to new signal's areas)
    - dictionary can be extended as new patterns emerge
- SURF improves upon the previous design by
  - Adding / removing neurons in the dictionary
  - Automatically detecting new / anomalous patterns
    - Add new neurons, train them and assess
    - Use standard compression techniques until new pattern is learned

[Fr94] Bernd Fritzke, "A Growing Neural Gas Network Learns Topologies," NIPS Proceedings of the 7th International Conference on Neural Information Processing Systems, 1994.

[Pal17] Esteban J. Palomo, Ezequiel López-Rubio, "The Growing Hierarchical Neural Gas Self-Organizing Network," IEEE Trans. on Neural Networks and Learning Systems, Vol. 28, No. 9, 2017.

SURF Dictionaries

- Current dictionary D1 (used for compression)
  - Main dictionary – must be the same at TX and RX
  - Only *recurrent* patterns are encoded in it
  - Sporadically updated (when error tolerance is no longer met)
- Reserved dictionary D2 (for assessment only)
  - Encodes new (*previously unseen*) patterns
  - Not used for compression
  - Codewords assessed against new patterns
  - If matches occur within a time window from their addition
    → codeword added to D1
- Updated dictionary D3
  - Like D1, but updated every time a match for a codeword in D1 is found
  - Used for selective replacement of codewords in D1

SURF – as a new segment is measured

IF no codeword in dictionary D1 accurately represents it

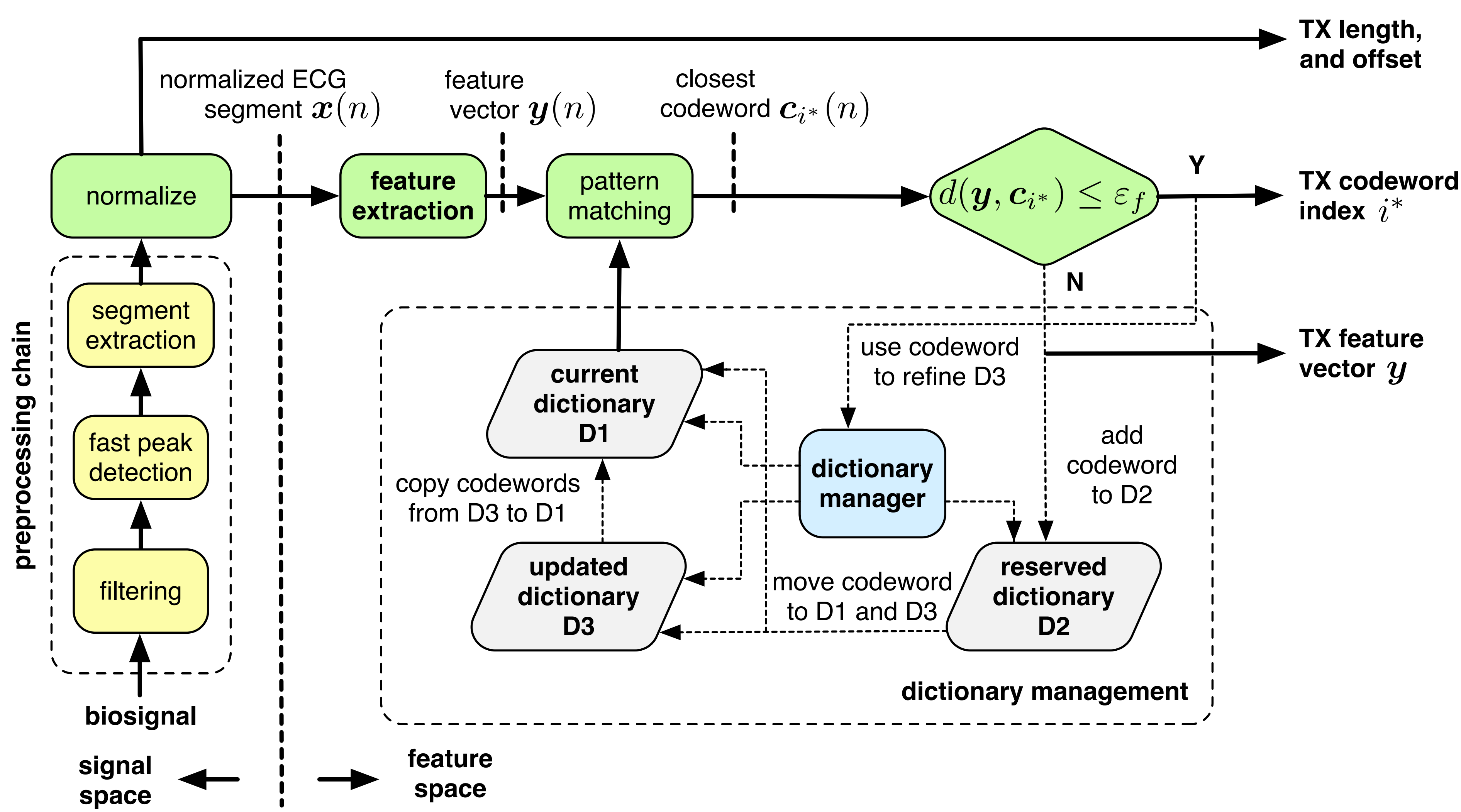
THEN

- Add codeword to D2
- Check for future matches with this codeword
- If multiple matches occur → move to D1 (and D3)
- Current ECG segment
  - compressed through standard methods (e.g., DCT-based)

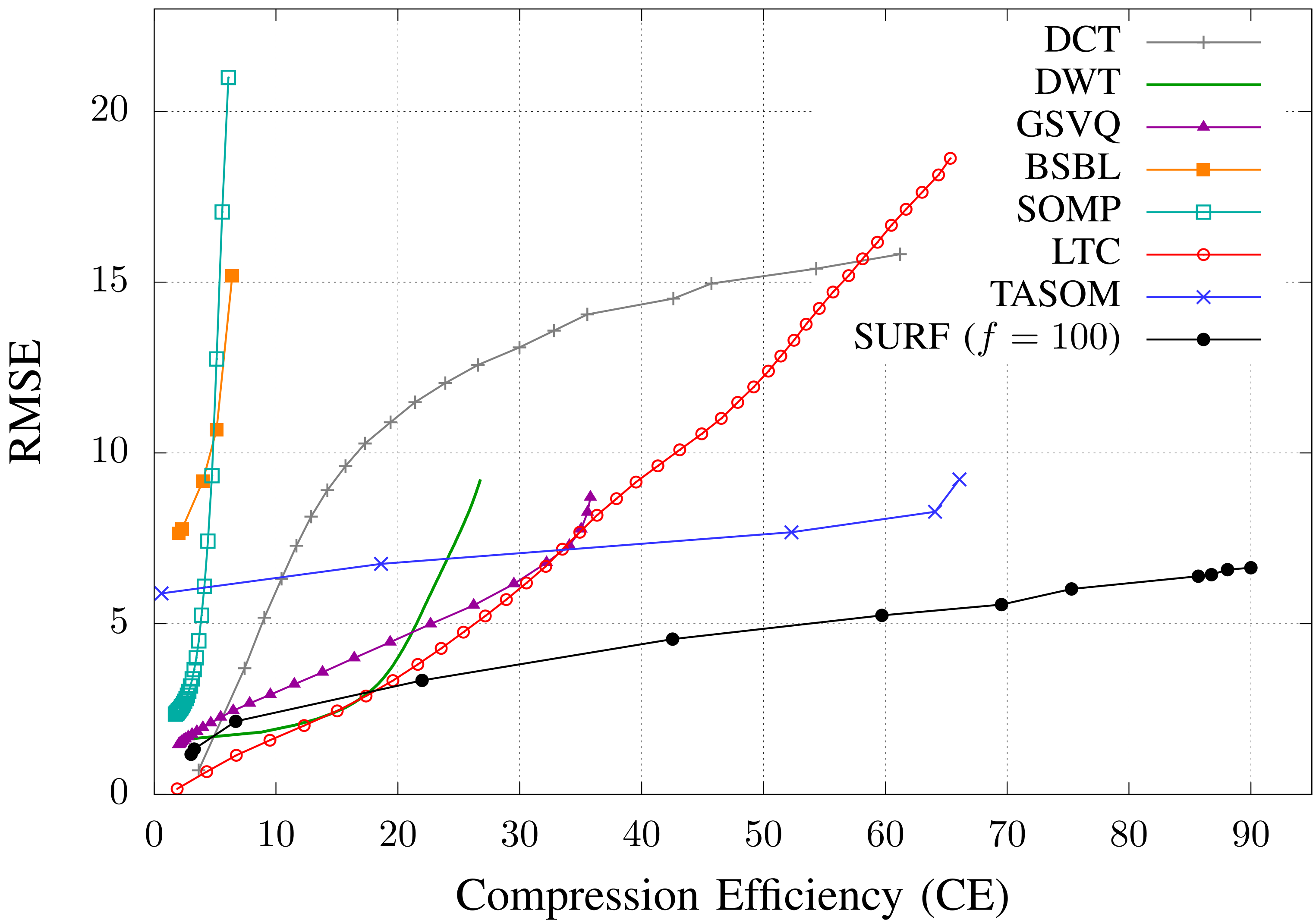
ELSE one codeword in D1 has distance within error tolerance

- Use it for compression at the TX - send codeword index

SURF Diagram



SURF: Compression Efficiency (MIT-BIH reference dataset)



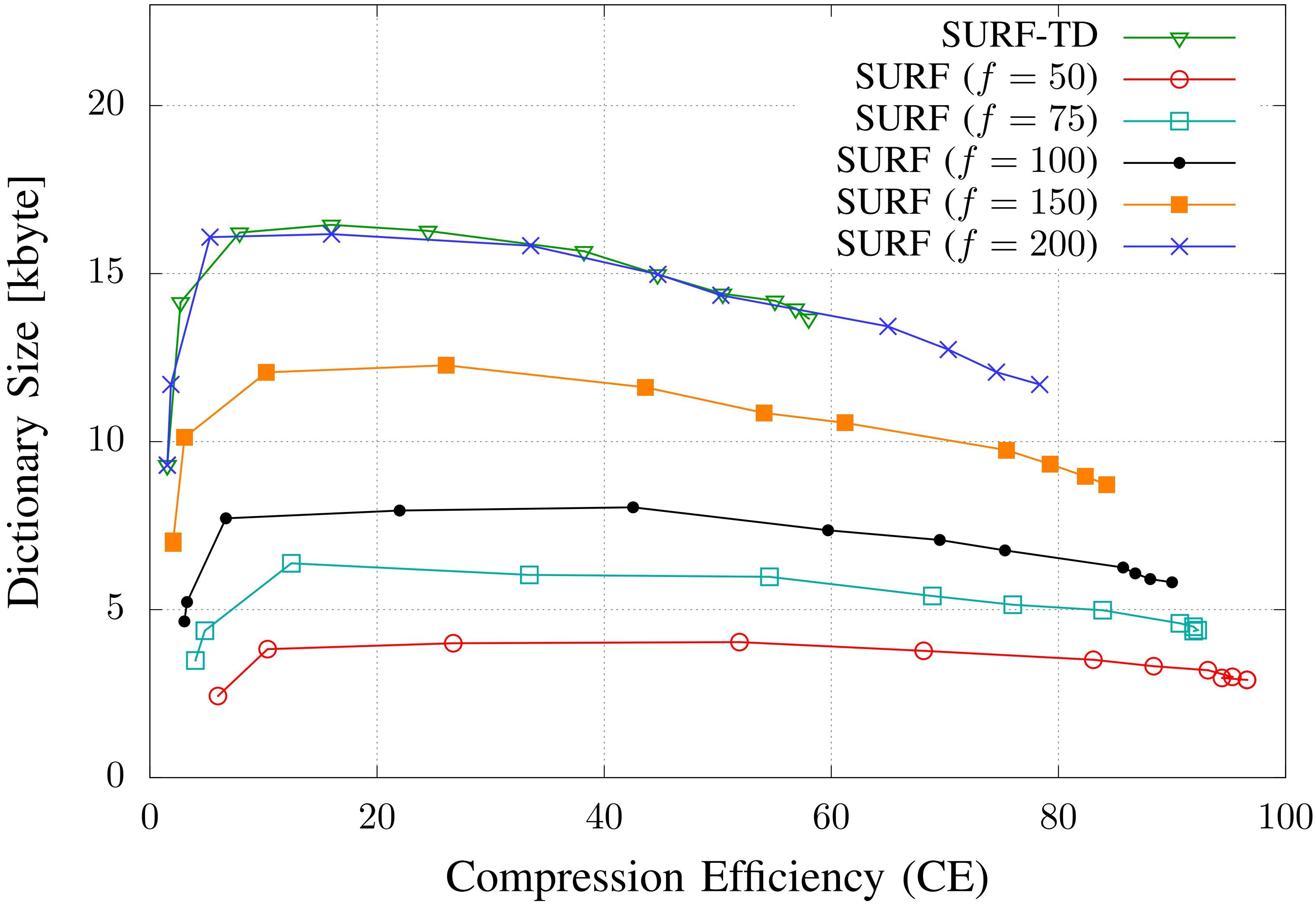
SURF vs features

- Two variants of the algorithms were implemented:

1) SURF-TD. It is a time domain implementation of SURF dictionaries, i.e., the feature vectors $\boldsymbol{y}(n)$ that are inputted into the dictionary correspond to the original signal segments, i.e., $\boldsymbol{y}(n) = \boldsymbol{x}(n)$. If an input segment is unmatched, the corresponding DCT coefficients are transmitted to the receiver. So, in this case the DCT transform is only applied if a new pattern that the current dictionary is unable to approximate is detected. In this case, f of its DCT coefficients are sent to reconstruct it at the receiver ($f = 200$ is used for the SURF-TD curve in Fig. 4).

2) SURF. It is the feature domain implementation that we have described in Section VI, for which we considered the following values for the feature space size $f \in \{50, 75, 100, 150, 200\}$

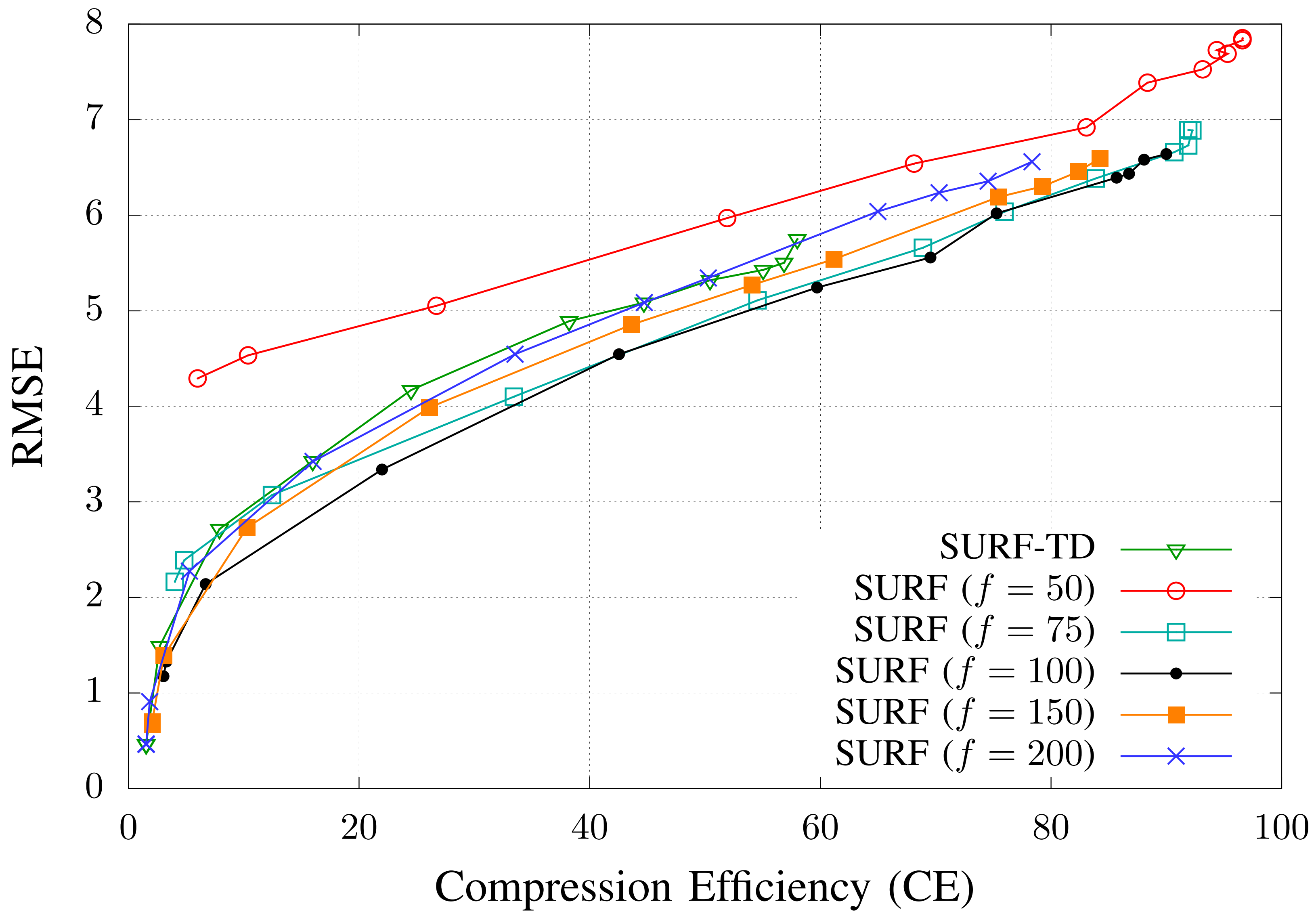
SURF: Dictionary Size



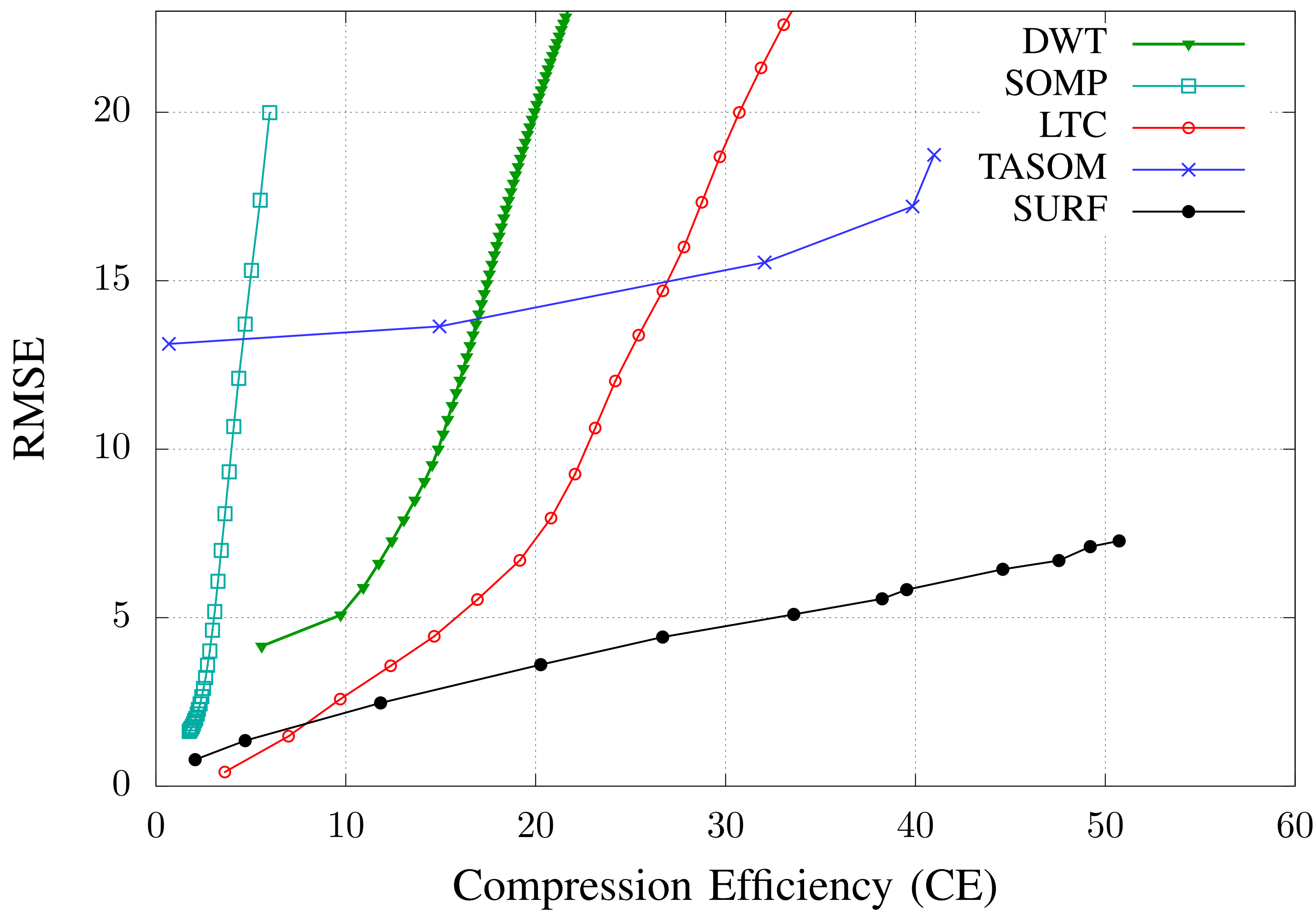
SURF: dictionary size

- Small error tolerance (left of previous graph)
  - Compressor often sends the feature vector
  - The codewords are seldom adequate for compression
  - New codewords are seldom added to the dictionary
    - Matches occur with smaller prob
    - Only a few codewords become recurrent
- Increasing error tolerance
  - New codewords will be added to the dictionary
- High error tolerance
  - Because of the relaxed accuracy requirements
  - Only a few good codewords suffice
  - The dictionary contains a small set of prototypes

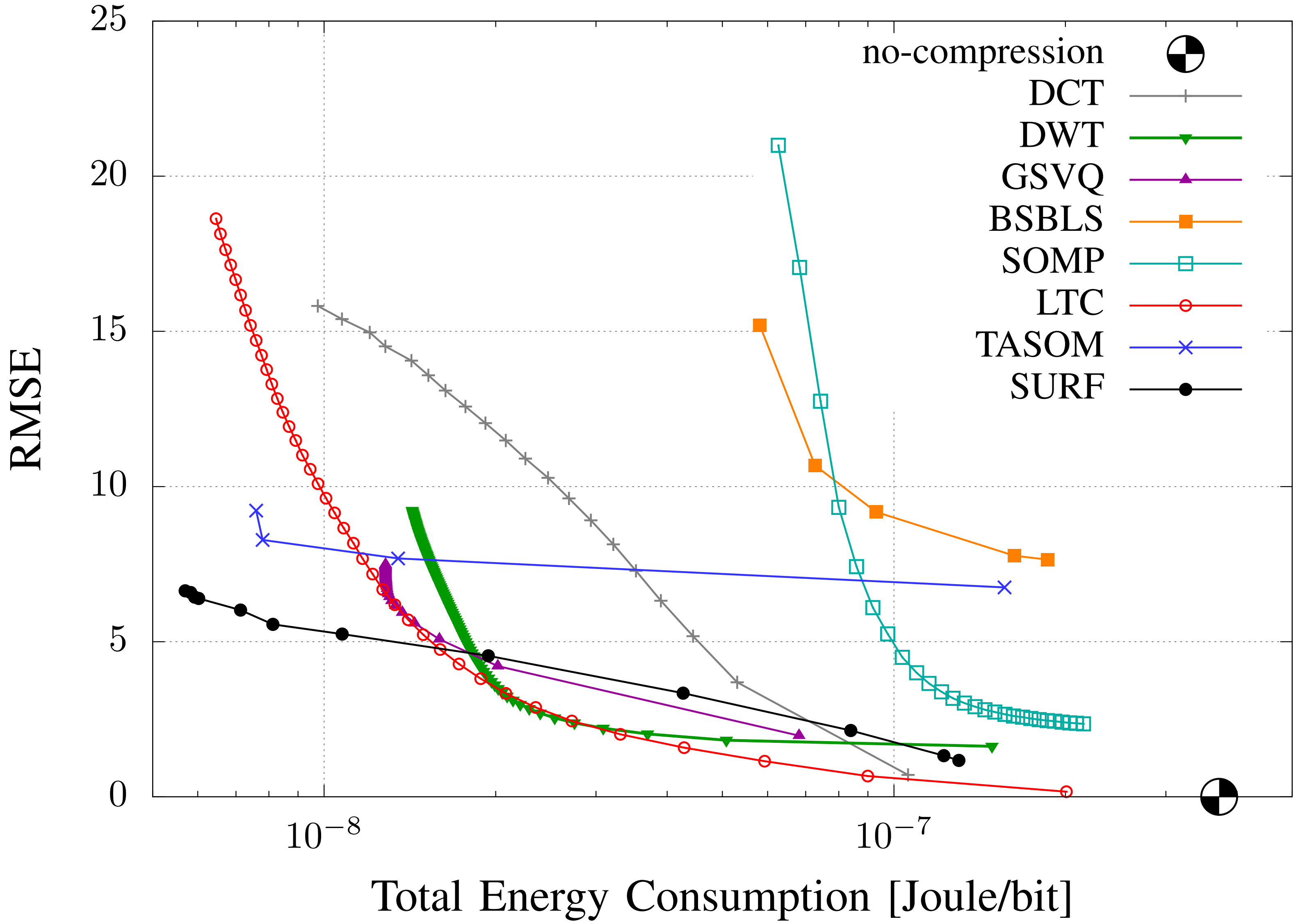
Impact of Number of DCT Coefficients



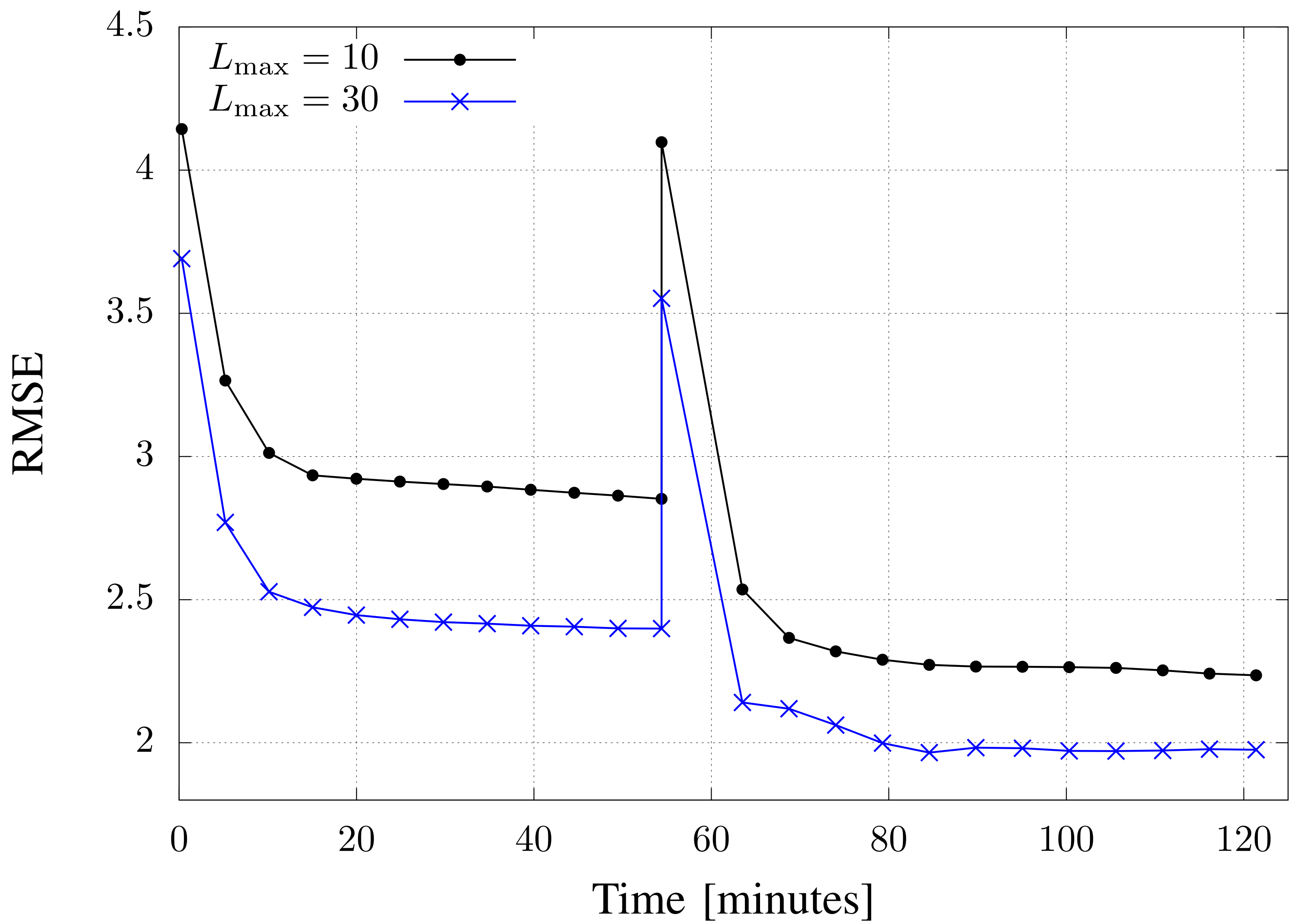
SURF: Compression Efficiency (artifacts-prone Bioharness ECG signals)



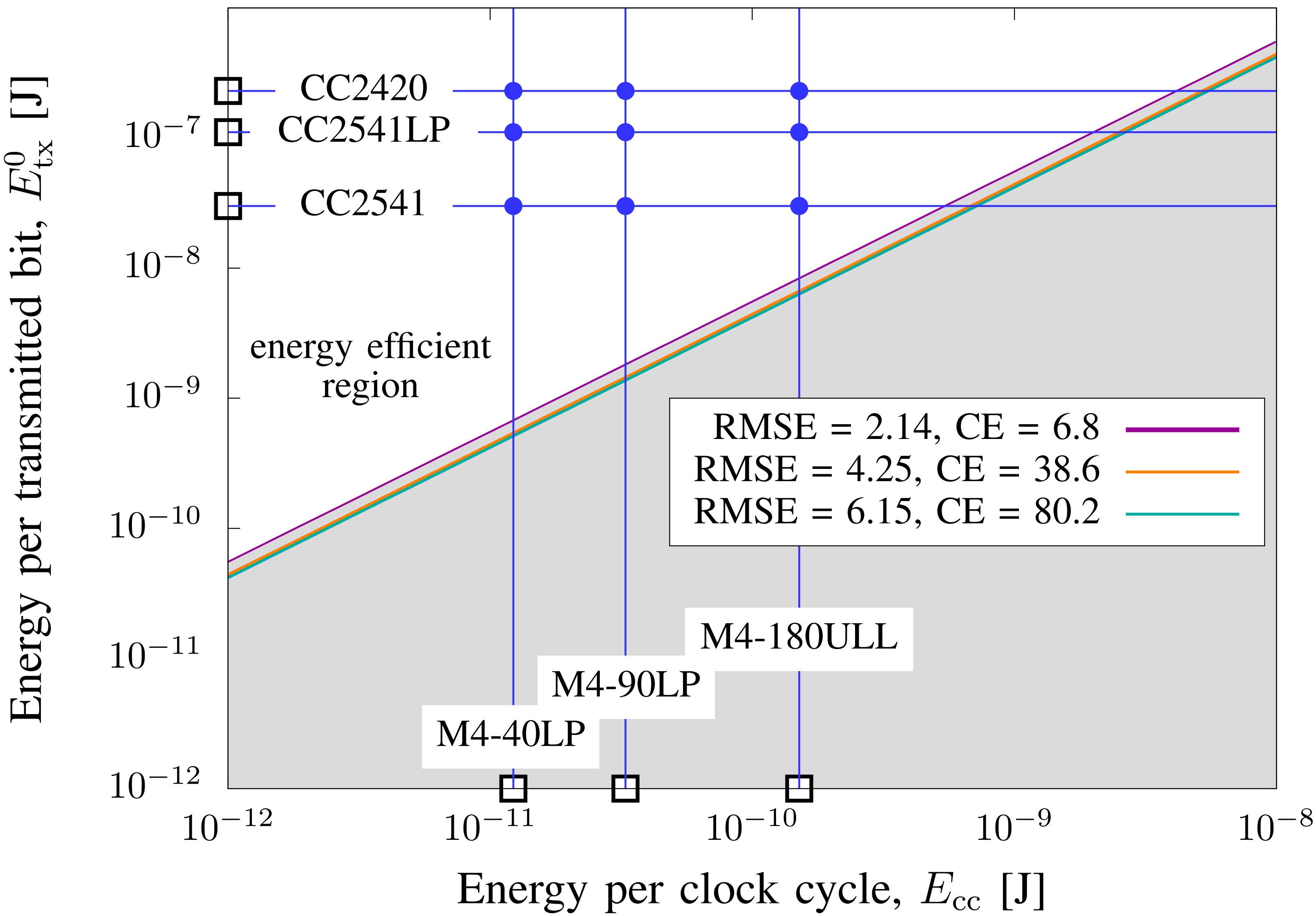
SURF: Total Energy Consumption



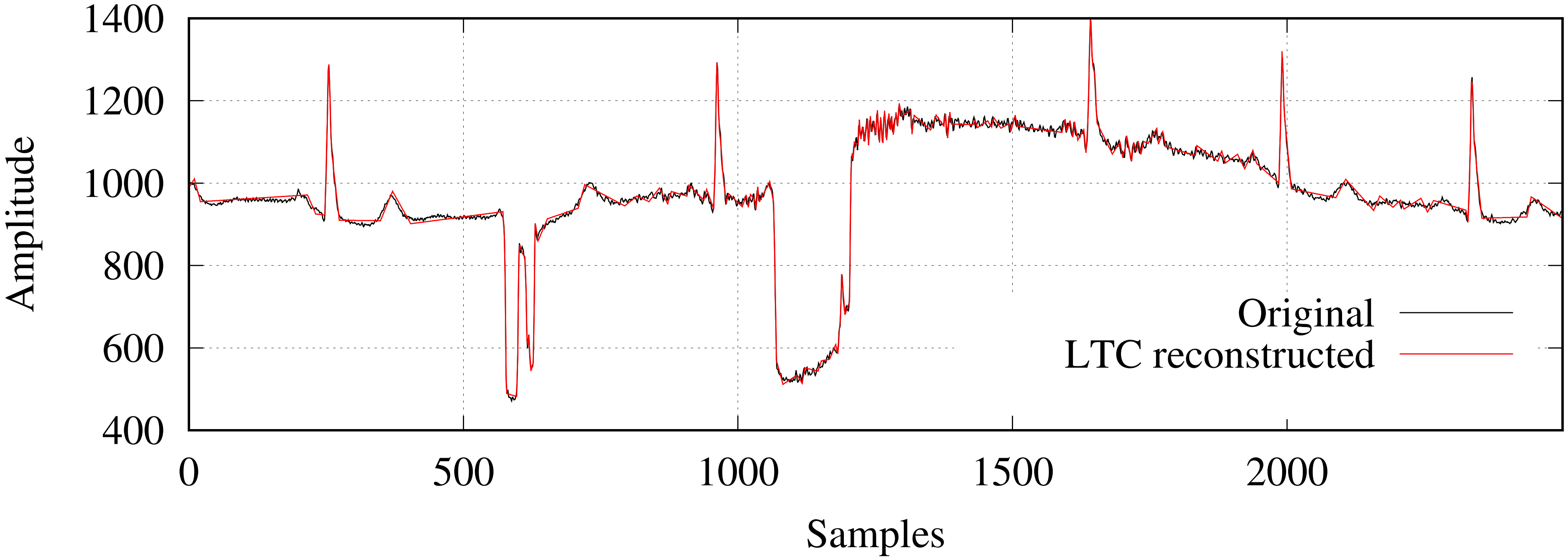
SURF: Adaptation to New Subjects



SURF: Energy Efficiency



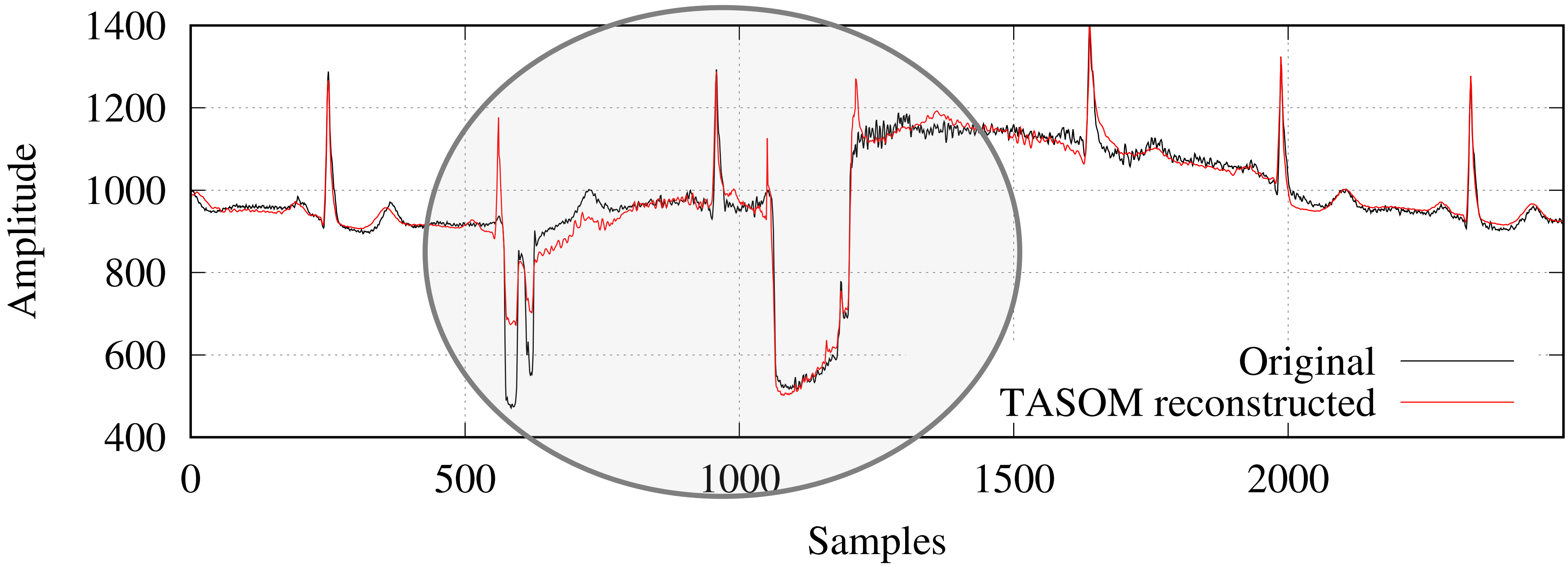
New results - LTC



(a) **LTC**: CE = 22 and RMSE = 2%.

New results - TASOM

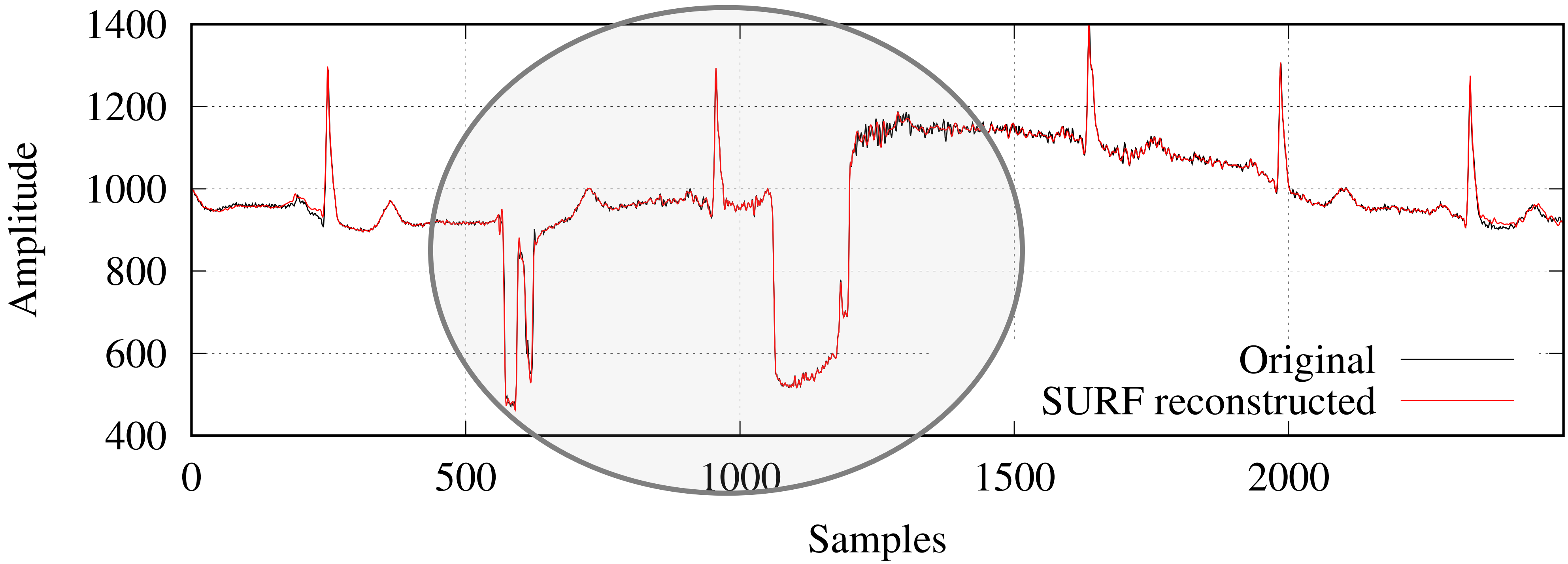
artifacts are poorly modeled by the dictionary



(c) **TASOM**: CE $= 34$ and RMSE $= 2\%$.

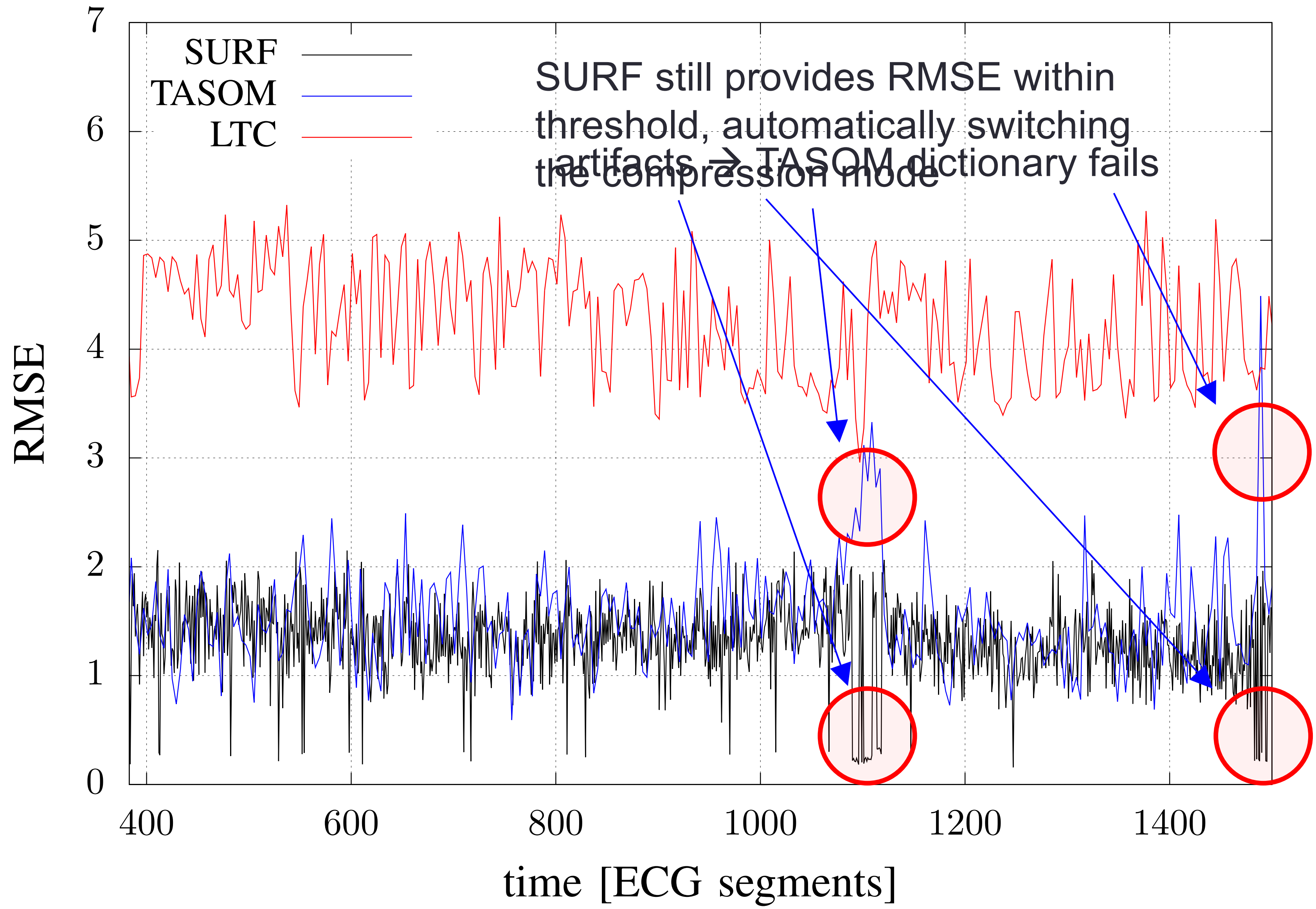
New results - SURF

artifact automatically detected – standard DCT compression is used in place of dictionary based

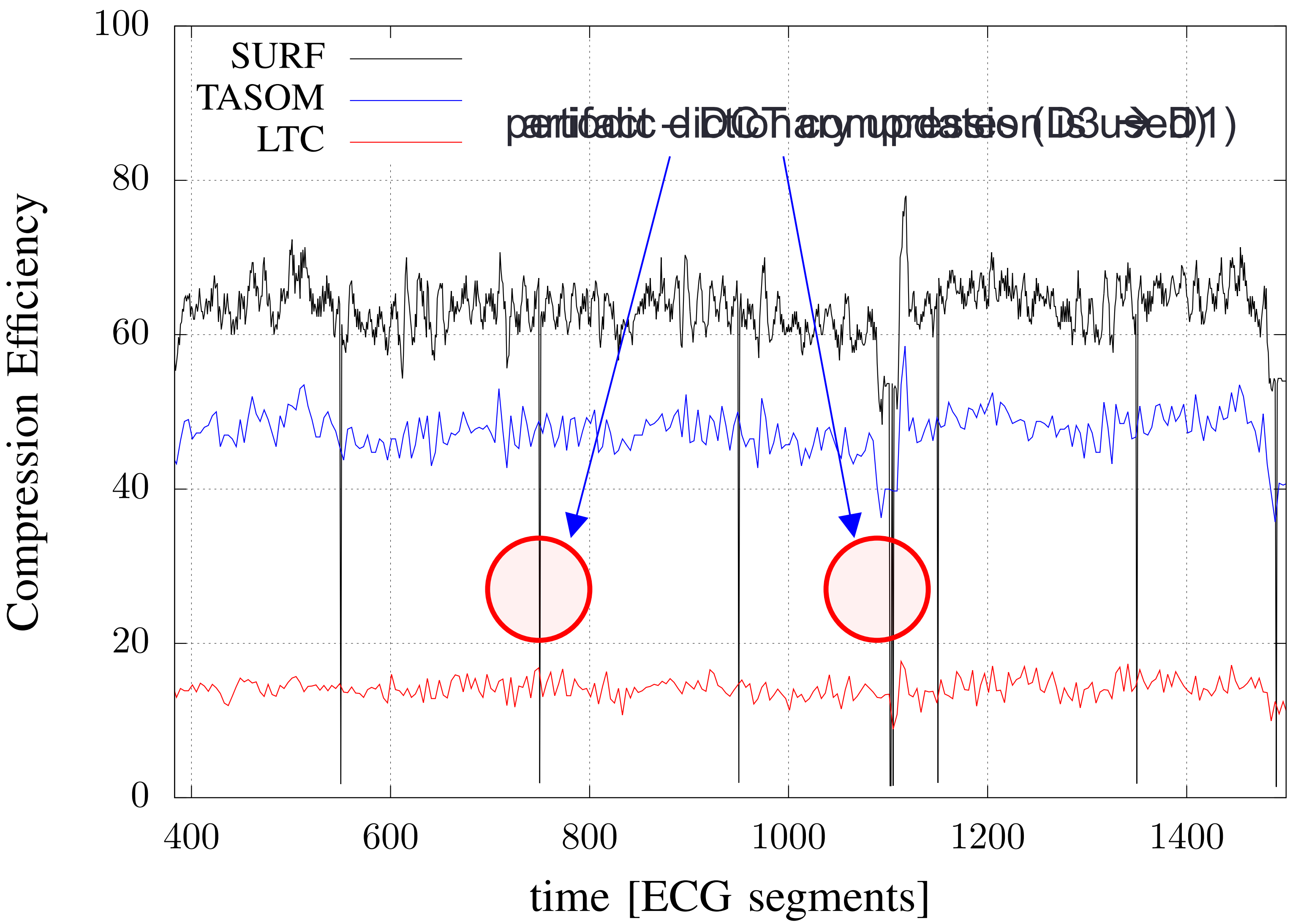


(e) **SURF**: CE $= 43$ and RMSE $= 2\%$.

Example Result: RMSE



Example Results: Compression Efficiency



Bibliography

[Hooshmand-2017-1] Mohsen Hooshmand, Davide Zordan, Tommaso Melodia, Michele Rossi, SURF: subject-adaptive unsupervised ECG signal compression for wearable fitness monitors, *IEEE Access,* 7 September 2017.

[Hooshmand-2017-2] Mohsen Hooshmand, Davide Zordan, Davide Del Testa, Enrico Grisan, Michele Rossi, Boosting the Battery Life of Wearables for Health Monitoring through the Compression of Biosignals, *IEEE IoT Journal*, 29 March 2017.

[DelTesta-2015] Davide Del Testa, Michele Rossi, Lightweight Lossy Compression of Biometric Patterns via Denoising Autoencoders, *IEEE Signal Processing Letters*, Vol. 22, No. 12, September 2015.

[Vadori-2016] Valentina Vadori, Enrico Grisan, Michele Rossi, Biomedical Signal Compression with Time- and Subject-adaptive Dictionary for Wearable Devices, *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, September 13-16, Vietri sul Mare, Salerno, Italy, 2016.

[Francescon-2015] Roberto Francescon, Mohsen Hooshmand, Matteo Gadaleta, Enrico Grisan, Seung Keun Yoon, Michele Rossi, Toward Lightweight Biometric Signal Processing for Wearable Devices, *IEEE Engineering in Medicine and Biology Society (EMBS)*, August 25-29, Milan, Italy, 2015.

BOOSTING THE EFFICIENCY OF SMART WEARABLE DEVICES THROUGH BIOMETRIC SIGNAL COMPRESSION

Michele Rossi

rossi@dei.unipd.it

Dept. of Inf. Engineering, Univ. of Padova, Italy

SAMSUNG ADVANCED INSTITUTE OF TECHNOLOGY

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE