Decomposition of Synthetic Surface Electromyograms Using Sequential Convolution Kernel Compensation

V. Glaser, A. Holobar, D. Zazula
Faculty of Electrical Engineering and Computer Science
University in Maribor
SLOVENIA
vojko.glaser@uni-mb.si

Abstract: - This paper reveals a sequential decomposition method based on Convolution Kernel Compensation (CKC). It operates in real time and can decompose linear mixtures of finite-length signals. Multiple-input multiple-output (MIMO) signal model is used with finite channel responses and trains of delta pulses as inputs. Our approach compensates the channel responses and reconstructs the input pulse trains in real time, when samples of the observed signals become available. Two versions of the sequential decomposition were tested: sample-by-sample and multiple sample input. The time complexity of the two approaches, is analytically evaluated and supported by the experimental measurements. Performance tests on the synthetic surface electromyograms (sEMG) provide estimations of average rate and standard deviation of the reconstructed pulse trains for single motor units (MU).

Key-Words: – Compound signal decomposition, Surface electromyogram, Real-time signal processing, Sherman-Morrison-Woodbury matrix inversion, Sequential convolution kernel compensation

1. Introduction

Linear modelling of multiple observations excited by multiple input source signals has been shown to be of practical value in many different signal processing problems. Over the past decades, it has been applied to the fields of biomedicine, radar, sonar, image processing, speech separation, etc.

In the field of electrophysiology, monitoring and diagnosing of the human neuromusclar system is frequently based on electromyograms (EMG). Different acquisition techniques have been proposed, enabling detection of electrical potentials in vicinity of muscle fibers or nerves. These measurable action potentials (AP) accompany muscle contractions and are controlled by the drive from the central nervous system (CNS). In normal conditions, a group of several tens or hundreds of muscle fibers is innervated by a single motoneuron, forming a basic functional unit of skeletal muscles, the so called motor unit (MU) [2]. In a muscle, several tens of MUs are active simultaneously, even at moderate contraction levels.

Recently, multichannel acquisition systems became available, enabling noninvasive simultaneous recordings of up to several hundreds of surface EMG (sEMG) channels. Due to low selectivity of surface electrodes, the measured signals are typically composed of contributions of many individual motor-units, whose action potentials (MUAPs) sum up to form highly interferential signal patterns [2]. A real EMG diagnostic value, whose application has been paved by the tradition of the needle EMG measurements, is based on the knowledge about its constituent components observed through individual MUAPs. Therefore, a strong need for reliable, robust and fast methods for decomposition of sEMG signals has recently emerged.

One of the new promising decomposition approaches was developed in the System Software Laboratory at the University of Maribor and called Convolution Kernel Compensation (CKC) [1]. Its original derivation utilizes the correlation matrix built out of several simultaneous measurements that are processed in rather long segments. Hence, the underlying approach is batch, which does not meet the requirements for a fast, real-time decomposition. A sequential version of CKC (sCKC) was partially first derived in [3]. Although being computationally efficient, sCKC suffers from an inferior quality of decomposed signals. In this paper, we introduce a novel

block-sequential CKC (bsCKC) method, which employs blocks of sEMG signals and combines the computation efficiency of sCKC with accuracy of batch CKC.

The main goal of this paper is to analyse the sequential methods' (sCKC and bsCKC) performance on synthetic sEMGs and their computational complexity.

Our manuscript is organized as follows. In Section 2, a brief summary of the batch version of CKC is provided, followed by the derivations of sCKC and bsCKC in Section 3. Section 4 outlines the computational efficiency of both sCKC and bsCKC methods, whereas the performance of sCKC on synthetic sEMGs is discussed in Section V. The conclusions are presented in Section VI.

2. Data Model and Convolution Kernel Compensation technique

The basic formulation of the CKC approach [1] uses a MIMO model of sEMG. It assumes M different observations $\mathbf{x}_i = \{x_i(n); n = 0,1,2,...\};\ i = 1,...,M$, each comprising superimpositions of responses from N sources:

$$x_i(n) = \sum_{j=1}^{N} \sum_{l=0}^{L-1} h_{ij}(l)\, s_j(n-l); \quad i = 1,...,M, \tag{1}$$

where $h_{ij}(l)$ is a L-samples long response of the j-th source, as it appears in the i-th observation, whereas $s_j(n-l)$ stands for a train of delta pulses from the j-th source. When noisy observations are considered, (1) is modified as follows:

$$y_i(n) = x_i(n) + \omega_i(n); \quad i = 1,...,M, \tag{2}$$

where $\boldsymbol{\omega}_i(n)$ stands for stationary, zero-mean white Gaussian random noise.

In order to improve the ratio between the number of observations and the number of unknowns, the model (1) is extended with K-1 delayed repetitions of each observation:

$$\bar{\mathbf{y}}(n) = \left[y_1(n), y_1(n-1), ..., y_1(n-K+1), ..., y_M(n), ..., y_M(n-K+1)\right]^T. \tag{3}$$

Adopting the model from Eq. (2) and the extensions from Eq. (3), an *index of global pulse train activity* can be defined as follows:

$$\gamma(n) = \bar{\mathbf{y}}^T(n) \cdot \mathbf{C}^{\#}_{\bar{\mathbf{y}}\bar{\mathbf{y}}} \cdot \bar{\mathbf{y}}(n) \approx \bar{\mathbf{s}}^T(n) \cdot \mathbf{C}^{-1}_{\bar{\mathbf{s}}\bar{\mathbf{s}}} \cdot \bar{\mathbf{s}}(n) \tag{4}$$

where $^{\#}$ stands for pseudoinverse, ^{T} and $^{-1}$ for transpose and inverse, respectively, $\mathbf{C}_{\bar{\mathbf{y}}\bar{\mathbf{y}}}$ is the correlation matrix of the noisy observations, $\bar{\mathbf{y}}_i$, and $\mathbf{C}_{\bar{\mathbf{s}}\bar{\mathbf{s}}}$ the correlation matrix of noise-free source signals, $\bar{\mathbf{s}}_i$. Suppose we fix the premultiplying vector $\bar{\mathbf{y}}(n_0)$ in (4) to the position n_0, and rewrite the expression:

$$\nu_{n_0}(n) = \bar{\mathbf{y}}^T(n_0)\mathbf{C}^{\#}_{\bar{\mathbf{y}}\bar{\mathbf{y}}}\bar{\mathbf{y}}(n). \tag{5}$$

It was proven in [1] that when only the j-th source is active in time sample n_0, $\nu_{n_0}(n)$ yields the estimation of the pulse train from the j-th source:

$$\nu_{n_0}(n) \approx \bar{s}_j(n). \tag{6}$$

Eq. (5) gives a recipe for decomposing a linear, convolutive mixture of pulse trains. Sensitivity of approximation (6) was explained in [1].

3. Sequential CKC with Single or Multiple Sample Input

Sequential CKC upgrades the batch CKC method by reducing the time support for computation of correlation matrix $\mathbf{C}_{\bar{\mathbf{y}}\bar{\mathbf{y}}}$ from the entire signal segment to a single sample or block of consequent samples. It also updates premultiplying vectors $\bar{\mathbf{y}}^T(n_0)$ from Eq. (5) in iterative manner, as already explained in [3].

Both sequential methods comprise two major steps. In the first step, premultiplying vector $\bar{\mathbf{y}}^T(n_0)$ is initialized and, afterwards, iteratively optimized to yield the filter $\mathbf{f}_j$ of the single MU innervation pulse train s_j [3]. This initialisation is explained in detail in Subsection 3.1. In the second step, the inverse of correlation matrix $\mathbf{C}^{\#}_{\bar{\mathbf{y}}\bar{\mathbf{y}}}$ is calculated in iterative manner, as outlined in Subsection 3.2.

3.1 sCKC Initialisation

Just like its batch version, sCKC uses activity index $\gamma(n)$ to pinpoint the time instant n_0 that defines the starting filter:

$$\mathbf{f}_j = \overline{\mathbf{y}}(n_0), \tag{7}$$

where index *j* counts the number of different filters that produce different pulse trains. Instead of using entire $\overline{\mathbf{y}}(n)$, sCKC uses only the first *S* samples of observations for the initialisation. In order to avoid the reidentification of the already identified pulse train, sCKC method prevents selection of another time instant n_j within a region ε:

$$\varepsilon = (n_j - \frac{L}{2}, n_j + \frac{L}{2}), \tag{8}$$

where *L* is the maximal MUAP length (in samples). When all possible candidates for starting filters f_j are selected from activity index, sCKC gradually improves them using element-wise product between decompositions of initially recognised pulse trains

$$\nu_{n_0 n_1}(n) = \nu_{n_0}(n) \cdot \nu_{n_1}(n), \tag{9}$$

which emphasizes only pulses triggered in a MUs active in time instants n_0 and n_1, respectively. It was estimated in [1] that already four proper positions n_i combine in a pulse train which most probably belongs to a single source. Denoting this train with $\nu_{n_0 n_1 n_2 n_3}(n)$, suppose that pulses in this train appear at $t_0, \ldots, t_{g-1}$. Then by averaging all the observation vectors in this time instants, an improved filter is constructed:

$$\mathbf{f}_j = \frac{1}{g} \sum_{i=0}^{g-1} \overline{\mathbf{y}}(t_i). \tag{10}$$

After initialization, sCKC checks for possible repetitions of the same pulse train using a threshold

$$T_j = \max(\nu_{n_i}) \cdot 0.85 \tag{11}$$

to discriminate the pulses in the j-th estimated train ν_{ni} from the baseline noise. Resulting MU discharge patterns are then mutually compared and all their repetitions discarded.

3.2 Sequential Inverse of Correlation Matrix

Sequential CKC method relies on iterative calculation of the inverse $\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}$, which proves to be computationally highly intensive. To cope with this problem, the following Sherman-Morrison formula was proposed in [5] for the cases with the single-sample inputs:

$$(\mathbf{C}_{\overline{\mathbf{yy}}} + \overline{\mathbf{y}}(n)\overline{\mathbf{y}}(n)^T)^{-1} = \mathbf{C}_{\overline{\mathbf{yy}}}^{-1} - \frac{\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}\overline{\mathbf{y}}(n)\overline{\mathbf{y}}(n)^T\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}}{1 + \overline{\mathbf{y}}(n)^T\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}\overline{\mathbf{y}}(n)} \tag{12}$$

where $\overline{\mathbf{y}}(n)$ is a vector of newly available samples of extended observations used for updating $\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}$. When a block of *P* signal samples of extended observations $\overline{\mathbf{Y}}_P(n) = \overline{\mathbf{y}}(n:n+P)$ is available, (12) can be replaced by Sherman-Morrison-Woodbury formula [4]:

$$\begin{aligned}&(\mathbf{C}_{\overline{\mathbf{yy}}} + \overline{\mathbf{Y}}_P(n)\overline{\mathbf{Y}}_P(n)^T)^{-1} = \\ &\mathbf{C}_{\overline{\mathbf{yy}}}^{-1} - \mathbf{C}_{\overline{\mathbf{yy}}}^{-1}\overline{\mathbf{Y}}_P(n)(\mathbf{I} + \overline{\mathbf{Y}}_P(n)^T\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}\overline{\mathbf{Y}}_P(n))^{-1}\overline{\mathbf{Y}}_P(n)^T\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}\end{aligned} \tag{13}$$

where **I** stands for matrix identity. Eq. (13) provides a theoretical framework for block-sequential CKC method. However, the application of (12) or (13) causes the estimated pulse amplitudes decrease in time, as shown in Fig. 1. This is mainly because (12) and (13) lack the normalization by the number of signal samples. As a result, the values of the matrix inverse $\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}$ decrease with iteration steps in (12) or (13).

To avoid this problem, the inverse matrix must be renormalized by the number of available samples, which turns (10) into:

$$\mathbf{C}_{\overline{\mathbf{yy}},n+1}^{-1} = (n+1)\left(\frac{\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}}{n} - \frac{\frac{\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}}{n}\overline{\mathbf{y}}(n)\overline{\mathbf{y}}(n)^T\frac{\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}}{n}}{1 + \overline{\mathbf{y}}(n)^T\frac{\mathbf{C}_{\overline{\mathbf{yy}}}^{-1}}{n}\overline{\mathbf{y}}(n)}\right) \tag{14}$$

and (11) into :

$$\mathbf{C}^{-1}_{\overline{\mathbf{y}}\overline{\mathbf{y}},n+P} = (n+P)\left(\frac{\mathbf{C}^{-1}_{\overline{\mathbf{y}}\overline{\mathbf{y}},n}}{n} - \frac{\mathbf{C}^{-1}_{\overline{\mathbf{y}}\overline{\mathbf{y}},n}}{n}\overline{\mathbf{Y}}_P(n)(\mathbf{I}+\overline{\mathbf{Y}}_P^T(n)\frac{\mathbf{C}^{-1}_{\overline{\mathbf{y}}\overline{\mathbf{y}},n}}{n}\overline{\mathbf{Y}}_P(n))^{-1}\overline{\mathbf{Y}}_P^T(n)\frac{\mathbf{C}^{-1}_{\overline{\mathbf{y}}\overline{\mathbf{y}},n}}{n}\right) \quad (15)$$

where $\mathbf{C}^{-1}_{\overline{\mathbf{y}}\overline{\mathbf{y}},n}$ is the correlation matrix inverse of extended observations, calculated over n samples of observations.

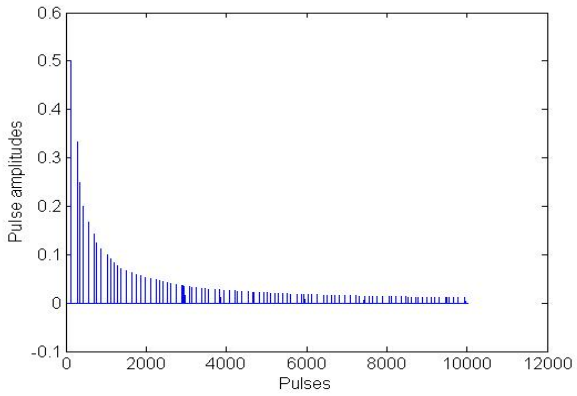

Fig. 1: Estimated pulse train as obtained by sCKC without normalisation.

3.3 Sequential Filter Update

Because the initialisation block of observations is rather short in comparison to the entire observation length, it is unlikely that the produced initialisation filters are sufficient for successful decompositions of pulses discharged in single MUs. Therefore, sequential filters have to be updated and improved after every new observation sample vector.

Sequential filter update relies on running average:

$$\mathbf{f}_{j,g+1} = \frac{g \cdot \mathbf{f}_{j,g} + \overline{\mathbf{y}}(t_{g+1})}{g+1}, \quad (16)$$

where g is the number of observations deployed in the filter $\mathbf{f}_j$ before the update with the new observation $\overline{\mathbf{y}}(t_{g+1})$.

3.4 Sequential Removal of Multiple Pulse Train Recognitions

Assume that $\mathbf{f}_{j_0}$ and $\mathbf{f}_{j_1}$ that produce trains $\nu_{\mathbf{f}_{j_0}}(n)$ and $\nu_{\mathbf{f}_{j_1}}(n)$ are improved in such way that from certain observation m_0 onward, their pulses belong to the same pulse train:

$$\nu_{\mathbf{f}_{j_0}}(m) = \nu_{\mathbf{f}_{j_1}}(m); \quad m = m_0, m_0+1\ldots, \quad (17)$$

whereas a different notation is used here for the pulse trains with subscripts identifying the premultiplying decomposition filter instead of a referential sample number.

Suppose that two pulse trains $\nu_{\mathbf{f}_{j_0}}(m)$ and $\nu_{\mathbf{f}_{j_1}}(m)$ are equal for $m = m_0 \ldots m_0 + D$, where D is a preselected threshold. In this case, the trains are considered produced by the same source, so that one of them can be discarded.

4. Computational Complexity

To evaluate the decomposition speed, we estimated computational complexity analytically and also measured on a standard PC with 3000 MHz CPU and 2 GB of memory, as described in the sequel. In Subsection 4.1, analytical calculations for the sCKC time complexity is presented, while in Subsection 4.2 the results of its measurements follow.

4.1 Analytical Estimate of Computational Complexity

The aim of sCKC is to be a real-time decomposition method. The most complex and most frequent operation is the update of the correlation matrix inverse, in particular when it is computed sample by sample. The complexity of optimized calculations of single-sample correlation matrix inverse using Sherman-Morrison formula is shown in Tables 1 and 2. Table 1 counts the number of multiplications, while Table 2 shows the number of additions. As already proven in [4], Sherman-Morrison formula has quadratic time complexity.

In the Tables 3 and 4, the operations and their time complexity using Sherman-Morrison-Woodbury formula for block-sample updates of the correlation matrix inverse are depicted. Again, Table 3 counts the number of multiplications, while Table 4 shows the number of additions. As we can see, the time complexity of Sherman-Morrison-Woodbury formula is in general cubic.

The times measured during our experimental runs of sCKC are gathered in the next section, and they comply with the analytical predictions.

Table 1: Number of multiplications in the application of Sherman-Morrison formula. *KM* stands for the size of correlation matrix inverse, *n* is the number of observations that contributed to the correlation matrix. Vector **a** and matrix **B** mean auxiliary variables used in the analytical derivation.

Operation	Number of multiplications
$\overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} = \frac{\mathbf{C}^{-1}_{\overline{\mathbf{yy}},n}}{n}$	$\frac{(KM)^2}{2} + \frac{KM}{2}$
$\mathbf{a} = \overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} \cdot \overline{\mathbf{y}}(n+1)$	$\frac{(KM)^2}{2} + \frac{KM}{2}$
$\mathbf{B} = \frac{\mathbf{a} \cdot \mathbf{a}^T}{1 + \mathbf{y}^T(n+1) \cdot \mathbf{a}}$	$(KM)^2 + 2KM$
$\mathbf{C}^{-1}_{\overline{\mathbf{yy}},n+1} = (n+1)\left(\overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} - \mathbf{B}\right)$	$\frac{(KM)^2}{2} + \frac{KM}{2}$
Total:	$\frac{5}{2}(KM)^2 + \frac{7}{2}KM$

Table 2: Number of additions in the application of Sherman-Morrison formula. *KM* stands for the size of correlation matrix inverse, *n* is the number of observations that contributed to the correlation matrix. Vector **a** and matrix **B** mean auxiliary variables used in the analytical derivation.

Operation	Number of additions
$\overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} = \frac{\mathbf{C}^{-1}_{\overline{\mathbf{yy}},n}}{n}$	0
$\mathbf{a} = \overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} \cdot \overline{\mathbf{y}}(n+1)$	$\frac{(KM)^2}{2} + \frac{KM}{2}$
$\mathbf{B} = \frac{\mathbf{a} \cdot \mathbf{a}^T}{1 + \mathbf{y}^T(n+1) \cdot \mathbf{a}}$	KM
$\mathbf{C}^{-1}_{\overline{\mathbf{yy}},n+1} = (n+1)\left(\overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} - \mathbf{B}\right)$	$\frac{(KM)^2}{2} + \frac{KM}{2} + 1$
Total:	$(KM)^2 + 2KM + 1$

Table 3: Number of multiplications in the application of Sherman-Morrison-Woodbury formula. *P* stands for the length of input blocks of observations, *KM* for the size of correlation matrix inverse, **I** is the identity, and **A**, $\mathbf{A}_1$, **B**, and **D** designate auxiliary matrices introduced for the sake of derivation.

Operation	Number of multiplications
$\overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} = \frac{\mathbf{C}^{-1}_{\overline{\mathbf{yy}},n}}{n}$	$\frac{(KM)^2}{2} + \frac{KM}{2}$
$\mathbf{A} = \mathbf{Y}^T \overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} \mathbf{Y}$	$2(KM)^2 P$
$\mathbf{A}_1 = (\mathbf{I} + \mathbf{A})^{-1}$	$(P)^3$
$\mathbf{B} = \overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} \mathbf{Y}$	$(KM)^2 P$
$\mathbf{D} = \mathbf{B}\mathbf{A}_1\mathbf{B}^T$	$2P^2KM$
$\overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} - \mathbf{D}$	0
$\mathbf{C}^{-1}_{\mathbf{YY},n+P}$	$\frac{(KM)^2}{2} + \frac{KM}{2}$
Total:	$P^3 + 3((KM)^2 P) + 2P^2KM + (KM)^2 + KM$

Table 4: Number of additions in the application of Sherman-Morrison-Woodbury formula. *P* stands for the length input blocks of observations, *KM* for the size of correlation matrix inverse, **I** is the identity, and **A**, $\mathbf{A}_1$, **B**, and **D** designate auxiliary matrices introduced for the sake of derivation.

Operation	Number of multiplications
$\overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} = \frac{\mathbf{C}^{-1}_{\overline{\mathbf{yy}},n}}{n}$	0
$\mathbf{A} = \mathbf{Y}^T \overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} \mathbf{Y}$	$2(KM)^2 P$
$\mathbf{A}_1 = (\mathbf{I} + \mathbf{A})^{-1}$	P
$\mathbf{B} = \overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} \mathbf{Y}$	$(KM)^2 P$
$\mathbf{D} = \mathbf{B}\mathbf{A}_1\mathbf{B}^T$	$2P^2KM$
$\overline{\mathbf{C}}^{-1}_{\overline{\mathbf{yy}},n} - \mathbf{D}$	$\frac{KM}{2}$
$\mathbf{C}^{-1}_{\mathbf{YY},n+P}$	0
Total:	$3(KM)^2 P + 2P^2KM + KM + P$

4.2 Measured Time Complexity

Three different experiments of computational complexity were performed on synthetic observations. In the first experiment, we measured the processing time needed to invert the correlation matrix when using different number of samples in the block inputs. In the second experiment, we investigated the impact of the size of correlation matrix on the processing time needed

for its inversion. The third experiment focused on the overall processing times of bsCKC and sCKC methods, when decomposing random mixtures of synthetic pulse trains.

Generated observations used in the experiments were produced from 10 different pulse trains, 10.000 samples long. The distance between pulses was normally distributed with a mean of 100 samples and variance of 30 samples. Channel responses were 8 samples long and the extension factor for CKC was set to 10 in all experiments.

The first experiment was conducted on a set of 10 and 50 observations. The number of samples P in the input block $\overline{\mathbf{Y}}_P(n)$ ranged from 200 to 1000 in steps of 100 samples. Forty Monte-Carlo runs were conducted with different input pulse trains and channel responses generated randomly for each simulated value of P, whereas the size of correlation matrix was set to 100×100 and 500×500, respectively. Results, averaged over all simulation runs, are depicted in Fig. 2. In all the cases, standard deviations were negligibly small and are not reported.

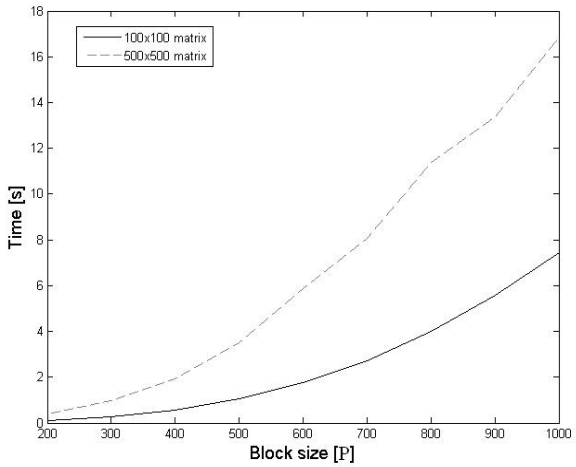

Fig. 2: Average computation time of matrix inverse with block inputs of size from 200 to 1000 samples. Standard deviations were negligibly small and are not reported. Simulations were conducted on a standard PC with 3000 MHz CPU and 2 GB of memory.

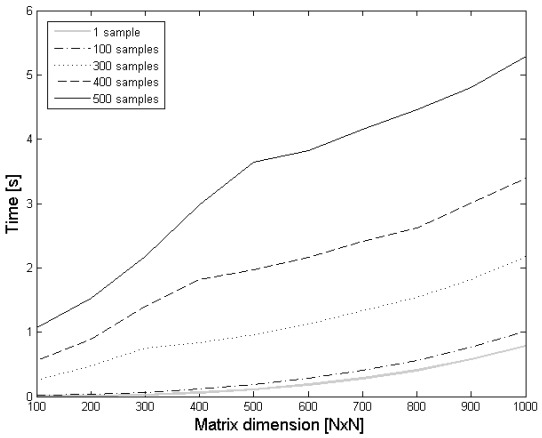

Fig. 3: Average computation time of matrix inverse for different input block lengths versus different matrix dimension. Standard deviations were negligibly small and are not reported. Simulations were conducted on a standard PC with 3000 MHz CPU and 2 GB of memory.

In the second experiment, length of input blocks $\overline{\mathbf{Y}}_P(n)$ was set equal to 1, 100, 300, 400 and 500 samples, respectively. Different sizes of correlation matrix were simulated, ranging from 100×100 to 1000×1000. Twenty Monte-Carlo runs were conducted with different observations, as described for the first experiment, per each matrix size and each length of input block $\overline{\mathbf{Y}}_P(n)$. The results are shown in Fig. 3.

Now take the analytical expression for the number of multiplications in Sherman-Morrison-Woodbury formula from Table 3. When KM is significantly smaller than P, time complexity of Sherman-Morrison-Woodbury formula is cubic, but when KM surpasses P significantly, the formula becomes quadratic, which can be best seen in Fig. 3 for P=500 samples.

In the last experiment, 10, 30 and 50 simulated observations were used. The initial window was 500 samples long. Twenty Monte-Carlo runs were conducted for three matrix dimensions versus different input block lengths. The results are depicted in Fig. 4.

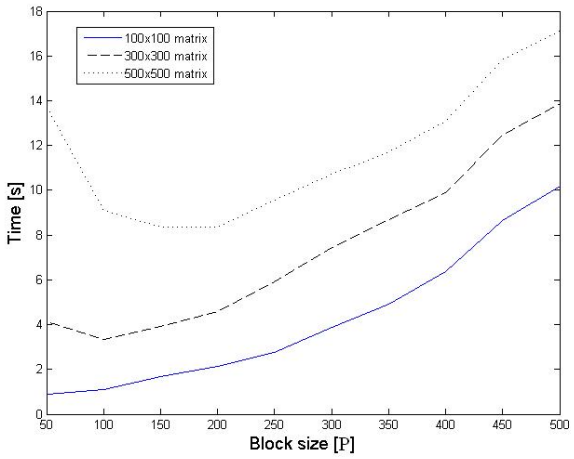

Fig. 4: Average decomposition time when using the bsCKC method versus the input block size. Standard deviations were negligibly small and are not reported.

Assume also we have observations with *Q* samples and *S* samples are used for the initialization and *P* samples for every input block. Then the number of required updates of correlation matrix inverse can be calculated as

$$R = \frac{(Q-S)}{P}. \tag{19}$$

The number of required updates *R* decreases with *P*, whereas the time needed for calculation of each update of correlation matrix inverse increases with *P*, as shown in previous experiments. For small values of *P*, the number of required updates *R* has a dominant effect on the total CKC processing time. Eventually, negative effect of increased block size *P* overrides the positive effect of smaller number of inverse updates *R*, causing the entire bsCKC processing time to increase with *P*. The optimal performance of bsCKC is achieved when

$$\frac{KM}{3} < P < \frac{KM}{2} \tag{20}$$

5. Decomposition Performance

SCKC decomposition performance was measured on synthetic sEMGs [6] generated by a sEMG simulator [6] developed in the LISiN Laboratory in Turin, Italy.

In the pulse train decomposition three situations can arise. Firstly, suppose $s_{j_0}(n_0) = 1$ and $v_{\mathbf{f}_{j0}}(n_0) < T_{j_0}$ where $v_{\mathbf{f}_{j_0}}(n)$ is the decomposition of pulse train s_{j_0}. All such pulses are termed missed pulses (false negative). Secondly, when $s_{j_0}(n_0) = 0$ and $v_{\mathbf{f}_{j_0}}(n_0) > T_{j_0}$, we have misplaced pulses (false positive). And thirdly, when $s_{j_0}(n_0) = 1$ and $v_{\mathbf{f}_{j_0}}(n_0) > T_{j_0}$ this means a correct pulse recognition (true positive).

We verified our decomposition method on synthetic sEMGs and studied the influence of noise corruption on the decomposition performance. Recognition rates and rates of missed and misplaced pulses are presented in subsequent sections.

5.1 Synthetic sEMG

A sEMG model [6] from the LISiN Laboratory in Torino was used to generate synthetic signals. Constant contraction forces, planar volume conductor muscle model [7], and double differential uptake electrodes were simulated. Innervation firing patterns were sampled by 4096 Hz, while the sEMG signal sample frequency was 1024 Hz. Firing patterns were 122880 samples long, producing sEMGs with 30720 samples.

Three different sets of sEMGs were generated, with 5, 10 and 20 simulated active MUs. Each set comprise 5 different sEMGs.

5.2 The Influence of the Number of Active MUs

In the first simulation, the influence of different number of active MUs on the decomposition was studied. Synthetic sEMG signals used were corrupted by zero-mean additive white Gaussian noise with 30 dB signal-to-noise ratio (SNR) [8].

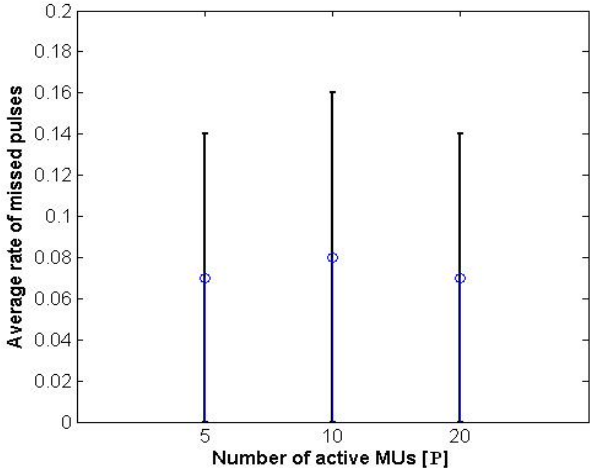

Fig. 5: Average rate and standard deviation of missed pulses versus the number of active MUs.

If the number of active MUs rises, so does the probability of MUAPs overlapping, which causes the decomposed pulse trains of several MUs superimpose. Consequently, the average rate of misplaced pulses grows with more active MUs, which can be seen in Fig. 6. Due to the decomposition filters that are improved iteratively, according to (16), some pulse trains are decomposed correctly only after some time delay, which means they may appear with misplaced pulses before.

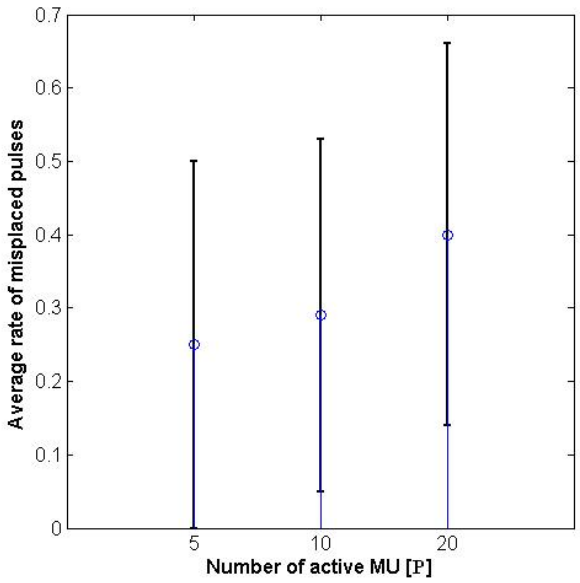

Fig. 6: Average rate and standard deviation of misplaced pulses versus the number of active MUs.

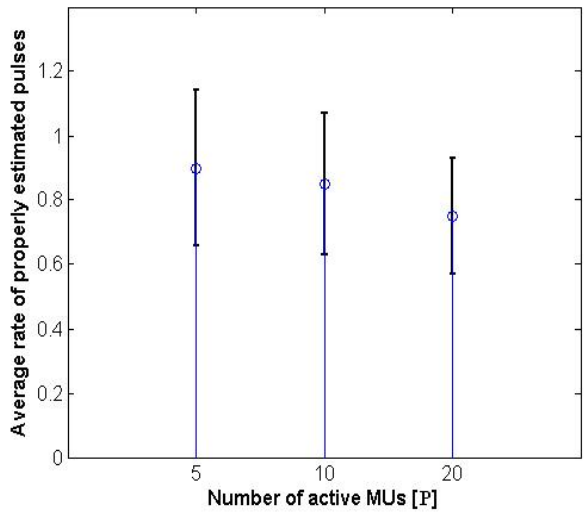

Fig. 7: Average rate and standard deviation of properly estimated pulses versus the number of active MUs.

Fig. 7 shows the average pulse recognition rate and standard deviation. Although the average recognition rate decreases with higher number of active MUs, sCKC still produces some decompositions with more than 90 % of properly placed pulses.

5.3 The Influence of Noise

We also ran simulations to study the influence of noise corruption on the average rate of missed, misplaced and recognised pulses. The batch CKC has proven to be very robust and noise resistant [1]. To estimate the robustness of sCKC, we experimented with zero-mean white Gaussian noise at three different SNRs, i.e. 30, 20 and 10 dB [8]. The number of active MUs was set to 5, whereas other properties of sEMG were the same as in the simulations described in previous subsections.

Fig. 8 depicts the average rate and standard deviation of missed pulses versus the different SNRs. It is evident the noise corruption does not have significant influence in this case.

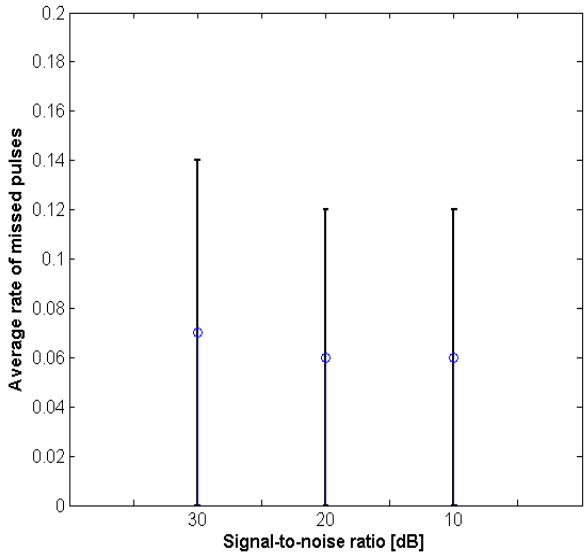

Fig. 8: Average rate and standard deviation of missed pulses versus different SNRs of 30, 20 and 10 dB. Zero-mean random white Gaussian noise was used.

Average rate and standard deviation of misplaced pulses is shown in Fig. 9. The rate increases considerably with higher noise corruption. This is due to the noise influence on activity index $\gamma(n)$ which is the main source for a selection of decomposition filters (7) and, therefore, recognition success.

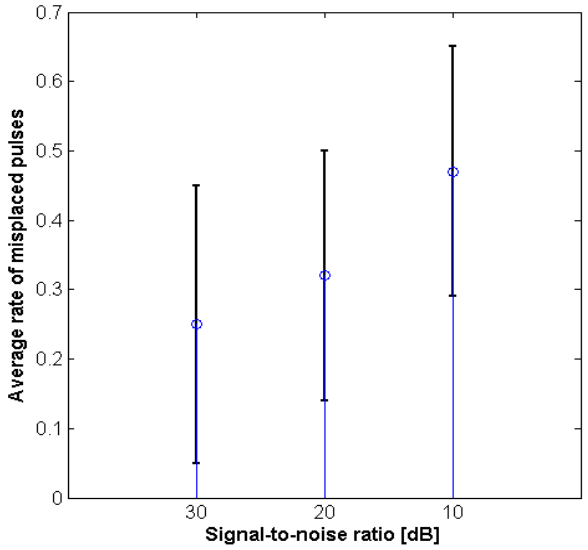

Fig. 9: Average rate and standard deviation of misplaced pulses versus the different SNRs of 30, 20 and 10 dB. Zero-mean random white Gaussian noise was used.

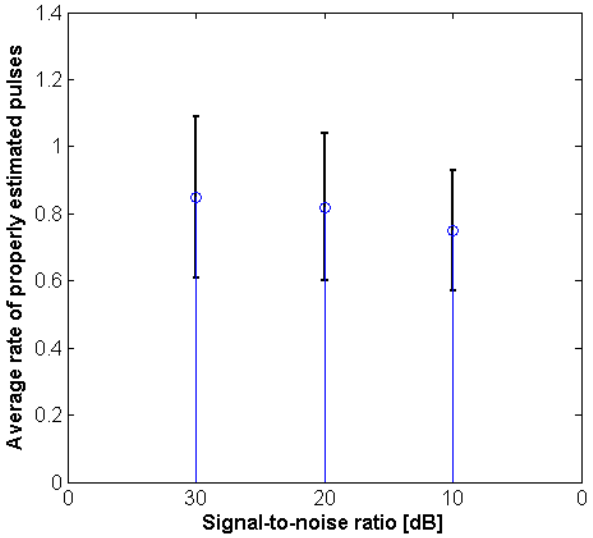

Fig. 10: Average rate and standard deviation of recognised pulses versus different SNR of 30, 20 and 10 dB. Zero-mean random white Gaussian noise was used.

Fig. 10 shows the average rate and standard deviation of properly placed pulses versus SNR do not aggravate significantly because of noise. Decompositions at 20 dB still have more than 90 % of properly placed pulses. Indeed, recognition rate decreases with the increased SNR, nevertheless the sCKC decomposition approach proves to be very robust.

6. Conclusion

In this paper, sequential and block-sequential CKC methods were introduced Both methods begin with a block of observations in order to initialise the correlation matrix inverse of observations and calculate the starting decomposition filters. After the initialisation, the sequential part comes into action, where two iterative operations play the main role: the update of correlation matrix inverse and the improvement of starting filters.

It was proven in previous work [5] that the best solution for updating the correlation matrix inverse is Sherman-Morrison formula when sample-by-sample updating is considered, and Sherman-Morrison-Woodbury formula when block updating is considered. The analysis of computational complexity in Subsection 4.1 reveals quadratic and/or cubic relationships, and the experiments in Subsection 4.2 back up the analytical findings. The relationship between correlation matrix size and input block size for the fastest sCKC is described by (20), proving that the block-sequential method is more computationally efficient than sample-by-sample approach.

We verified the sCKC decomposition performance on synthetic sEMG signals and also analysed its noise resistance. The experimental results show the average rate of the missed pulses is independent of the number of active MUs and the level of additive noise. Unfortunately, the same can not be said for the misplaced pulses; their average rate increases with the number of active MUs and the level of noise. The main reason for this can be attributed to less successful selection and improvement of decomposition filters.

References:

[1] A. Holobar, D. Zazula: *Multichannel Blind Source Separation Using Convolution Kernel Compensation,* IEEE Trans. on Sig. Proc., Vol. 55, pp. 4487-4496, 2007.

[2] A. Holobar: *Blind decomposition of convolutive mixtures of close-to-orthogonal pulse sources applied to surface electromyogram, PhD Thesiss,* FEECS, University of Maribor, 2004.

[3] V. Glaser, A. Holobar, D. Zazula: *An approach to the real-time surface electromyogram decomposition*, 11th Mediterrnanean Conference on

Medical and Biological Enginering and Computing 2007, Ljubljana, Slovenia, pp. 105-108.

[4] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, SIAM, Philadelphia, 2001.

[5] V. Glaser, *Sequential Convolution Kernel Composition for Composite Signal Decomposition, Diploma Thesis,* FERI, University of Maribor, 2006.

[6] R. Merletti, L. Lo Conte, E Avignone, P. Guglielminotti, *Modelling of surface EMG signal. Part I: model and implementation*, IEEE Trans BME 46, pp. 810-820, 1999

[7] R. Plonsey, *Action potential sources and their volume conductor fields*, IEEE Trans BME 56, pp. 601-611, 1977

[8] C. W. Therrien, *Discrete random signals and statistical signal processing*, Prentice-Hall, New Jersey, 1992

[9] A. Holobar, D. Zazula, *Correlation-based decomposition of surface electromyograms at low contraction forces*, Springer Berlin / Heidelberg, pp. 487-495, 2004

[10] L. Mesin, D. Farina, *A model for surface EMG generation in volume conductors with spherical inhomogeneities,* Biomedical Engineering, IEEE Transactions on, vol. 52, Issue 12, pp. 1984 – 1993, 2005

[11] L.Mesin, *Simulation of Surface EMG Signals for a Multilayer Volume Conductor With Triangular Model of the Muscle Tissue*, IEEE Transactions on vol. 53, Issue 11, pp. 2177 – 2184, 2006žđ

[12] J. H. Blok, D. F. Stegeman, A. van Oosterom, *Three-layer volume conductor model and software package for applications in surface electromyography*, Ann Biomed Eng 30: 566–577, 2002

[13] D. Farina, R. Merletti, *A novel approach for precise simulation of the EMG signal detected by surface electrodes*, IEEE Trans Biomed Eng 48: pp. 637–646, 2001