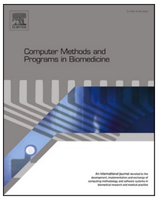
Computer Methods and Programs in Biomedicine 251 (2024) 108217

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.sciencedirect.com/journal/computer-methods-and-programs-in-biomedicine



# SynergyAnalyzer: A Matlab toolbox implementing mixed-matrix factorization to identify kinematic-muscular synergies

Marta Russo [1,2,a,b], Alessandro Scano [1,c,*], Cristina Brambilla [c], Andrea d'Avella [3,a,d]

[a] Laboratory of Neuromotor Physiology, IRCCS Fondazione Santa Lucia, Rome, Italy
[b] Deparment of Neurology, Fondazione Policlinico Tor Vergata, Rome, Italy
[c] Institute of Intelligent Industrial Systems and Technologies for Advanced Manufacturing (STIIMA), Italian Council of National Research (CNR), Via A.Corti 12, Milan, Italy
[d] Department of Biomedical and Dental Sciences and Morphofunctional Imaging, University of Messina, Messina, Italy

ARTICLE INFO

*Keywords:*
Mixed-matrix factorization
Non-negative matrix factorization
Kinematic-muscular synergies
Functional synergies
Muscle synergies
Matlab toolbox

ABSTRACT

*Background and Objective:* A new direction in the study of motor control was opened about two decades ago with the introduction of a model for the generation of motor commands as combination of muscle synergies. Muscle synergies provide a simple yet quantitative framework for analyzing the hierarchical and modular architecture of the human motor system. However, to gain insights on the functional role of muscle synergies, they should be related to the task space. The recently introduced mixed-matrix factorization (MMF) algorithm extends the standard approach for synergy extraction based on non-negative matrix factorization (NMF) allowing to factorize data constituted by a mixture of non-negative variables (e.g. EMGs) and unconstrained variables (e.g. kinematics, naturally including both positive and negative values). The kinematic-muscular synergies identified by MMF provide a direct link between muscle synergies and the task space. In this contribution, we support the adoption of MMF through a Matlab toolbox for the extraction of kinematic-muscular synergies and a set of practical guidelines to allow biomedical researchers and clinicians to exploit the potential of this novel approach.
*Methods:* MMF is implemented in the SynergyAnalyzer toolbox using an object-oriented approach. In addition to the MMF algorithm, the toolbox includes standard methods for synergy extraction (NMF and PCA), as well as methods for pre-processing EMG and kinematic data, and for plotting data and synergies.
*Results:* As an example of MMF application, kinematic-muscular synergies were extracted from EMG and kinematic data collected during reaching movements towards 8 targets on the sagittal plane. Instructions and command lines to achieve such results are illustrated in detail. The toolbox has been released as an open-source software on GitHub under the GNU General Public License.
*Conclusions:* Thanks to its ease of use and adaptability to a variety of datasets, SynergyAnalyzer will facilitate the adoption of MMF to extract kinematic-muscular synergies from mixed EMG and kinematic data, a useful approach in biomedical research to better understand and characterize the functional role of muscle synergies.

## 1. Introduction

To control body movements, the central nervous system (CNS) must coordinate a vast number of variables due to the redundant degrees of freedom of the musculoskeletal system [1]. About two decades ago, it has been proposed that the CNS might simplify motor control by generating motor commands through a linear combination of muscle synergies, each composed by a group of muscles that are activated together. Muscle synergies can be considered as an efficient and parsimonious way to generate spatiotemporal patterns of muscle activation by reducing the number of control variables required for motor coordination [2–8].

Muscle synergies opened a new direction in the study of motor control, by providing a simple yet quantitative framework for analyzing

---

* Corresponding author.
*E-mail address:* alessandro.scano@stiima.cnr.it (A. Scano).

[1] These authors contributed equally to this work.

[2] Current address: Institute of Cognitive Sciences and Technologies (ISTC), Italian Council of National Research (CNR), Rome, Italy

[3] Current address: Department of Biology, University of Rome Tor Vergata, Rome, Italy

https://doi.org/10.1016/j.cmpb.2024.108217
Received 13 December 2023; Received in revised form 24 April 2024; Accepted 6 May 2024

the hierarchical and modular structure of the motor system, based on neurophysiological evidences [3,5,6,9,10]. In particular, the initial approach consisted in using NMF [11] to identify muscle synergies from electromyographic (EMG) signals recorded from multiple muscles [3,5]. NMF allows to decompose the EMG signals, according to a spatial model, into spatial synergies and temporal coefficients, or, according to the dual temporal model, into temporal synergies and spatial loads [12]. Additional models and decomposition algorithms, such as spatiotemporal [6] and space-by-time [13], have also been introduced.

While muscle synergy analysis provides a parsimonious characterization of the organization of motor commands, synergies extracted from EMGs should be related to the task space variables to gain insights on how motor commands relate to task performance [14,15]. This problem has emerged both in the study of the physiological role of synergies [16] and in the evaluation of pathological synergistic patterns, which are characterized by specific changes in the synergies or temporal coefficients [17–21]. For all these scenarios, it is natural to investigate the intimate relationship that connects muscle synergies to their effect on task space variables. Thus, it comes naturally to extend the concept of muscle synergies to hybrid (or functional) synergies that include both muscle activities and task space variables into synergy weights.

The recently introduced mixed-matrix factorization (MMF, [22]) algorithm extends the standard approach to synergy identification using NMF by allowing to factorize a configurable number of unconstrained input data (i.e., potentially also including negative values), such as kinematic data, providing a direct link between muscle synergies and the task space. The MMF algorithm has been presented in detail in a previous contribution [22], where its performance was assessed on simulated data and shown to be superior to that of a state-of-the-art algorithm for factorizing non-constrained inputs [23].

The definition and extraction of synergies have been approached in various ways, incorporating different algorithms such as principal component analysis (PCA), non-negative matrix factorization (NMF [11]), and independent component analysis (ICA) applied to kinematic and EMG data [24–26]. Recently, Santuz introduced an R-based toolbox designed for extracting muscle synergies using NMF [27]. Notably, such toolbox also allows for synergy classification after extraction. On the other hand, Chiovetto and colleagues proposed a novel comprehensive framework based on the Fourier-based Anechoic Demixing Algorithm (FADA) and compared it with established model-specific algorithms to extract synergies. The relative Matlab toolbox (FADA-T) is shared implementing multiple synergy models [26,28]. However, a standardized approach and a toolbox that allows for extraction with different methods, is still missing. Moreover, a toolbox specifically designed for extracting functional synergies, such as the kinematic-muscular synergies identified using the MMF algorithm is not available. The MMF approach recently proved to be superior than previous attempts to extract synergies with positive and negative channels [22,23]. Given that, with the aim of supporting the adoption and standardization of kinematic-muscular synergies, we distribute a Matlab toolbox for synergy extraction (SynergyAnalyzer) that includes an implementation of MMF, together with standard methods for the extraction of kinematic and muscle synergies, separately. In addition, the toolbox allow for a standardized approach for processing EMG and kinematic data. The toolbox is available on GitHub with accompanying demonstrations. It is accessible to users with basic Matlab proficiency, and it is designed to adapt to a variety of algorithms and datasets.

Our toolbox adopts an object-oriented programming approach, which allows a clear definition of a few classes incorporating data structures and data processing methods commonly used in muscle synergy analysis, empowering users to easily extend and customize them to meet specific data analysis needs. This design approach reduces code redundancy and encapsulates the complexity of the underlying code, making it accessible to a wider audience, including biomedical researchers and clinicians without extensive programming experience.

To provide a comprehensive and stand-along tool for synergy extraction, the SynergyAnalyzer toolbox can handle various types of data (such as EMGs, kinematics, or a combination of both types), includes data preprocessing and visualization methods, and allows for synergy extraction using NMF [11] and principal component analysis (PCA) [29] in addition to the recently introduced MMF [22]. Filtering and averaging of EMG and kinematic data can be easily customized to match the requirements of specific datasets with minimal effort. Here, we use an example of extraction of kinematic-muscular synergies from combined kinematic and EMG data collected during reaching movements to provide a set of practical guidelines. In this way we hope to make MMF available to the biomedical community to foster research on synergistic models, which we foresee can expand already available applications in the field of rehabilitation [19], robot-assisted therapies [30], and understanding human-robot interaction [31].

## 2. Methods

*SynergyAnalyzer* includes all the functionalities for performing synergy analysis on different types of data. It is implemented as a MATLAB (Natick, USA; version required 7.6+) toolbox as MATLAB is commonly used by neuroscientists and biomedical engineers working on motor control. The toolbox incorporates functions for all the analysis steps, which include signal preprocessing (alignment and filtering), normalization, removal of tonic EMG components, and different methods for synergy extraction. It also provides data visualization methods. In this section, we first describe the organization and general functionalities of the toolbox and we then focus on the novel MMF synergy extraction algorithm. The toolbox is freely available on GitHub at this link under the GNU General Public License version 3.

### *2.1. Toolbox organization*

A general overview of the toolbox is shown in Fig. 1. Following an object-oriented programming approach, the toolbox consists of a top-level SynergyAnalyzer class, three data-related classes customized for specific types of data (EmgData for EMG data, KinData for kinematic data, and EmgKinData for combined EMG and kinematic data), and a Syn class for synergy extraction. Each data class includes methods to filter, align, normalize, and average the data in the preprocessing phase. The SynergyAnalyzer class uses the data-related classes to pre-process that data in data-specific manner and the Syn class to extract the synergies.

A sample dataset and two scripts with step-by-step examples for the extraction of muscle synergies using standard NMF (demo_nmf_emg.m) and of kinematic-muscular synergies using MMF (demo_mmf_mix.m) are also provided. The dataset is a subset of the dataset presented in [22] and includes EMGs (14 channels, shoulder and elbow muscles) and joint angles (4 channels, 3 shoulder joint and elbow joint angles) recorded during 10 repetitions of reaching movements towards each of 8 targets in the sagittal plane from one participant. This subset of data is included in the toolbox and can be downloaded from the same repository.

#### *2.1.1. Data-related classes*

Two basic data-related classes are implemented to process EMG and kinematic data, with data-specific methods. For example, the EmgData class includes a method for the rectification of the EMG signals. On the other hand, the KinData class includes a method for computing a numerical derivative of the position signal to obtain speed or acceleration. A third data-related EmgKinData class is implemented by merging EmgData and KinData and it allows for the extraction of kinematic-muscular synergies. The EmgKinData class includes EmgData and KinData objects as properties in order to re-use their methods.

Details about the experimental conditions and key events of a specific dataset that can be used to extract individual movement trials and to average over repeated trials in the same conditions must be provided using a user-defined function (getInfo). This function can be customized

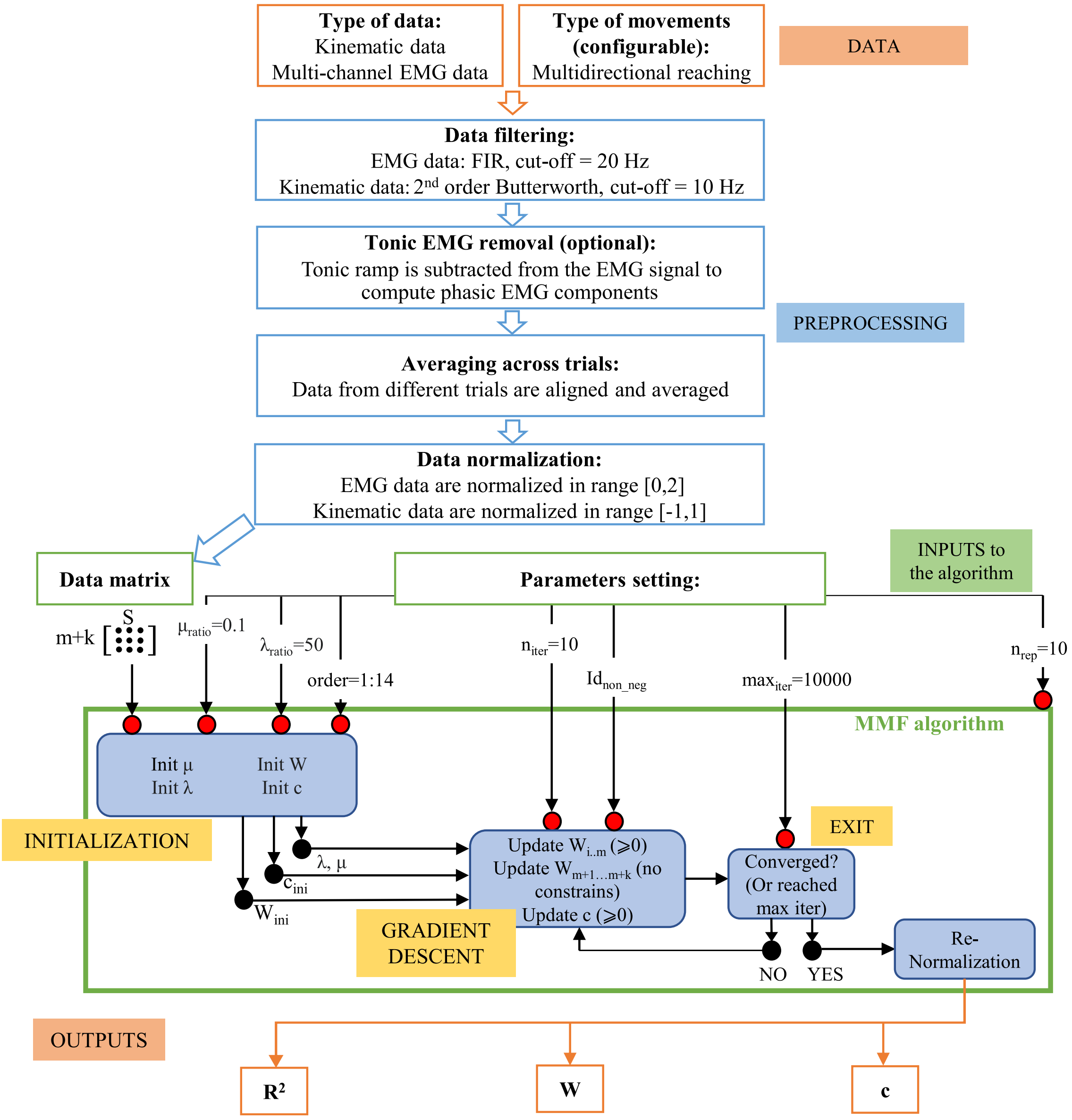


**Fig. 1.** Toolbox overview. All the steps taken to extract kinematic-muscular synergies are displayed in this flowchart. Data preprocessing, with customable parameters is the first step. Then, once the details for the extraction are set by the user, synergies can be extracted with MMF. The outputs are the quality of the reconstruction $R^2$ for each set of synergies extracted, the set of synergies W and the corresponding coefficients c.

to import a specific dataset into the data-related objects. In particular, the user can transform a set of data with any function, in order to get the data shaped as the inputs to the SynergyAnalyzer class constructor (see paragraph 3.1). More detailed information is available in the toolbox documentation and can be retrieved from the demo scripts provided.

### 2.1.2. *Preprocessing*

To extract synergies, raw EMG and kinematics data must be pre-processed, according to standard pipelines. For EMG signals, pre-processing includes rectification, filtering, subtraction of tonic activity (to extract phasic EMG components [32]), averaging, normalization, and resampling. For kinematic signals, pre-processing includes filtering, averaging, normalization, and resampling. For filtering, the toolbox provides three options: low-pass, high-pass, and band-stop (notch). For each filter, cut-off frequency, type of filter (e.g., finite impulse response or Butterworth) and filter-specific parameters (e.g. order) can be set by the user. Data can be time-shifted and aligned, according to specific events, before averaging. The maximum value of each channel or the maximum across all channels can be used for normalization. The outcome of pre-processing is a data matrix with channels as rows and samples for each condition as columns.

### 2.1.3. *Synergy extraction methods*

In addition to MMF (described in detail below), two standard methods for synergy extraction have been included in the toolbox: PCA and NMF. PCA is a standard linear technique for dimensionality reduction. The principal components are the eigenvectors of data covariance matrix. Such method has been previously used to extract kinematic synergies [29,33]. Given the inherent non-negativity of muscle activity, NMF has been widely used to identify muscle synergies [34]. An efficient iterative algorithm based on a multiplicative update rule has been introduced by Lee and Seung [11], and it allows to decompose a non-negative data matrix into the product of two non-negative matrices (synergies and coefficients). In summary, both methods allow to reconstruct a data matrix ***D*** with a matrix of synergies ***W*** multiplied by a matrix of coefficients ***C***:

$$D \approx WC \tag{1}$$

### 2.1.4. Visualization

The toolbox includes several visualization methods that allow to examine the data throughout the entire analysis process. Raw data can be plotted, including EMG and kinematics data aligned to the same temporal event. In the extraction phase, plotting functions have been implemented to visualize reconstruction error, synergy vectors, and synergy combination coefficients.

## 2.2. The MMF algorithm

The MMF algorithm is an extension of NMF allowing to factorize a data matrix mixing non-negative and unconstrained signals. As the *order of factorization* (i.e., the number of extracted synergies, $N$) is a free parameter, the MMF algorithm is typically called multiple times within a loop that increases $N$ from 1 to the number of rows of the data matrix $\boldsymbol{D}$. For each order of factorization, synergies and coefficients are randomly initialized and updated iteratively with a gradient descent algorithm. Denoting with $\boldsymbol{W}$ the synergies and with $\boldsymbol{C}$ the temporal coefficients, the update rule derived from the gradient of a cost function including the reconstruction error ($\|\boldsymbol{D} - \boldsymbol{WC}\|$) and the norm of $\boldsymbol{W}$ is the following:

$$W = W + 2\mu\left[(D - WC)C^T - \lambda W\right] \tag{2}$$

$$C = C + 2\mu\left[W^T(D - WC)\right] \tag{3}$$

with a learning rate $\mu = \mu_{ratio}/||\boldsymbol{D}||$ and a regularization weight $\lambda = \lambda_{ratio}/(N{*}sample)$ defined as in [22]. Nonnegativity is enforced by constraining some of the rows (e.g. those including EMG signals) to be positive or zero at each iteration of the optimization. The iterative updates stop when a convergence criterion is reached. The default criterion is 10 consecutive iterations for which the increase in reconstruction quality was smaller than the error convergence threshold. The reconstruction quality $R^2$ is defined as:

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{tr\left[(D - WC)(D - WC)^T\right]}{tr\left[(D - \overline{D})(D - \overline{D})^T\right]} \tag{4}$$

where $tr$ is the trace of the matrix. To minimize the probability of finding local minima, for each $N$, we repeated the optimization 10 times and selected the solution with the best reconstruction quality.

### 2.2.1. Description of the algorithm

Below we provide a brief explanation of the main part of the algorithm and a pseudocode to explain more in detail each step.

*Input and initializations.* First, we define the criteria for convergence (maximum error allowed to stop the algorithm, maximum number of iterations) and the variables μand λadapted to the characteristics of the dataset. We used the same values as in [22] with $\mu_{ratio} = 0.1$ and $\lambda_{ratio} = 50$. $W_{ini}$ and $C_{ini}$ are randomly generated at each iteration.

*Gradient descent algorithm.* The iterative optimization algorithm is repeated several times for each $N$. At each repetition, the algorithm updates $\boldsymbol{W}$ and $\boldsymbol{C}$ according to the gradient descent update rule described above. Non-negativity is constrained only on some rows of $\boldsymbol{W}$, while other rows are free to be either positive or negative. All $\boldsymbol{C}$ coefficients are non-negative as they represent synergy recruitment over time.

*Convergence and termination.* The algorithm stops when it converges to a stable solution, which is determined according to two termination criteria: either reaching a given data reconstruction error threshold or reaching a given maximum number of iterations. Parameters like the required error threshold and the number of iterations can be set before the extraction with a line of command, as shown below (see paragraph 3.2 for more details).

*Normalization and output.* $\boldsymbol{W}$ and $\boldsymbol{C}$ are normalized and returned as outputs. Since in the present formulation the algorithm extracts spatial kinematic-muscular synergies, each column of $\boldsymbol{W}$ is normalized by its norm. This makes synergies comparable across subjects or sessions. Each row of $\boldsymbol{C}$ is multiplied by the same norm to preserve the reconstruction of the original EMG/kinematic data.

*Pseudocode.*

```
Input:
  - D: Data matrix with mixed variables (time samples of kinematic and muscular data)
  - N: Number of synergies to extract
  - k: Number of unconstrained components in D
  - μ: Learning rate for W and C updates
  - λ: Regularization parameter
  - MaxIterations: Maximum number of iterations
  - ErrorConvergenceThreshold: Error threshold for convergence
  Initialization:
  - Initialize matrix W to uniform random values in [0 1] with dimensions ($k + m$) x N
  - Initialize matric C to uniform random values in [0 1] with dimensions N x number_of_samples (S)
  - Normalize columns of W to unit norm vectors
  Iteration:
  for iteration in 1 to MaxIterations:
        Update W:
$W = W + 2\mu\left[(D - WC)C^T - \lambda W\right]$
       Update C:
  $C = C + 2\mu\left[W^T(D - WC)\right]$
       Ensure non-negativity of non-negative rows of W and C by setting to zero negative values
       Normalize columns of W to unit vectors
       Calculate reconstruction quality to monitor convergence:
$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{tr\left[(D - WC)(D - WC)^T\right]}{tr\left[(D - \overline{D})(D - \overline{D})^T\right]}$
      If the increase of $R^2$ is less than ErrorConvergenceThreshold for 10 subsequent iterations, break the loop.
  Output:
  - Extracted spatial synergies W
  - Time-varying combination coefficients C
```

## 3. Results

As a demonstration of the usage of the SynergyAnalyzer toolbox for the extraction of kinematic-muscular synergies using MMF, this section illustrates all the steps involved in the analysis of a sample dataset of kinematic and EMG data collected during reaching movements.

## 3.1. Data import and preprocessing

The first step is the creation of SynergyAnalyzer object importing the data from the sample dataset included in the toolbox (“data.mat”). The SynergyAnalyzer class constructor requires as inputs 3 structures: data, par (parameters) and info. data is a structure with fields: emg (pos), emgtime (postime), which contains EMG (kinematics) data matrix and the corresponding time vector.

Importing requires a dataset-specific function (getInfo) to provide information about experimental conditions, events, and data. Info will be an input to the SynergyAnalyzer class constructor and requires the following fields:

```
info(i).id        % trial number
info(i).type      % could be a vector of n different types
info(i).selected    % indicates if the trial can be included in the analysis

info(i).events.code = [13,14]; % codes for the time events
info(i).events.time = [data(i).info.t_onset
data(i).info.t_end]; % times at which the events occurred
```

The getInfo function should be customized by the user to retrieve such information from any dataset. More detailed information is available in the toolbox documentation in Matlab and in the GitHub repository.

```
info = getInfo(data);
par.type = ‘EmgKinData’;
par.chlabels = [emgchannels(:)’, kinchannels(:)’];
par.order = 2;
par.delay = -0.05; %50 ms -> KIN data were anticipated of 50 ms to account for an estimated electromechanical delay between EMG and KIN (Scano et al. 2022)
sa = SynergyAnalyzer(data,info,par);
```

Parameters are explicitly set in this example to illustrate how they can be provided to the SynergyAnalyzer object. The Matlab documentation of the toolbox includes further details on the different options. In particular, the *type* field allows to create a data object suitable for the MMF extraction, *chlabels* indicates the name of each channel, *order* determines the derivative order of the kinematic signals, *delay* sets the time shift between the EMG and the kinematic signals to account for the electromechanical delay between EMG signal and muscle force production.

The next preprocessing step is filtering. In this example, the EMG signals, after rectification, are low-pass filtered using a finite-impulse response filter with a 20 Hz cutoff frequency and the kinematic signals are low-pass filtered with a 2nd order Butterworth filter with a 10 Hz cutoff frequency. EMG signals are also resampled every 10 ms. All these parameters can be modified using the *filter.type* and *filter.par* fields of the option structures (see the code below). The type of the filter and the cutoff frequency can be set depending on the sampling rate of the collected data.

```
sa.opt.emgFilter.type = ‘fir1’;
sa.opt.emgFilter.par = [50 20/(1000/2)]; %50 order 20 Hz @ 1KHz EMG sampling rate
sa.opt.emgFilter.resample = 1;
sa.opt.emgFilter.resample_period = .01; % resampling period [s]
sa.opt.kinFilter.type = ‘butter’;
sa.opt.kinFilter.par = [2 10/(100/2)]; %2 order 10 Hz @ 100 Hz sampling rate
saf = sa.dataFilter;
```

Next, the tonic component is subtracted from the EMG signals to maintain only the phasic component, as in [32].

```
saf = saf.emgPhasic;
```

Data are then averaged across reaching movement repetitions towards each of 8 targets (indicated by the ‘type3’ and [1:8] input pairs). To this end, data are first aligned to the movement onset time, and then the time interval from 300 ms before movement onset to 700 ms after it, as indicated by the *trange* parameter, is selected.

```
saf.opt.average.gr = saf.groupTrials(’type3’, [1:8]’); % type3 is the target number
saf.opt.average.trange = [-.3 .7]; % time interval in s before and after onset for averaging
sav = saf.average;
```

Next, data are normalized. Different normalization options are available, with normalization to the maximum absolute value of each channel as default. Since EMG and kinematic data have different ranges of values, in particular rectified EMGs are non-negative, while kinematic data may have both positive and negative values, to ensure that both types of data have the same normalized range, one option is to normalize the EMG signals to half of the maximum value so that the EMG data range is [0,2] and the kinematic data range is [−1,1].

```
sav.opt.normalize.type = 32;
sav.opt.normalize.nonnegch = sav.data(1).nonnegch;
sav = sav.normalize;
```

In Fig. 2 EMG and kinematic data after two pre-processing steps are shown, generated using the plot method of the EmgKinData class.

### 3.2. Synergy extraction

Once data have been pre-processed, synergies can be extracted by setting specific parameters. The toolbox provides multiple extraction algorithms; therefore, one must be specified:

```
sav.opt.find.algo = ‘mmf’;
```

Then, the range of number of synergies to extract (N) and the number of repetitions of synergy extractions for each number of synergies (nrep) must be set.

```
sav.opt.find.N = [1:14];
sav.opt.find.nrep = 10;
```

To allow the gradient descent algorithm to find the best solution for each iteration we used a convergence criterion of 10 consecutive iterations for which the increase in the data reconstruction quality was $<10^{-6}$. Also, the algorithm should run for at least 100 iterations and no more than 10,000.

```
sav.opt.find.niter = [ 100, % number of minimum iterations
10, % consecutive iterations below a threshold
10^-6, % error threshold
10,000]; % max number of iterations
```

Finally, the entire set of extractions can be started calling the *find* method of the SynergyAnalyzer class.

```
s1 = sav.find;
```

Fig. 3A shows the reconstruction quality ($R^2$) for different numbers of extracted synergies. This figure can be easily generated using the plot method of the SynergyAnalyzer class.

```
s1.opt.plot.type = ‘rsq’;
plot(s1)
```

Based on the $R^2$ curve, we select 6 synergies as the minimum number with $R^2$ above the threshold of 0.8 These are the default options implemented in the numsel function. These synergies are displayed in Fig. 3B, which can be generated with the following commands.

```
Nsel = numsel(s1.syn);
s1.opt.plot.N = Nsel;
s1.opt.plot.type = ‘W’;
plot(s1)
```

Alternatively, the user can choose the number of synergies N based on the detection of a “knee” (i.e. change in slope) in the curve of the $R^2$ [25,35]. To detect a change in slope in the $R^2$ curve, for each N, we implemented a linear fit of the portion of the curve from N to the end and the user can select N for which the mean square error of the fit is smaller than a threshold (for example $10^{-4}$). Please refer to the toolbox documentation for further details.

How the extracted spatial synergies and the associated temporal coefficients reconstruct the original signal is shown in Fig. 4. The thick black line indicates the reconstruction, the grey areas represent the original signals, as in previous figures. The coefficients at the bottom are colored coded as the corresponding synergy. To create Fig. 3 the command is the following.

```
s1.opt.plot.type = ‘rec’;
s1.opt.plot.isect = [1:8];
```

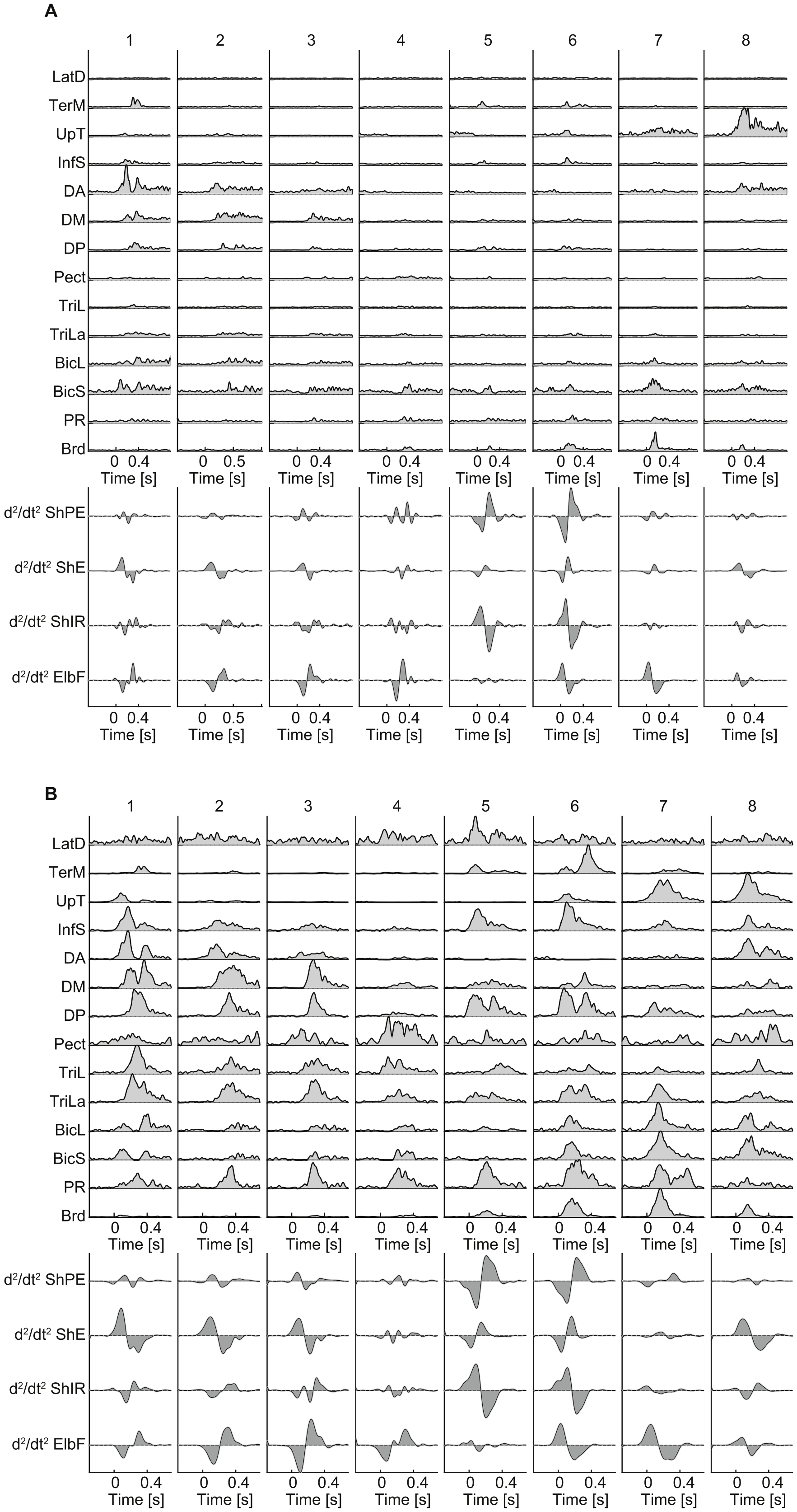


**Fig. 2.** A: EMG and kinematic data for one reaching movement for each of eight targets. EMG data are rectified and filtered, kinematic data are filtered and numerically differentiated to compute joint accelerations. B: EMG and kinematic data after averaging across repeated trials to the same target and normalization.

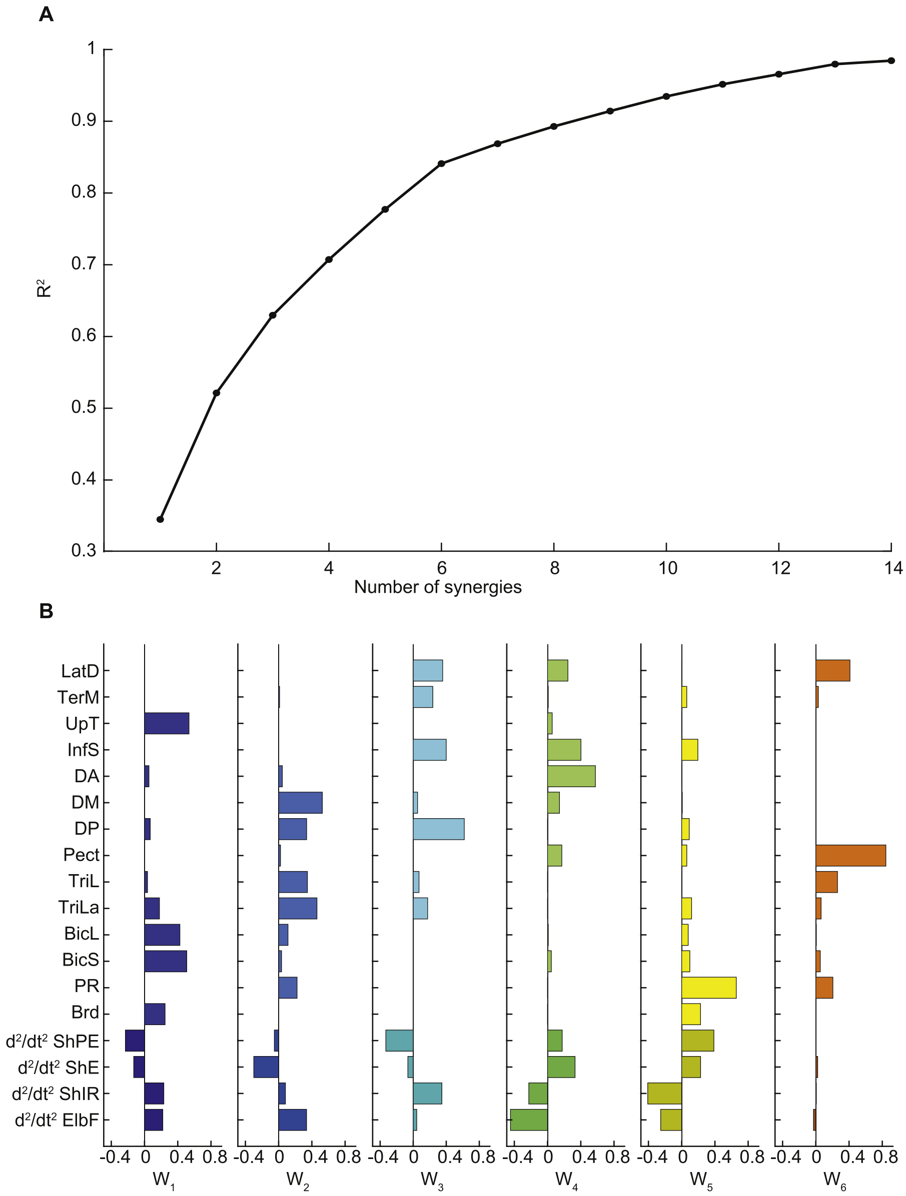


**Fig. 3.** A: $R^2$ curve for kinematic-muscular synergies. B: extracted kinematic-muscular synergies.

```
plot(s1)
```

The synergies are similar to those previously extracted from the same dataset [22]. However, some slight differences may arise due to the different choices made for filtering and for the convergence criteria that have been updated in this toolbox.

## 4. Discussion

SynergyAnalyzer is a novel and comprehensive toolbox for synergy extraction from EMG and kinematic data. It includes standard synergy extraction methods for EMG data (NMF) and kinematic data (PCA) as well as the recently introduced MMF methods for combined EMG and kinematic data. The toolbox architecture is object-oriented, and it includes preprocessing and visualization functions, allowing to perform all steps required for synergy analysis. Below, we provide some comments and guidelines for using the toolbox.

### *4.1. Algorithms for synergy extraction*

Muscle activation patterns have been reconstructed as the combination of a small number of synergies. These synergies can be obtained using several decomposition techniques, such as PCA, factor analysis (FA), independent component analysis (ICA). However the vast majority of the studies in the field uses non-negative matrix factorization (NMF) [11]. Therefore, we have implemented NMF in the toolbox for muscle synergies extraction. Additionally, to allow for extracting kinematic synergies, we also implemented PCA, which is a linear method that can be applied to unconstrained data, i.e. with both positive and negative values. However, PCA generates components that are orthogonal to each other, which is not required for physiological synergies. MMF is a novel extension of NMF that allows to extract synergies with negative loads, needed when kinematics signals are combined with EMG signals as inputs. This approach can also be extended to phasic muscle synergies with negative components [36] and to any other multimodal approach. Therefore, we focused the present demonstration on MMF.

In addition to the spatial synergy models, several other decomposition have been considered in previous studies. By reformatting data matrix appropriately, it is possible to extract temporal synergies [12,29,37], spatiotemporal or time-varying synergies [38], space-by-time synergies [13]. The object-oriented architecture of the toolbox allows to

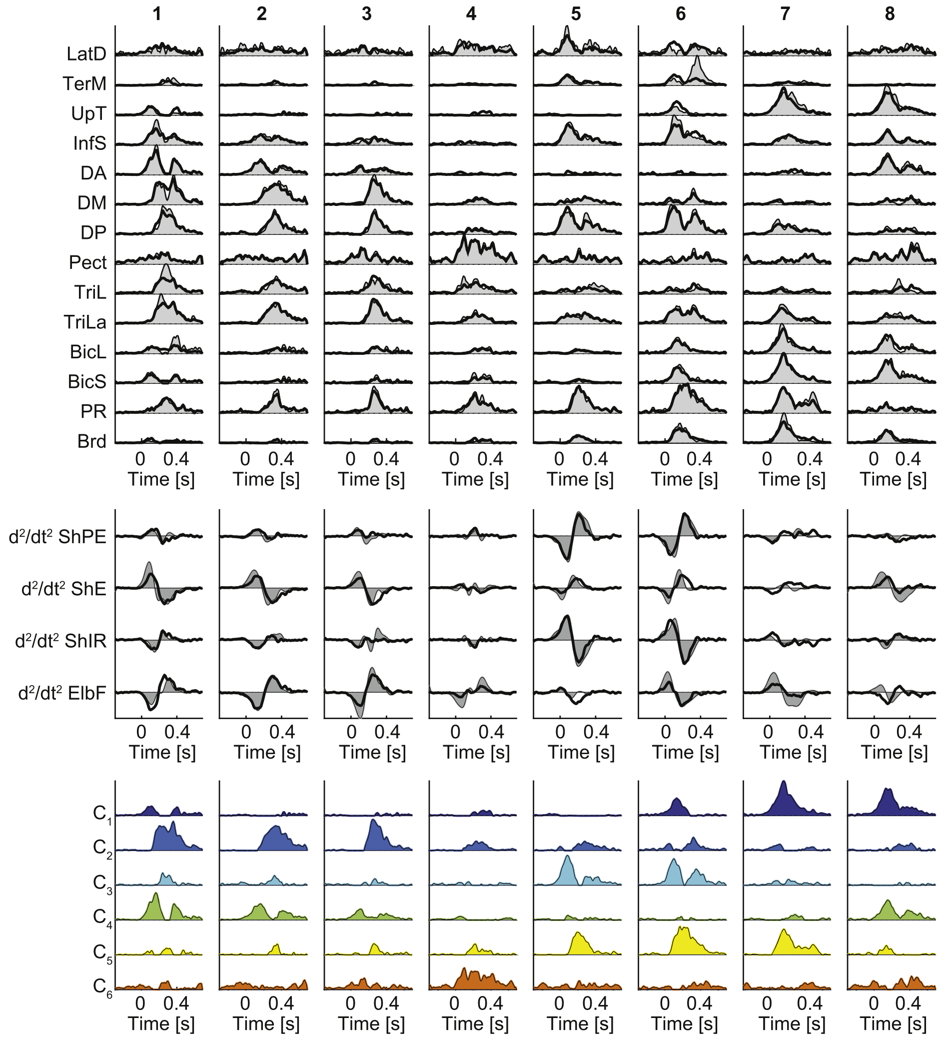


**Fig. 4.** EMG and kinematic data reconstruction; top panel: EMG; middle panel: joint accelerations; lower panel: temporal coefficients of the kinematic-muscular synergies.

easily extend it to include these additional extraction methods and future releases of the toolbox might implement them.

### 4.2. Limitations

The first limitation of the toolbox is the use of a commercial computing platform to implement the toolbox, i.e. Matlab. However, Matlab is commonly used in the motor control community and many universities have licenses for their students. Future releases of the toolbox will be tested and customized for use with GNU Octave, a free computing platform largely compatible with Matlab. In addition, while a Graphical User Interface (GUI) could enhance accessibility, adapting synergy extraction for diverse datasets or actions might necessitate custom code, potentially challenging to implement in a GUI. Nonetheless, we offer comprehensive preprocessing and visualization functions, aiding less experienced users in familiarizing themselves with synergy extraction. Standardizing the extraction procedure and the entire pipeline can be facilitated by employing consistent preprocessing methods, even for not experienced users. The toolbox is also missing a procedure to cluster synergies extracted from multiple datasets (e.g. different participants or experimental conditions). Nevertheless, the scope of the present contribution is to provide users with standardized methods for data processing and extraction, giving the freedom to interpret synergies according to each specific scenario. Finally, the toolbox implements the novel extraction method MMF, which already represents a novel contribution to the field and its implementation has never been shared. With this in mind, we aim at promoting the extraction of functional synergies as a significant and compact approach to investigate motor behavior.

### 4.3. Tuning of MMF parameters

In the MMF algorithm, the learning rate (μ) and regularization weight (λ) parameters must be tuned for each specific dataset. Increasing μ fastens the convergence of the algorithm but might prevent reaching the optimal solution. It is thus recommended to tune this parameter by keeping it at the highest possible value that in the specific dataset ensures obtaining a stable solution across multiple runs. λ avoids unplausible solutions (e.g., synergies that generate cancellations in the unconstrained channels; see [22] for details). The value of λ should be optimized trading off between high values useful to generate meaningful synergies and low values ensuring accurate reconstruction (high $R^2$) of the original data. The algorithm already attempts to adapt μ and λ to the specific dataset size by normalizing them using the Frobenius norm of the data matrix. We have tested this normalization in few cases, but we recommend performing a calibration procedure to determine reasonable regions of use for μ and λ according to the properties of the specific dataset.

### 4.4. EMG and kinematic data normalization

Normalization of EMG envelopes is needed to allow inter-subject and inter-session comparisons, and to account for muscles that intrinsically produce less EMG activity due to their properties. Normalization can be done according to several approaches. Torres-Oviedo and collaborators normalized EMG signals in order to have unit variance [23,39]. However, the most frequent approach is to normalize all EMG samples with respect to the maximum EMG value found in all the dataset for each muscle. This approach has been commonly used in upper-limb studies [32], and is particularly effective when many trials in different conditions are available. Regarding kinematics, the same approach is recommended, i.e., normalizing each joint angular acceleration by the maximum for that joint across all trials. For both EMGs and kinematics the toolbox implements different types of normalization, i.e. to the maximum of each channel or of all channels. With this approach EMG activity ranges between 0 and 1, and kinematic activity ranges between $-1$ and $+1$. However, for MMF is desirable that the normalization guarantees a balance between kinematic and muscle contributions. Thus, one option is to rescale EMGs between 0 and 2, i.e. to have the same range as kinematics, and this normalization option has been implemented in the toolbox. Further works should specify what normalization is more suitable to link the two domains.

### 4.5. Movements and data collection

In the provided example, we derived synergies from a collection of reaching data. To optimize the algorithm's effectiveness, it is crucial to format the data with certain precautions. The following suggestions are primarily tailored for comparable datasets. Data from different motor behaviors, e.g. walking, might require slightly different configurations. Nevertheless, the toolbox's modular design empowers users to adapt the synergy analysis to the distinctive features of each dataset through the getInfo function. This flexibility enables the utilization of various parameters to fine-tune the algorithm according to the provided data. Detailed specifications are indicated in the toolbox available on the GitHub repository.

First, it is recommended that the data samples include a rest period prior to the movement to estimate baseline EMG. Thus, it is suggested to introduce pauses between movements. In fact, the toolbox removes tonic components from the whole EMG, as in [32,40] and the signal should be stationary in such time intervals. Moreover, the conditions for each movement trial (e.g., different targets and different planes where the targets are arranged) must be specified before synergy extraction, using the getInfo function.

Data from trials in different reaching directions [41] where average and concatenated. This allowed to achieve higher SNR with respect to concatenating all individual trials without averaging. However, depending on the specific research question, different data structures can be chosen; synergy extraction will simply reflect such choices. The toolbox does not impose any constraint.

As the EMG signal anticipates force output, due to the dynamics of muscle contraction, to capture EMG-kinematic causal relationship when extracting spatial synergies, one might choose to delay EMG data, or anticipate kinematic ones. Typical electromechanical delay values used in the literature range from 30 ms to 100 ms [42] and might be tuned depending on the application. Here we chose a delay of 50 ms as in [22].

MMF was conceived as an extension of NMF; most of the pipelines for applications are identical, even though MMF requires the tuning of some variables and few additional inputs parameters. Any kind of movement data can be factorized with MMF. We suggest gathering EMG data as it would be normally done with NMF for muscle input. For kinematics, we suggest using data from joint angular accelerations as they should be approximated reasonably well by a linear relationship between the EMG and the joint motion. However, muscle data can be virtually coupled with any kind of multimodal data, including joint displacement, velocity, cartesian 3D movement, or data from other domains. In the original formulation (EMG and acceleration data), we expected normal or fast movement to work better as acceleration profiles are more pronounced and visible. In our formulation, we also removed tonic activity [32,40,43] to better link motion only to the phasic EMG that generated it; fast motion helps such EMG patterns to be more visible. Tonic EMG removal is in any case recommended for upper-limb applications, even if it is not mandatory [43].

## 5. Conclusion

In this work, we introduced a novel toolbox for synergy extraction, implementing standard factorization algorithms such as NMF and PCA but also the recently introduced MMF algorithm. We hope that this contribution will help researchers in the study of motor control and in various fields and settings, fostering novel research based on the evolution of muscle synergies to the concepts of functional and task-related synergies. This approach might provide a step forward to the understanding of underlying mechanisms of motor control towards practical applications.

## Ethics approval and consent to participate

Not applicable.

## Consent for publication

Not applicable.

## Availability of data and materials

All data and materials are available on GitHub: github.com/SynergyAnalyzer/SynergyAnalyzerToolbox.git

## CRediT authorship contribution statement

**Marta Russo:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis. **Alessandro Scano:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Cristina Brambilla:** Writing – review & editing, Visualization, Validation, Methodology, Formal analysis, Data curation. **Andrea d'Avella:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Project administration, Methodology, Conceptualization.

## Declaration of competing interest

Authors declare no competing interests.

### Funding

This work was supported in part by the Italian Ministry of Research, under the complementary actions to the Plan of National Recovery and Resilience (PNRR) "Fit4MedRob-Fit for Medical Robotics" Grant (# PNC0000007), and in part by the Starting Grant of the Italian Ministry of Health SG-2018–12366101 - RF 2018.

### Acknowledgements

The authors acknowledge the "Fit4MedRob-Fit for Medical Robotics" Grant (#PNC0000007) under the complementary actions to the Italian Plan of National Recovery and Resilience (PNRR).

## References

[1] N.A. Bernstein, The Co-Ordination and Regulation of Movements, 1st English ed., Pergamon Press, Oxford, New York, 1967.

[2] E. Bizzi, F.A. Mussa-Ivaldi, S. Giszter, Computations underlying the execution of movement: a biological perspective, Science 253 (1991) 287–291, https://doi.org/10.1126/science.1857964.

[3] M.C. Tresch, P. Saltiel, E. Bizzi, The construction of movement by the spinal cord, Nat. Neurosci. 2 (1999) 162–167, https://doi.org/10.1038/5721.

[4] P. Saltiel, K. Wyler-Duda, A. D'Avella, M.C. Tresch, E. Bizzi, Muscle synergies encoded within the spinal cord: evidence from focal intraspinal nmda iontophoresis in the frog, J. Neurophysiol. 85 (2001) 605–619, https://doi.org/10.1152/jn.2001.85.2.605.

[5] L.H. Ting, J.M. Macpherson, A limited set of muscle synergies for force control during a postural task, J. Neurophysiol. 93 (2005) 609–613, https://doi.org/10.1152/jn.00681.2004.

[6] A. d'Avella, P. Saltiel, E. Bizzi, Combinations of muscle synergies in the construction of a natural motor behavior, Nat. Neurosci. 6 (2003) 300–308, https://doi.org/10.1038/nn1010.

[7] Y.P. Ivanenko, R.E. Poppele, F. Lacquaniti, Five basic muscle activation patterns account for muscle activity during human locomotion, J. Physiol. 556 (2004) 267–282, https://doi.org/10.1113/jphysiol.2003.057174.

[8] M.C. Tresch, A. Jarc, The case for and against muscle synergies, Curr. Opin. Neurobiol. 19 (2009) 601–607, https://doi.org/10.1016/j.conb.2009.09.002.

[9] C.B. Hart, S.F. Giszter, A Neural Basis for Motor Primitives in the Spinal Cord, J. Neurosci. 30 (2010) 1322–1336, https://doi.org/10.1523/JNEUROSCI.5894-08.2010.

[10] W.J. Kargo, S.F. Giszter, Individual premotor drive pulses, not time-varying synergies, are the units of adjustment for limb trajectories constructed in spinal cord, J. Neurosci. 28 (2008) 2409–2425, https://doi.org/10.1523/JNEUROSCI.3229-07.2008.

[11] D.D. Lee, H.S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791, https://doi.org/10.1038/44565.

[12] C. Brambilla, M. Atzori, H. Müller, A. d'Avella, A. Scano, Spatial and temporal muscle synergies provide a dual characterization of low-dimensional and intermittent control of upper-limb movements, Neuroscience 514 (2023) 100–122, https://doi.org/10.1016/j.neuroscience.2023.01.017.

[13] I. Delis, S. Panzeri, T. Pozzo, B. Berret, A unifying model of concurrent spatial and temporal modularity in muscle activity, J. Neurophysiol. 111 (2014) 675–693, https://doi.org/10.1152/jn.00245.2013.

[14] R. Leib, M. Russo, A. d'Avella, I. Nisky, A bang-bang control model predicts the triphasic muscles activity during hand reaching, J. Neurophysiol. 124 (2020) 295–304, https://doi.org/10.1152/jn.00132.2020.

[15] C. Alessandro, I. Delis, F. Nori, S. Panzeri, B. Berret, Muscle synergies in neuroscience and robotics: from input-space to task-space perspectives, Front. Comput. Neurosci. 7 (2013) 43, https://doi.org/10.3389/fncom.2013.00043.

[16] D. O'Reilly, I. Delis, Dissecting muscle synergies in the task space, (2023) 2023.03.17.533096. 10.1101/2023.03.17.533096.

[17] G. Cappellini, Y.P. Ivanenko, G. Martino, M.J. MacLellan, A. Sacco, D. Morelli, F. Lacquaniti, Immature spinal locomotor output in children with cerebral palsy, Front. Physiol. 7 (2016). www.frontiersin.org/articles/10.3389/fphys.2016.00478. accessed October 19, 2023.

[18] G. Martino, Y. Ivanenko, M. Serrao, A. Ranavolo, F. Draicchio, C. Casali, F. Lacquaniti, Locomotor coordination in patients with Hereditary Spastic Paraplegia, J. Electromyogr. Kinesiol. 45 (2019) 61–69, https://doi.org/10.1016/j.jelekin.2019.02.006.

[19] V.C.K. Cheung, L. Piron, M. Agostini, S. Silvoni, A. Turolla, E. Bizzi, Stability of muscle synergies for voluntary actions after cortical stroke in humans, Proc. Natl. Acad. Sci. U. S. A. 106 (2009) 19563–19568, https://doi.org/10.1073/pnas.0910114106.

[20] V.C.K. Cheung, A. Turolla, M. Agostini, S. Silvoni, C. Bennis, P. Kasi, S. Paganoni, P. Bonato, E. Bizzi, Muscle synergy patterns as physiological markers of motor cortical damage, Proc. Natl. Acad. Sci. U. S. A. 109 (2012) 14652–14656, https://doi.org/10.1073/pnas.1212056109.

[21] D.J. Berger, F. Ferrari, A. Esposito, M. Masciullo, M. Molinari, F. Lacquaniti, A. d'Avella, J. Ibáñez, J. González-Vargas, J.M. Azorín, M. Akay, J.L. Pons, Changes in muscle synergy organization after neurological lesions. Converging Clin. Eng. Res. Neurorehabilitation II, Springer International Publishing, Cham, 2017, pp. 939–943, https://doi.org/10.1007/978-3-319-46669-9_152.

[22] A. Scano, R.M. Mira, A. d'Avella, Mixed matrix factorization: a novel algorithm for the extraction of kinematic-muscular synergies, J. Neurophysiol. 127 (2022) 529–547, https://doi.org/10.1152/jn.00379.2021.

[23] G. Torres-Oviedo, J.M. Macpherson, L.H. Ting, Muscle synergy organization is robust across a variety of postural perturbations, J. Neurophysiol. 96 (2006) 1530–1546, https://doi.org/10.1152/jn.00810.2005.

[24] S.F. Giszter, Motor primitives–new data and future questions, Curr. Opin. Neurobiol. 33 (2015) 156–165, https://doi.org/10.1016/j.conb.2015.04.004.

[25] M.C. Tresch, V.C.K. Cheung, A. d'Avella, Matrix factorization algorithms for the identification of muscle synergies: evaluation on simulated and experimental data sets, J. Neurophysiol. 95 (2006) 2199–2212, https://doi.org/10.1152/jn.00222.2005.

[26] E. Chiovetto, M.A. Giese, Kinematics of the coordination of pointing during locomotion, PLoS ONE 8 (2013) e79555, https://doi.org/10.1371/journal.pone.0079555.

[27] A. Santuz, musclesyneRgies: factorization of electromyographic data in R with sensible defaults, J. Open Source Softw. 7 (2022) 4439, https://doi.org/10.21105/joss.04439.

[28] E. Chiovetto, A. Salatiello, A. d'Avella, M.A. Giese, Toward a unifying framework for the modeling and identification of motor primitives, Front. Comput. Neurosci. 16 (2022) 926345, https://doi.org/10.3389/fncom.2022.926345.

[29] M. Russo, M. D'Andola, A. Portone, F. Lacquaniti, A. d'Avella, Dimensionality of joint torques and muscle patterns for reaching, Front. Comput. Neurosci. 8 (2014), https://doi.org/10.3389/fncom.2014.00024.

[30] T. Lencioni, L. Fornia, T. Bowman, A. Marzegan, A. Caronni, A. Turolla, J. Jonsdottir, I. Carpinella, M. Ferrarin, A randomized controlled trial on the effects induced by robot-assisted and usual-care rehabilitation on upper limb muscle synergies in post-stroke subjects, Sci. Rep. 11 (2021) 5323, https://doi.org/10.1038/s41598-021-84536-8.

[31] A. Cancrini, P. Baitelli, M.L. Nicora, M. Malosio, A. Pedrocchi, A. Scano, The effects of robotic assistance on upper limb spatial muscle synergies in healthy people during planar upper-limb training, PLoS ONE 17 (2022) e0272813, https://doi.org/10.1371/journal.pone.0272813.

[32] A. d'Avella, A. Portone, L. Fernandez, F. Lacquaniti, Control of fast-reaching movements by muscle synergy combinations, J. Neurosci. 26 (2006) 7791–7810, https://doi.org/10.1523/JNEUROSCI.0830-06.2006.

[33] M. Santello, M. Flanders, J.F. Soechting, Postural hand synergies for tool use, J. Neurosci. 18 (1998) 10105–10115, https://doi.org/10.1523/JNEUROSCI.18-23-10105.1998.

[34] K. Zhao, H. Wen, Z. Zhang, M. Atzori, H. Müller, Z. Xie, A. Scano, Evaluation of methods for the extraction of spatial muscle synergies, Front. Neurosci. 16 (2022). www.frontiersin.org/articles/10.3389/fnins.2022.732156. accessed October 24, 2023.

[35] S. Ranaldi, C. De Marchis, G. Severini, S. Conforto, An objective, information-based approach for selecting the number of muscle synergies to be extracted via non-negative matrix factorization, IEEE Trans. Neural Syst. Rehabil. Eng. 29 (2021) 2676–2683, https://doi.org/10.1109/TNSRE.2021.3134763.

[36] A. Scano, C. Brambilla, M. Russo, A. d'Avella, Upper limb phasic muscle synergies with negative weightings: applications for rehabilitation, (2023). 10.36227/techrxiv.22795553.v1.

[37] N. Dominici, Y.P. Ivanenko, G. Cappellini, A. d'Avella, V. Mondì, M. Cicchese, A. Fabiano, T. Silei, A. Di Paolo, C. Giannini, R.E. Poppele, F. Lacquaniti, Locomotor primitives in newborn babies and their development, Science 334 (2011) 997–999, https://doi.org/10.1126/science.1210617.

[38] A. D'Avella, M.C. Tresch, Modularity in the motor system: 15th annual neural information processing systems conference, NIPS 2001, in: Proceedings of the Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, 2002. www.scopus.com/inward/record.url?scp=84898937182&partnerID=8YFLogxK. accessed October 19, 2023.

[39] G. Torres-Oviedo, L.H. Ting, Subject-specific muscle synergies in human balance control are consistent across different biomechanical contexts, J. Neurophysiol. 103 (2010) 3084–3098, https://doi.org/10.1152/jn.00960.2009.

[40] A. Scano, L. Dardari, F. Molteni, H. Giberti, L.M. Tosatti, A. d'Avella, A comprehensive spatial mapping of muscle synergies in highly variable upper-limb movements of healthy subjects, Front. Physiol. 10 (2019). www.frontiersin.org/articles/10.3389/fphys.2019.01231. accessed October 19, 2023.

[41] A.S. Oliveira, L. Gizzi, D. Farina, U.G. Kersting, Motor modules of human locomotion: influence of EMG averaging, concatenation, and number of step cycles, Front. Hum. Neurosci. 8 (2014). www.frontiersin.org/articles/10.3389/fnhum.2014.00335. accessed October 19, 2023.

[42] H. Begovic, G.-Q. Zhou, T. Li, Y. Wang, Y.-P. Zheng, Detection of the electromechanical delay and its components during voluntary isometric contraction of the quadriceps femoris muscle, Front. Physiol. 5 (2014). www.frontiersin.org/articles/10.3389/fphys.2014.00494. accessed October 19, 2023.

[43] C. Brambilla, M. Russo, A. d'Avella, A. Scano, Phasic and tonic muscle synergies are different in number, structure and sparseness, Hum. Mov. Sci. 92 (2023) 103148, https://doi.org/10.1016/j.humov.2023.103148.