Mimicking human neuronal pathways in silico: an emergent model on the effective connectivity

Önder Gürcan, Kemal Türker, Jean-Pierre Mano, Carole Bernon, Oguz Dikenelli, Pierre Glize

Mimicking human neuronal pathways in silico: an emergent model on the effective connectivity

**Önder Gürcan · Kemal S. Türker · Jean-Pierre Mano ·
Carole Bernon · Oğuz Dikenelli · Pierre Glize**

Abstract We present a novel computational model that detects temporal configurations of a given human neuronal pathway and constructs its artificial replication. This poses a great challenge since direct recordings from individual neurons are impossible in the human central nervous system and therefore the underlying neuronal pathway has to be considered as a *black box*. For tackling this challenge, we used a branch of complex systems modeling called artificial self-organization in which large sets of software entities interacting locally give rise to bottom-up collective behaviors. The result is an emergent model where each software entity represents an integrate-and-fire neuron. We then applied the model to the reflex responses of single motor units obtained from conscious human subjects. Experimental results show that the model recovers functionality of real human neuronal pathways by comparing it to appropriate surrogate data. What makes the model promising is the fact that, to the best of our knowledge, it is the first realistic model to self-wire an artificial neuronal network by efficiently combining neuroscience with artificial self-organization. Although there is no evidence yet of the model's connectivity mapping onto the human connectivity, we anticipate this model will help neuroscientists to learn much more about human neuronal networks, and could also be used for predicting hypotheses to lead future experiments.

Ö. Gürcan (✉) · O. Dikenelli
Computer Engineering Department, Ege University,
Universite cad. 35100 Bornova, Izmir, Turkey
e-mail: onder.gurcan@gmail.com

O. Dikenelli
e-mail: oguz.dikenelli@ege.edu.tr

K. S. Türker
School of Medicine, Koc University,
Rumelifeneri Yolu, 34450 Sariyer, Istanbul, Turkey
e-mail: kturker@ku.edu.tr

Ö. Gürcan · C. Bernon · P. Glize
IRIT Laboratory, Universite Paul Sabatier,
118 Route de Narbonne, Toulouse, France

C. Bernon
e-mail: carole.bernon@irit.fr

P. Glize
e-mail: pierre.glize@irit.fr

J. P. Mano
Upetec 10 Avenue de l'Europe, 31520
Ramonville-Saint-Agne, France
e-mail: jean-pierre.mano@upetec.fr

1 Introduction

For decades, scientists have dreamed of building computer systems that could replicate the human neuronal system. Computational neuroscience is a major subdiscipline of neuroscience which is used for reaching this goal by integrating theoretical and experimental information using simulation and mathematical theory. Broadly, this integration can be done in two different ways (Gerstner et al. 2012; Dayan and Abbott 2005): integrating what is known on a micro-level (e.g., properties of ion channels, see (Colquhoun and Hawkes 1990)) to explain phenomena

observed on a macro-level (e.g., generation of action potentials, see (Hodgkin and Huxley 1952; Koch and Segev 1989; Mainen et al. 1995; Hay et al. 2011)) – namely *bottom-up approaches*, or starting with macro-level functions of the nervous system (e.g. working memory) and deduce from them how micro-level components need to behave in order to achieve these functions (e.g., neurons or groups of neurons) – namely *top-down approaches*. Both *bottom-up* and *top-down approaches* increased our understanding especially at the microscopic scales, however, the transition from microscopic to macroscopic scales still relies on mathematical arguments (Gerstner et al. 2012). Thus, simply "building" a human neuronal system from bottom-up by replicating its parts, connections, and organization fails to capture its macro-level complex behavior (Izhikevich and Edelman 2008). Hence, it is still unknown which microscopic rules under which conditions are useful at the macroscopic level. Studies for filling this gap are still missing.

In this study, our aim is to fill this gap by mimicking human neuronal pathways without assuming the transition from microscopic to macroscopic scales depend upon mathematical arguments (Gürcan et al. 2010a, b). Human neuronal pathways are natural *complex systems* in which large sets of neurons interact locally and give bottom-up rise to collective macroscopic behaviors. In this sense, correct knowledge of the synaptic functional connections between neurons is a key prerequisite for relating them to the operation of their central nervous system (CNS). However, estimating these functional connections between neurons in the human CNS poses a great challenge since direct recordings are impossible. Consequently, the network between human neurons is often expressed as a *black box* and the properties of connections between neurons are estimated using indirect methods (Türker and Powers 2005). In indirect methods a particular receptor system is stimulated and the responses of neurons that are affected by the stimulus recorded to estimate the properties of the circuit (Rossi et al. 2003; Misiaszek 2003; Grande and Cafarelli 2003). However, these neuronal circuits in human subjects are only estimations and their existence cannot be directly proven. Furthermore, there is no satisfactory theory on how these unknown parts of the CNS operate. Therefore, we need techniques that are able to learn inside this *black box* preserving both microscopic and macroscopic properties as biologically plausible as possible. However, success of such a technique would not be possible without an efficient effective connectivity[1] analysis method that successfully captures some of the essential functional principles of biological behaviors. Once we establish correctly how neurons are connected, by using the benefits of the resulting complex network representation and its spatially and temporally correlated signals, we can extract features that may be informative for the underlying dynamic processes. In this way, we can increase our understanding about the modulation of synaptic input during movement and learning, and also we can find new methods for diagnoses and treatment of disorders of the CNS.

Standing on these observations, we propose a novel computational model[2] based on the principles of *artificial self-organizing systems*. The goal of this model is to integrate knowledge from neuroscience and *artificial self-organization* to derive from it the fundamental principles that govern CNS function and its simulation, and ultimately, to reconstruct the human CNS pathways in silico. Self-organization essentially refers to a spontaneous, dynamically produced organization in a system without any external control (Serugendo et al. 2011). From the behavior of dynamically evolving natural systems perspective,[3] self-organization is a set of dynamical mechanisms whereby (spatial, temporal and/or functional) structures appear at the macro-level of a system as a result of interactions among its micro-level components (Bonabeau et al. 1999). The rules specifying these interactions are executed on the basis of purely local information, without reference to any macro-level pattern. Considering these definitions, in computer science, the term *artificial self-organization* refers to a process enabling a software system to dynamically alter its internal organization (structure and functionality) during its execution time without any explicit external directing mechanism (Serugendo et al. 2011).

The artificially self-organizing computational model proposed here uses temporal data collected from human subjects as an emergent macro-level description of the underlying neuronal pathway. Dynamic activity and spiking are modeled at the individual neuron (cell) scale. This scale was chosen since it represents the best compromise between dynamics, complexity and observability for simulating the effective connectivity of neuronal networks (Buibas and Silva 2011).[4] Consequently, the local information in the model is the knowledge about the behavior of individual neurons, such as generation of spikes and transmission of these spikes to their postsynaptic neurons.

---

[1]The actual coupling that we are trying to model is referred to as *effective connectivity*. In some studies the term *functional connectivity* is also used but this term more refers to the statistical correlations between nodes (for a review see (Friston 2011)). In this sense, we used the term *effective connectivity* throughout this article.

[2]Part of this work was previously published in a conference proceeding (Gürcan et al. 2012).

[3]There are various definitions of the concept of self-organization from different perspectives and disciplines. For a list of most common definitions in the literature see Section 3.2.2 in (Serugendo et al. 2011).

[4]However, it is still unknown what level of biological detail is needed in order to mimic the way CNS behaves.

The effect of a spike on a target neuron is defined as a temporal membrane potential change in response to the influence of a source neuron that connects to it. That influence is not instantaneous, and is delayed by the physical distance between neurons (the speed of transmission is assumed the same for all connections). However, the interactions of neurons that result in macro-level emergent behaviors are unknown and obviously neurons alone are not able to deal with this information. To this end, we defined mechanisms of artificial self-organization for individual neurons based on biological knowledge. Moreover, to be able to specify purely local information about the reference macro-level pattern, we used the peristimulus frequency (PSF) analysis method (Türker and Powers 2005). This way, using a self-organizing model and reference data encoded as PSF, the model is able to build an artificial neuronal network from an initial setting that is functionally equivalent to its reference biological network.

The accuracy of this new model has been proved using recorded discharge rates of motoneurons in human subjects. Driven by intermittent activations of sensory neuron(s) and the spontaneous activity of the motoneuron(s), an artificial neuronal pathway emerges through recruitment, dismission and modification of neurons and synapses until it reaches a state where further organizational changes do not occur. The outcome is a final neuronal pathway, the emergent behavior and underlying neuronal dynamics of which can now be studied in *ideal conditions*. The results obtained show that the model simulates remarkably similar networks to their reference human neuronal pathways from the point of view of functionality.

Besides, it is worth pointing out the main contributions of this article as follows:

– We have, for the first time, used artificial self-organization for mimicking human neuronal pathways. This led us to generate artificial replications of real human neuronal pathways without defining the transition from microscopic to macroscopic scales using mathematical arguments. The *micro-level* entities of the system, individual neurons, find their right organization *without knowing* the macro-level function.
– We have, for the first time, used the PSF technique for mimicking human neuronal pathways in a computational simulation study. The PSF records obtained from humans are used as a reference macro-level function for the entire system.
– The principal advantage of our model is that it is flexible and not bound to particular pathways, therefore can learn new pathways.

We view the impact of this work as twofold. First, it provides a practical approach for mimicking human neuronal pathways, which are often expressed as *black box*, by

automating the neuronal network design step that is practically impossible. Second, it opportunistically combines the strengths of neuroscience and artificial self-organizing systems into a single approach. Though the neuroscience and artificial self-organization communities are mostly disjoint and focus on somewhat different problems, we find that each can benefit from the progress of the other. On the one hand, we show that methods for artificial self-organization can help mimicking human neuronal pathways successfully. On the other hand, we show that established techniques from the neuroscience community can make artificial self-organization applicable to real neuroscience problems.

In what follows, we first provide the background of our artificially self-organizing neural network model. Then, based on this background we present our model. The key idea in our model is that the artificial self-organization scheme places plausible biological constraints (of the sort that apply to neurons) on the models that emerge from this process. The example we use to illustrate the approach is the construction of neuronal circuits that reproduce the distribution of neuronal firing in response to an external (sensory) stimulation. The crucial aspect of this process is that the number of elements (neurons) and connections (synapses) is optimized in a self-supervised way, thereby exploring a potentially large space of models automatically, using simple and local rules. Finally, we discuss our model from several perspectives and conclude the article.

2 Foundations of the model

This section presents the fundamentals for understanding the proposed model, which are the concepts related to *artificial self-organization* and *the peristimulus frequency analysis*, and notations used in the rest of the paper.

2.1 Artificial self-organization

Artificial self-organizing systems are those built from the beginning with embedded self-organization capabilities for enabling the engineering of underspecified software. There are various artificial self-organization approaches whose mechanisms are borrowed from existing natural systems, and/or created explicitly for that purpose (Serugendo et al. 2011).

The approach proposed here to make the neuronal pathways find the right wiring to reflect experimental data and then be able to explain how neurons link together, is based on multi-agent systems (MAS). A MAS is an artificial (or physical) system made up of lower-level entities, called agents, which are able to behave autonomously and interact in order to realize a collective behavior at the system

higher-level (Wooldridge 2002). More precisely, MAS considered here are adaptive ones and developed based on the Adaptive Multi-Agent Systems (AMAS) theory (Capera et al. 2003). The AMAS theory has successfully been applied to various application domains such as ambient systems (Guivarch et al. 2012), manufacturing control (Kaddoum and Georgé 2012), maritime surveillance (Brax et al. 2012), optimization (Combettes et al. 2012), bioprocesses control (Videau et al. 2011), crisis management (Lacouture et al. 2011), user profiling (Lemouzy et al. 2010), simulation of the functional behavior of a yeast cell (Bernon et al. 2009), aircraft/avionics design (Welcomme et al. 2009), dynamic ontologies (Ottens et al. 2007) and aeronautical mechanisms design (Capera et al. 2004).

Adaptation is commonly defined by the capability a system has to modify its behavior to accurately predict its environment, especially when the environment is changing. Considering that each part of a system carries out a partial function, the global function of this system results from a combination of the partial functions which mainly depends on the current organization of the parts. Therefore, if the parts are autonomously able to change their interactions, and therefore their organization, the global function is also modified.

In AMAS, cooperation is the criterion that makes an agent decide to modify its behavior and/or interactions with others in order to indirectly adapt the function performed at the collective level (Georgé et al. 2004). This collective function is not hard-coded in the system but emerges from the interactions at the agent-level.

2.1.1 AMAS model description

In the AMAS approach, the system is composed of a set of dynamic number of autonomous cooperative agents $\mathcal{A} = \{a_0, a_1, ...\}$. In this approach, a system is said to be *functionally* adequate if it produces the function for which it was designed, according to the viewpoint of an external observer who knows its finality.[5] To reach a functional adequacy, it has been proven that each autonomous agent $a_i \in \mathcal{A}$ must keep relations as cooperative as possible with its social (other agents) and physical environment (Capera et al. 2003; Camps et al. 1998). Cooperation can be defined as the ability agents possess to work together in order to realize a common global goal.[6] Cooperative agents try, on the one hand, to anticipate cooperation problems and, on the other hand, to detect cooperation failures and try to repair them (Picard and Gleizes 2005).

A non-desired configuration of inputs causes a non-cooperative situation (NCS) to occur. An agent a_i is able to memorize, forget and spontaneously send feedbacks related to desired or non-desired configurations of inputs coming from other agents. We denote the set of feedbacks as $\mathcal{F}$ and we model sending a feedback $f_a \in \mathcal{F}$ using an action of the form send($f_a, \mathcal{R}$) where a is the source of f and $\mathcal{R} \subset \mathcal{A} \setminus \{a\}$ is the set of receiver agents. A feedback $f_a \in \mathcal{F}$ can be about increasing the value of the input ($f_a\uparrow$), decreasing the value of the input ($f_a\downarrow$) or informing that the input is good ($f_a\approx$).

When a feedback about a NCS is received by an agent, at any time during its lifecycle, it acts in order to overcome this situation or subsequently avoid it (Bernon et al. 2009) for coming back to a cooperative state. This provides an agent with learning capabilities which is therefore able to constantly adapt to new situations that are judged harmful. Consequently, the behaviors that agents are using for adaptation purposes are called *adaptive behaviors*. In case an agent cannot overcome a NCS using its *adaptive behaviors*, it keeps track of this situation by using a level of annoyance value ψ_{f_a} where f_a is the feedback about this NCS and propagates f_a (or an interpretation of f_a) to all or some of its neighbor agents that may handle it. When a NCS is overcome using an *adaptive behavior*, ψ_{f_a} is set to 0, otherwise it is increased by 1 until it reaches an annoyance threshold, which may cause execution of another *adaptive behavior* that makes an immediate organizational change.[7] The first adaptive behavior an agent tries to adopt to overcome a NCS is a *tuning behavior* in which it tries to adjust its internal parameters without changing the organization. If this tuning is impossible because a limit is reached or the agent knows that a worse situation will occur if it adjusts in a given way, it may propagate the feedback (or an interpretation of it) to other agents that may handle it. If such a behavior of tuning fails many times and ψ_{f_a} crosses the reorganization annoyance threshold $\psi_{reorganization}$ (reorganization condition), an agent adopts a *reorganization behavior* for trying to change the way in which it interacts with others (e.g., by changing a link with another agent, by creating a new one, by changing the way in which it communicates with another one and so on).[8] In the same way, for many reasons, this behavior

---

[5]Considering their global scope, emergent (functional) phenomena can generally be identified by some observer located outside the system that produces them (de Haan 2006; Goldstein 1999).

[6]For a detailed discussion on cooperation refer to (Georgé et al. 2011).

[7]This idea is based on physical self-organizing systems where there exists some critical threshold which causes an immediate change to the system state when reached (Nicolis and Prigogine 1977). Thus, adaptive behaviors are typically governed by a *power law* (Heylighen 1999) which states that large adjustments are possible but they are much less probable than small adjustments.

[8]The term *reorganization* in terms of self-organization was first used by Koestler in late 1960s (Koestler 1967). In his study, he defines *holons* and *holarchies* where order can result from disorder with progressive reorganization of relations between complex structural elements (see (Serugendo et al. 2011) referring to (Koestler 1967)).

may fail counteracting the NCS and a last kind of behavior may be adopted by the agent: *evolution behavior*. This is detected when ψ_{fa} crosses the evolution annoyance threshold $\psi_{evolution}$ (evolution condition).[9] In the evolution step, an agent may create a new agent (e.g., for helping itself because it found nobody else) or may accept to disappear (e.g., it was totally useless and decides to leave the system). In these two last levels, propagation of a problem to other agents is always possible if a local processing is not achieved. The overall algorithm for suppressing a NCS by an agent is given in Algorithm 1.

Algorithm 1 Cooperative behaviors of an agent a_i upon receiving a feedback f_{a_j} about a NCS of the agent a_j.

1:   **if** $\psi_{f_{a_j}} < \psi_{reorganization}$ // tuning condition
2:      **if** tuning succeeds **then** $\psi_{f_{a_j}} \leftarrow 0$
3:      **else** $\psi_{f_{a_j}} \leftarrow \psi_{f_{a_j}} + 1$ **endif**
4:   **else if** $\psi_{f_{a_j}} < \psi_{evolution}$ // reorganization condition
5:      **if** reorganization succeeds **then** $\psi_{f_{a_j}} \leftarrow 0$
6:      **else** $\psi_{f_{a_j}} \leftarrow \psi_{f_{a_j}} + 1$ **endif**
7:   **else** // evolution condition
8:      **if** evolution succeeds **then** $\psi_{f_{a_j}} \leftarrow 0$
9:      **else** $\psi_{f_{a_j}} \leftarrow \psi_{f_{a_j}} + 1$ **endif**
10: **endif**
11: **if** $\psi_{f_{a_j}} > 0$ // propagation condition
12:    send(f_j, R)
13: **endif**

The AMAS approach is a proscriptive one because each agent must first of all anticipate, avoid and repair NCSs. By contrast, the designer, according to the problem to be solved, has: 1) to determine what an agent is, then 2) to define the nominal behavior which represents an agent's behavior when no NCS exist, then 3) to deduce the NCSs the agent can be faced with, and finally 4) to define the cooperative behavior (tuning, reorganization and/or evolution) the agent has to perform when faced with each NCS in order to come back to a cooperative state. This is the process adopted in the design of the proposed computational model.

Moreover, to build a real self-adaptive system, the designer has to keep in mind that agents only have a local view of their environment and that they do not have to base their reasoning on the collective function that the system must achieve.

[9]Naturally $\psi_{evolution} > \psi_{reorganization} > 0$.

2.2 Effective connectivity analysis of human neuronal pathways

In this subsection, we provide a brief overview of the techniques that can be used for extracting a comprehensible information about effective connectivity of human neuronal pathways.

The ability to record motor unit activity in human subjects has provided a wealth of information about the neural control of motoneurons, and in particular has allowed the study of how reflex and descending control of motoneurons change as a function of task, during fatigue and following nervous system injury. Although synaptic potentials cannot be directly recorded in human motoneurons, their characteristics can be inferred from measurements of the effects of activating a set of peripheral or descending fibers on the discharge probability of one or more motoneurons.

There are several techniques for quantifying stimulus-evoked changes in responses of neurons. The three most common are full-wave rectification and averaging of the electromyography (EMG) record around the time of stimulation (Jenner and Stephens 1982), compiling peristimulus time histograms (PSTHs) from single motor unit (SMU) records (Stephens et al. 1976) and compiling a peristimulus frequency-gram (PSF) that uses the instantaneous discharge rates of single motor units for estimating the synaptic potentials produced by afferent stimulation (Türker and Powers 1999, 2003, 2005). The PSF plots instantaneous discharge rates against the time of the stimulus and is used to examine reflex effects on motoneurons, as well as the sign of the net common input that underlies the synchronous discharge of human motor units. The instantaneous discharge rates comprising the PSF should not necessarily be affected by previous (prestimulus) activity at any particular time. However, since the discharge rate of a motoneuron reflects the net synaptic current reaching its soma (Powers et al. 1992; Gydikov et al. 1977), any significant change in the poststimulus discharge rate should indicate the sign and the profile of the net input. Consequently, PSF has been established to be a superior method for studying synaptic potential in human subjects where all estimations have to be performed indirectly (see Fig. 1). Other methods such as the rectified surface electromyography (EMG) and peristimulus time histogram (PSTH) rely upon probability of spike occurrence and hence introduce several count and synchronization type errors (reviewed in (Türker and Powers 2005)). Consequently, since the discharge-rate-based method was superior to the other methods, it was used in the proposed model to indicate synaptic connections between neurons and for generating the artificial neuronal network.

Nevertheless, the accuracy of the reference discharge-rate data and their efficient interpretation are important as
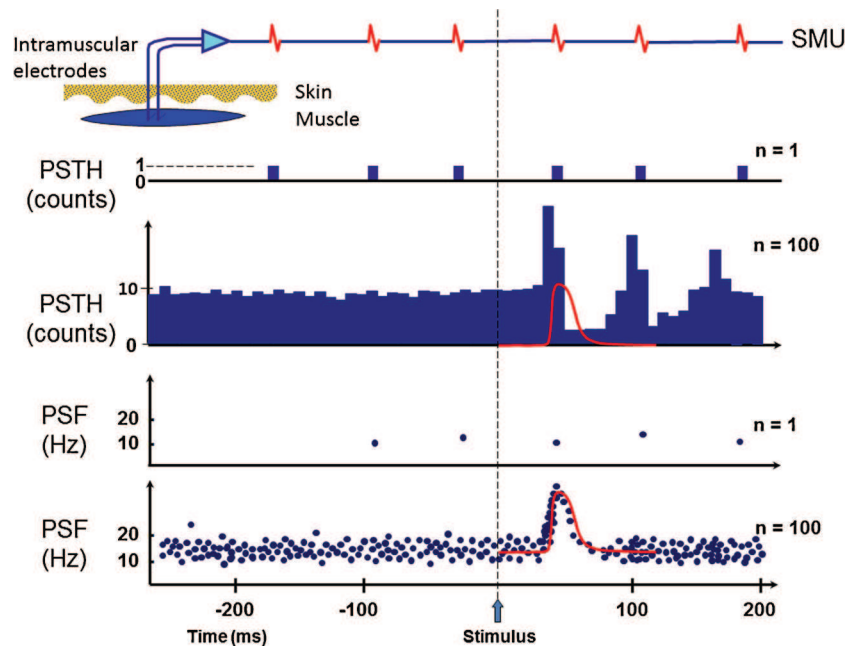
Fig. 1 PSF and PSTH methods: To examine how a sensory nerve (afferent) that transmit information from a sensory receptor (such as touch, temperature, etc) is connected to a motor neuron, we stimulate the nerve and record the response of the motor neuron to this stimulus by inserting a needle into a muscle that carries information from the motor neuron within the central nervous system. Therefore, we deliver an input into the system and record its response. Using the characteristics of this response, we work out the pathway that connects the stimulated afferent nerve to the motor neuron that connect to the muscle single unit that we record from. This figure illustrates responses of a single motor unit (SMU) to a simulus delivered at time zero (arrow; red traces in PSTH and PSF records represent stimulus induced synaptic potentials that develop in the motor neuron membrane). Single motor unit (SMU) action potentials (spikes) are recorded using intramuscular wire electrodes (bipolar configuration) around the time of the stimulus (top trace). Peristimulus time histogram (PSTH) converts each SMU spike (top trace) into acceptance pulses and indicates them as counts exactly at each bin they occur (second trace from the top). When a large number of stimuli are delivered and SMU acceptance pulses are piled up, we obtain PSTH (middle trace). Interspike intervals (ISI) of SMU spikes are measured in seconds and converted into instantaneous discharge rates (1/ISI=Hz). To measure ISI we need two

spikes and the discharge rate value is indicated exactly in the same bin of the second spike. Since the very first SMU spike in the PSF trace (second trace from bottom) does not have a preceding spike, it has no discharge rate value. The other ISIs are converted into Hz and indicated as shown. When a large number of instantaneous discharge rates are superimposed (not piled up as the PSTH records), we obtain the PSF (bottom trace). The red traces in PSTH and PSF represent actual injected current measured from a living motor neuron during the process in the development of the PSF technique (for details see (Türker and Powers 1999)). In the brain slice experiments similar injected currents induced similar PSTHs and PSFs that are represented in this diagram. As can be seen, the PSTH does not indicate the actual injected current into a motor neuron. Even worse, it generates secondary peaks and troughs that are not due to the injected current but due to the count and synchronization errors that are applied to this method. PSF on the other hand, very closely represents the profile of the actual injected current (synaptic potential) developed on the motor neuron. For this very reason, we used PSF to represent the synaptic potential in our system which indicates not only the number of synapses in the system (using the latency of the response) but also the sign and strength of the connection (using the profile of the response) in the neuronal pathway

they affect the correctness of the feedbacks in the model. In this sense, to determine significant deflections, the cumulative sum (CUSUM) of PSF record is used (see Fig. 2a and b). The CUSUM is calculated by subtracting the mean prestimulus baseline from the values in each bin and integrating the remainder (Ellaway 1978). Stimulus-induced effects are considered significant if the poststimulus CUSUM values exceed the maximum prestimulus CUSUM deviation from zero (i.e. the error box (Türker et al. 1997; Brinkworth and Türker 2003), indicated by the horizontal lines in Fig. 2b). As can be seen from this figure, there is an early and long-lasting excitation (LLE) indicated by the increased discharge rate from about 40 ms poststimulus to about 100 ms. After this LLE there is a period of long-lasting

inhibition (LLI) going from about 100 ms poststimulus to about 300 ms. However, since after 200 ms of stimulation the subject is able to change the discharge rate of his/her motor unit voluntarily (minimum reaction time), the events later than 200 ms cannot be considered as reflex events. Only before 200 ms of poststimulus discharge rates might give exact information about the network of the motoneuron. However, CUSUM does not give comprehensible local information from the point of view of a motor neuron. For tackling this problem, a solution could be calculating the derivative values of PSF-CUSUM values (Fig. 2c). The derivative values at each time *t* give (local) information about the instant pre-synaptic effects on a motoneuron.
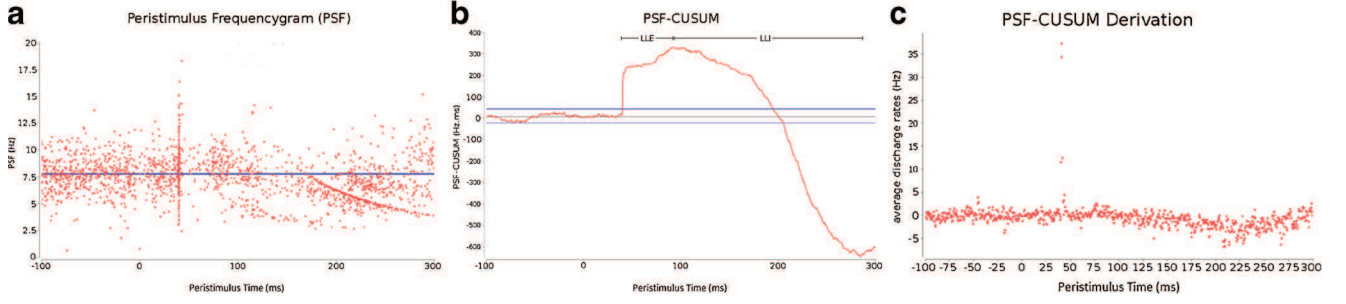
Fig. 2 Motoneuron responses to single post-synaptic potential (PSP) compiled by recording the motoneuron discharges from human soleus muscle using a surface electrode. **a** shows the Peristimulus frequency-gram (PSF) that plots the instant discharge rate values of these responses (*red dots*) to their occurence latency against the stimuli applied (the horizontal *blue line* represents the mean-prestimulus base-line), **b** shows the cumulative sum (CUSUM) of the PSF given in (**a**), and (**c**) shows the derivative values obtained from the CUSUM in (**b**)

3 The computational model

This section describes the computational neuronal network model which is based on the AMAS theory. The right number of neurons composing the network as well as the links between them will emerge enabling this network to self-build.

The basic design of this model was previously endorsed by the artificial self-organizing systems community (Gürcan et al. 2012). Here we present its improved version by taking into account more details and validating its results on various different data sets.

3.1 Identification of agents and their nominal behaviors

We model the computational model Sim basically capturing all taken design decisions based on the AMAS theory as $Sim = (\mathcal{G}, \nu)$ where $\mathcal{G}$ is the agent-based neuronal network and ν is the viewer agent. The agent-based neuronal network $\mathcal{G}$ is presented in Section 3.1.1. For clarity, the viewer agent ν is presented separately in Section 3.1.2.

3.1.1 Agent-based neuronal network model

In this subsection, we present a model of biological neuronal networks. Since network theory is a subset of graph theory, we model the neuronal network as a dynamic directed graph $\mathcal{G}(t) = (\mathcal{N}(t), \mathcal{S}(t))$ where $\mathcal{N}(t) \subset \mathcal{A}$ denotes the time varying neuron agent (vertex) set and $\mathcal{S}(t)$ denotes the time varying synapse (edge) set. The set of excitatory (respectively inhibitory) neuron agents at time t is denoted by $\mathcal{N}^+(t)$ (resp. $\mathcal{N}^-(t)$) where $\mathcal{N}(t) = \mathcal{N}^+(t) \cup \mathcal{N}^-(t)$. The nominal behavior of neuron agents is *spike firing*. A neuron agent n fires a spike when its membrane potential p_n crosses the firing threshold. We define $\mathcal{N}^{spike}(t)$ to be the set of neuron agents that fired their last spike at time t. We also denote t_n for indicating the last spike firing time of the neuron agent n where $n \in \mathcal{N}^{spike}(t_n)$.

When a neuron agent n fires a spike, this spike is emitted through its synapses to the postsynaptic neuron agents. We denote the set of presynaptic neighbors of a neuron agent n at time t as $Pre_n(t) = \{m \in \mathcal{N}(t)|\{m, n\} \in \mathcal{S}(t)\}$ and the set of postsynaptic neighbors of a neuron agent $n \in \mathcal{N}$ at time t as $Post_n(t) = \{k \in \mathcal{N}(t)|\{n, k\} \in \mathcal{S}(t)\}$. Apart from pre- and postsynaptic neighbors, a neuron agent has also another type of neighborhood, called friends, which contains all the neuron agents it has contacted during its lifetime, even the synapses in between removed after some time. Formally, a friend agent m of a neuron agent n at time t is denoted by there exists some time $t' \leq t$ such that $m \in Pre_n(t')$ or $m \in Post_n(t')$. All friends of n at time t are denoted by $Friend_n(t)$. The set of presynaptic neuron agents that contributes to the activation of a postsynaptic neuron n at time t_n is modeled as $Cont_n(t_n)$ where $Cont_n(t_n) \subseteq Pre_n(t_n)$ and $t_n > t_k > t_n - d_{psp}$[10] for all $k \in Cont_n(t_n)$. Lastly, a neuron agent has to know the friends who activated temporally closest to its activation. Formally, a temporally closest friend agent m of a neuron agent n at time t is denoted by there exists some time $t' < t_n \leq t$ such that $m \in Friend_n(t)$, $t' = t_m$ and there is no $t' < t'' < t_n$ such that for all $k \in Friend_n(t)$, $t'' = t_k$. All temporally closest friends of n at time t are denoted by $Temp_n(t)$.

A synapse $\{n, m\}$ conducts a spike from n to m through the interval $[t_n, t']$ if $n \in \mathcal{N}^{spike}(t_n)$, and $t' = t_n + d_{nm}$ where d_{nm} is the delay for delivering the spike from n to m. We denote the spike delay as $d_{nm} = d_{nm}^{ax} + d_{nm}^{den}$ where d_{nm}^{ax} is the *axonal delay* of $\{n, m\}$ and d_{nm}^{den} is the *synaptic processing time*. We assume that for all $\{n, m\} \in \mathcal{S}(t)$, $d_{nm}^{den} = 0.5$ ms (Kandel et al. 2000). The axonal delay d_{nm}^{ax}, on the other hand, may change depending on the length and type of the axon. When a spike transmitted by n

---

[10]It is assumed that the *post-synaptic potential (PSP) duration* $d_{psp} = 4.0$ ms.

reaches $\{n, m\}$, a postsynaptic potential (PSP) occurs on m for 4.0 ms (*PSP duration*). The PSP for a unitary synapse can range from 0.07 mV to 0.60 mV (see Figure 4 in Iansek and Redman 1973). Thus, we say that a synapse $\{n, m\}$ potentiates (respectively depresses) the membrane potential p of m with a synaptic strength η at time t' where $0.07 \leq |\eta| \leq 0.60$ during the *PSP duration* $d_{psp} = 4.0$ ms if $n \in \mathcal{N}^+$ (resp. $n \in \mathcal{N}^-$) and m is not removed at any time during the interval$[t', t' + d_{psp}]$.

We model the set of sensory neuron agents at time t as $\mathcal{K}(t) \subset \mathcal{N}^+(t)$ where for all $s \in \mathcal{K}(t)$, we have $Pre_s(t) = \varnothing$ and $Post_s(t) \neq \varnothing$. Since $Pre_s(t) = \varnothing$, they have a nominal action of the form activate() triggered by the viewer agent (see next subsection) in order to be able to fire.

We model the set of motoneuron agents at time t as $\mathcal{M}(t) \subset \mathcal{N}^+(t)$ where for all $m \in \mathcal{M}(t)$, we have $Pre_m(t) \neq \varnothing$ and $Post_m(t) = \varnothing$. Motoneurons are tonically active and are affected by neurons connected to them. Hundreds of EPSPs and IPSPs from sensory neurons and interneurons arrive at different times onto a motoneuron. This busy traffic of inputs create the 'synaptic noise' on the membrane of the motoneuron. As the consequence of this noise, spikes occur at nearly random times (Fig. 3). For mimicking such a noisy nominal behavior, *the motoneuron agent m uses the prestimulus part of the reference data.*

When the reference data are provided to m, it calculates a statistical distribution using the prestimulus discharge rate values. Then using the statistical parameters of this distribution, a discharge rate generator, which is used to generate consecutive interspike interval (ISI) values for m, is created. Each time a new ISI value is calculated using this generator, the instant membrane potential increase Δp is calculated as $\Delta p = AHP_m/ISI$ where AHP_m is the after-hyperpolarization (AHP) level of m and at each $tick$, p_m is increased by $\Delta p * tick$. As a result, the AHP time course of m is a straight line as shown in Fig. 3.

Finally, we model the set of interneuron agents as $I \subset \mathcal{N}$ where for all $n \in I$, we have $Pre_n \neq \varnothing$, $Post_n \neq \varnothing$ and $d^{ax} = 0$ since their axonal delays are extremely low.

3.1.2 The viewer agent

The viewer agent v is designed to trigger the modification of synaptic connections and the effective connectivity of the agent-based neuronal network $\mathcal{G}$. It knows the experimental reference data but it does not know why and how $\mathcal{G}$ simulates the behavior of the real system. The viewer agent v acts like a surface electrode and gives inputs to $\mathcal{G}$ by coordinating random activation of all sensory neuron agents $s \in \mathcal{K}$ (by triggering their activate() action). Meanwhile, it
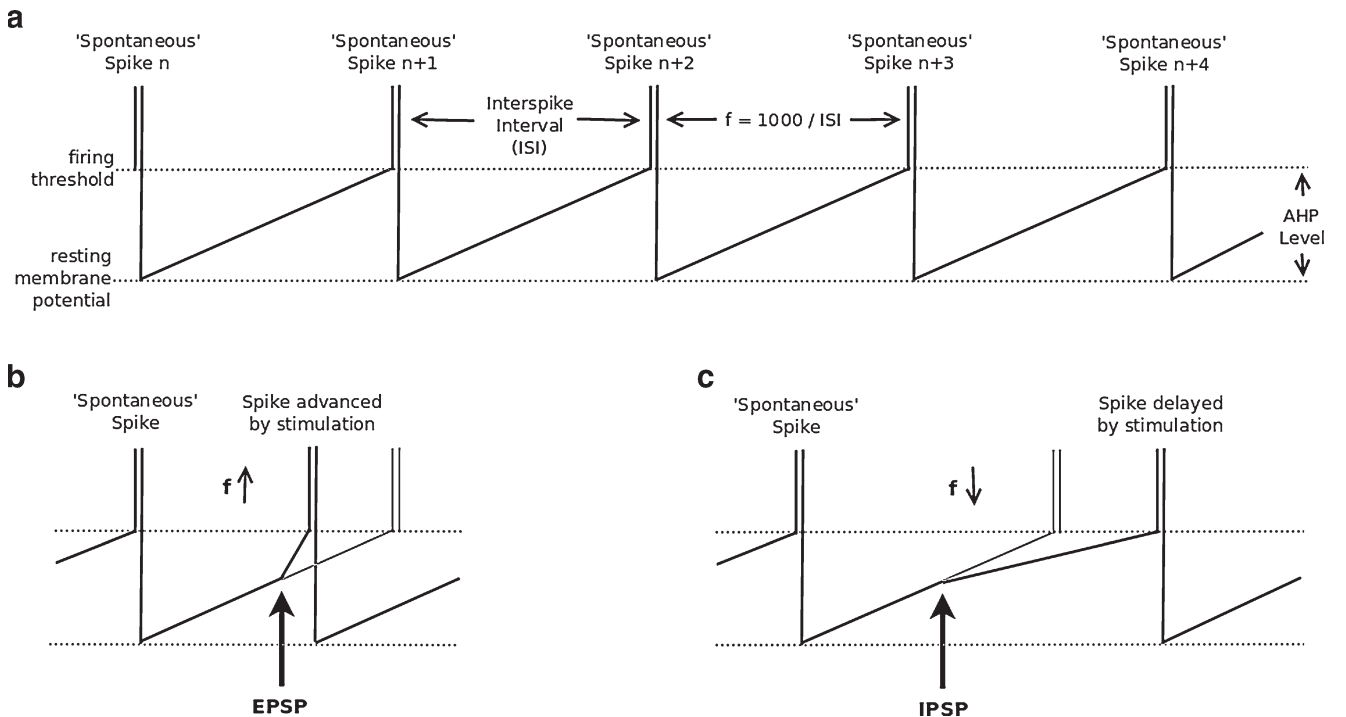
Fig. 3 Tonic firing of a neuron (modified from (Türker and Miles 1991)). During tonic firing, a neuron receives continuous current and hence its membrane potential continuously rises to the firing threshold and makes the neuron fire spontaneous spikes (**a**). The time intervals between consecutive spikes are called interspike intervals (ISI) and the instantaneous rate of a spike is calculated as $f = 1000/ISI$. While an EPSP induces a phase forward movement of the next spike (and thus increases the instant rate) (**b**), IPSP delays the occurrence of the next spike (and thus decreases the instant rate) (**c**)

monitors and records the outputs of the motoneuron agents $m \in \mathcal{M}$ that take place over time to the simulated data $\mathcal{E}_S$ for comparing them with reference data $\mathcal{E}_R$. This comparison takes place between the latency of the beginning (l_{begin}) and the end (l_{end}) of the network where $l_{end} > l_{begin} > 0$. According to this comparison, ν makes assessments about the behavior of m for detecting if it is functionally adequate to the reference real motoneuron or not. If the output observed at the latency l is generated between l_{begin} and l_{end} where $l_{end} \geq l \geq l_{begin}$, ν sends an appropriate instant discharge rate feedback $f \in \mathcal{F}_{dr}$ to m. Otherwise, ν considers this output as a nominal behavior of m and does not send any feedback.

In detail, the above comparison is conducted as follows. When an output is observed by ν at time t, first the latency of this output (l) is calculated (Algorithm 2, line 1). Then, to be able to determine the similarity between the reference output at latency l and the simulated output at latency l, firstly both $\mathcal{E}_R$ and $\mathcal{E}_S$ are converted into their respective PSF-CUSUM values $\mathcal{C}_R$ and $\mathcal{C}_S$ (Algorithm 2, line 3). The discharge rate of motor units integrates the excitatory and inhibitory synaptic activities (Gydikov et al. 1977) and these activities can be identified by calculating the derivative values of PSF-CUSUM of the motoneuron as described in Section 2.2. Thus, the derivative values for each PSF-CUSUM diagram ($\mathcal{C}'_R$ and $\mathcal{C}'_S$) are then calculated (Algorithm 2, line 4). However, these values involve also the noisy behavior of the motoneuron. In this sense, the moving averages of $\mathcal{C}'_R$ and $\mathcal{C}'_S$ are calculated (Algorithm 2, line 5) in order to reduce the noise of the motoneuron. However, comparison of these moving average values is not reliable since both moving average diagrams may not be statistically identical: they may have different prestimulus means and different standard deviations (σ). In this sense, the similarity of the average values at the latency l is calculated in terms of their prestimulus standard deviations (Algorithm 2, line 6) and is set to δ. This δ value is then compared to a tolerance of τ, and finally an appropriate feedback is sent to m (Algorithm 2, lines 7, 8 and 9).

In order to provide a good feedback mechanism, the moving average PSF-CUSUM derivative diagram is developed in 4 stages: (1) the bins in the PSF diagram with no discharge rate value are given the values in the preceding bin. This is to ensure that the discharge rate CUSUM value does not suddenly drop down in these bins (Türker and Cheng 1994). This approach assumes that empty bins represent the same rate as the preceding bin even though they fail to be filled due to the low number of trials and/or due to the chance. However, it is also possible that a large PSP may cause a large number of *consecutive empty bins*. To overcome this phenomenon, a histogram of the consecutive empty bins is built. For a given number of trials, the distribution of the empty bins illustrates the occurence of *consecutive empty bins*. It was found by Türker and Cheng (1994) that some of the large numbers could be interpreted as forming the tail of a normal distribution. The occurences that are larger than "the mean +3 standard deviations" are taken to indicate that these empty bins did not occur by chance and/or low number of trials. When this occured, the position of these empty bins is not filled in with the values in the preceeding bin. (2) Then the CUSUM of this filtered record of the PSF is calculated using the rate values in each bin (from -400 ms to 200 ms – 600 ms bandwidth).[11] (3) The derivative values in each bin is then calculated by using the PSF-CUSUM record. (4) Finally, the PSF-CUSUM derivative record is filtered by the moving averager using 8 bin average and smoothed 10 times.

Algorithm 2 Response of the viewer agent ν upon monitoring an output at time t from motoneuron $m \in \mathcal{M}(t)$ where t_{sti} is the last stimulation time of sensory neurons, $\mathcal{E}_S$ is the set of outputs observed by m and σ represents a standard deviation.

1:    $l \leftarrow (t - t_{sti})$
2:    **if** ($l_{begin} \leq l \leq l_{end}$) **then**
3:       $\mathcal{C}_S$ (res. $\mathcal{C}_R$) = calculate psf-cusum of $\mathcal{E}_S$ (res. $\mathcal{E}_R$)
4:       $\mathcal{C}'_S$ (res. $\mathcal{C}'_R$) = calculate derivative of $\mathcal{C}_S$ (res. $\mathcal{C}_R$)
5:       $\mathcal{C}''_S$ (res. $\mathcal{C}''_R$) = calculate moving avg. of $\mathcal{C}'_S$ (res. $\mathcal{C}'_R$)
6:       $\delta = \mathcal{C}''_S(l)/\sigma''_S - \mathcal{C}''_R(l)/\sigma''_R$
7:       **if** ($\delta > \tau$) **then** send($f_\nu \downarrow, m$)
8:       **else if** ($\delta < -\tau$) **then** send($f_\nu \uparrow, m$)
9:       **else** send($f_\nu \approx, m$)
10:     **endif**
11: **endif**

Additionally, ν is responsible for stopping the simulation run when the evolution of the neuronal network ends. ν detects this situation by evaluating the output of m. Formally, $\mathcal{G}$ is said to be stable at time t' if for all $t_1, t_2 \in \mathbb{R}^+$ where $t_2 > t_1 \geq t'$, for all feedbacks f, we have $f \in \mathcal{F}_{dr} \approx$.

3.2 Identification of non-cooperative situations and feedbacks

The proposed agent-based neuronal network model, in which neuron agents and synapses can be inserted or

---

[11]This bandwidth is chosen to make sure that the poststimulus 'event' is larger than the maximum possible prestimulus variations in both directions (above and below the line of equity). We double the length of the prestimulus time to account for the by chance 'excitation' and 'inhibition' as the CUSUM can go both directions.

removed, is subjected to NCSs. All NCSs are identified by analyzing the possible bad situations of real human motor units.

3.2.1 Bad temporal integration

The temporal integration of the inputs provided by synapses affects what a neuron agent does. For an interneuron agent n, these inputs affect whether n can fire or not, while for a motoneuron agent m they affect the instant discharge rate of the spikes of m since m is tonically active. Sensory neurons, however, never detect this situation since they do not need input neurons in order to fire. Consequently, when this temporal integration is bad, a neuron agent either cannot fire or have a bad firing behavior. When such a situation is detected at time t, the neuron agent should improve its existing inputs or should search for new inputs with the right timing. To do so, it sends a *temporal integration increase* or *decrease* feedback ($f \in \mathcal{F}_{ti}\uparrow$ or $f \in \mathcal{F}_{ti}\downarrow$) to some or all of its neighbor neuron agents. Otherwise, the temporal integration is good and a *temporal integration good* feedback ($f \in \mathcal{F}_{ti}\approx$) is sent to $Pre_n(t)$.

An interneuron agent detects a *bad temporal integration NCS* at time t if during the interval $[t - \Delta t_{max}, t]$ it did not generate any spike where Δt_{max} is the maximum time slice an interneuron agent can stay without spiking. However, the motoneuron agent m is unable to detect the same situation by itself. It detects when it receives an instant discharge rate feedback $f_v \in \mathcal{F}_{dr}$ at time t about its last spike at time t_m (see Section 3.2.2). Since this spike is related to the temporal integration of its own membrane potential and its presynapses, there are two cases: (1) $Cont_m(t_m) = \varnothing$, and (2) $Cont_m(t_m) \neq \varnothing$. In the first case, m sends a feedback $f_m \in \mathcal{F}_{dr}$ to its temporally closest friend neurons $Temp_m(t_m)$ to be able to have contributor synapses. In the second case, the problem can be turned into a temporal integration problem (see Fig. 3b and c) and a *temporal integration feedback* $f_m \in \mathcal{F}_{ti}$ is sent to its temporally closest contributed neurons $Temp_m(t_m)$.

3.2.2 Bad instant discharge rate

A motoneuron agent fires continuously and its firing behavior might be affected by its pre-synapses when a stimulation is given to the sensory neuron agents. The motoneuron agent is expected to generate discharge rates similar to the reference data. When the motoneuron agent emits a spike, the viewer agent observes it and calculates the instant discharge rate value for that spike using the previously emitted spike. However, it is not logical to compare an individual discharge rate information to the reference data since there can be many discharge rate values at a specific time and the reference data contain the noisy behavior of the motoneuron.

In this sense, to reduce the noise and to facilitate the comparison, the moving average discharge rate values are used. Consequently, the average discharge rate at time of spike is expected to be close enough to the average discharge rate of the reference data. As a result of this comparison, the viewer sends an *instant discharge rate is good, increase* or *decrease* feedback ($f_v \in \mathcal{F}_{dr}\uparrow$, $f_v \in \mathcal{F}_{dr}\downarrow$ or $f_v \in \mathcal{F}_{dr}\approx$) to the motoneuron agent.

3.3 Cooperative behaviors

The *tuning behavior* of neuron agents is modelled using an action of the form tune($\{n, m\}, f$) for $n, m \in \mathcal{N}(t)$ and $f \in \mathcal{F}$, which correspond to the adjustment of $\{n, m\}.\eta$ by f at time t. An autonomous and cooperative neuron agent must be able to decide by itself the modification of its synapse. By contrast, this action can only be executed by n over $\{n, m\}$. Moreover it is ensured that no opposite adjustment is done at the same time by n. The *reorganization behaviors* of neuron agents are modeled using actions of the form add($\{n, m\}$) and remove($\{n, m\}$) for $n, m \in \mathcal{N}(t)$, which correspond to the formation and suppression (respectively) of $\{n, m\}$ at time t. It is assumed that no synapse is both added and removed at the same time. The *evolution behaviors* of neuron agents are modelled using actions of the form create(n, m), createInverse(n, m) and remove(n) for $n, m \in \mathcal{N}(t)$. The create(n, m) action corresponds to the creation of a neuron agent between n and m by n having the same type of n. The createInverse(n, m) action corresponds to the creation of a neuron agent between n and m by n having the opposite type of n. The remove(n) action corresponds to the suppression of the neuron agent n by itself. It is assumed that no neuron agent is both added and removed at the same time. Figure 4 presents examples of these cooperative actions.

NCSs are suppressed by processing the aforementioned actions as described in the following subsections.

3.3.1 Suppression of "bad instant discharge rate" NCS

When the motoneuron agent m receives $f_v \in \mathcal{F}_{dr}$ from v about its last spike (t_m) at time t, it evaluates f_v taking into account $Temp_m(t_m)$ in a temporal manner. There might be two cases: some $n \in Temp_m(t_m)$ that affect the last spike of m in the right time exists or not. If there exists some $n \in Temp_m(t_m)$ where $t_n = t_m - d_{nm}$,[12] m turns the problem into a temporal integration problem and sends $f_m \in \mathcal{F}_{ti}$ to all $n \in Temp_m(t_m)$. Otherwise, m propagates f_v as $f_m \in \mathcal{F}_{dr}$ to all $n \in Temp_m(t_m)$.

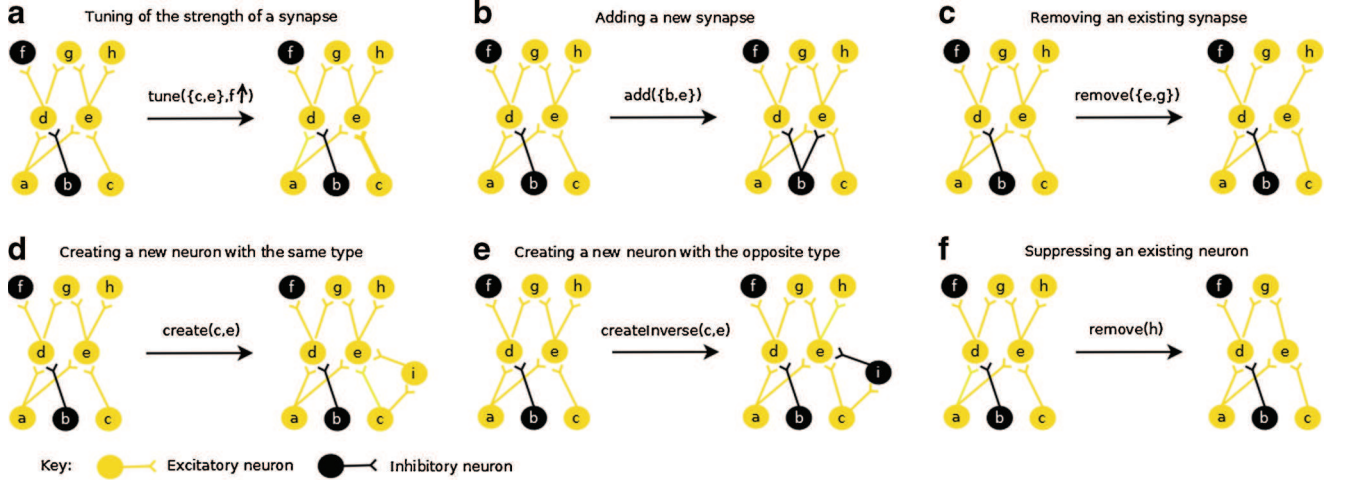[12]In other words, if m spiked as soon as the spike coming from n reached its membrane.

Fig. 4 Examples of cooperative behaviors for modifying the structure of the network. In (**a**), the strength of the synapse $\{c, e\}$, shown with a thicker yellow line, is increased. In (**b**), neuron b creates a new synapse with neuron e. In (**c**), neuron e removes its synapse with neuron g. In (**d**), excitatory neuron c creates a new excitatory neuron whose post-synaptic neuron is e. In (**e**), excitatory neuron c creates a new inhibitory neuron whose post-synaptic neuron is e. In (**f**), neuron h removes itself

When a feedback $f_m \in \mathcal{F}_{dr}$ is received by a neuron agent n, it understands that m needs a temporarily closer neighbor. Since n cannot provide such a neighbor to m by tuning or reorganization, it directly executes its evolution behavior: it creates a new excitatory interneuron agent by including itself as a presynaptic neuron of this new neuron (Algorithm 3, line 2). Therefore, this new neuron will likely fire after n and there will be a time shift in the network. If n is unable to help (e.g., $n \in \mathcal{N}^-$), it propagates the feedback f_m to all its temporally closest friends $Temp_n(t_n)$ (Algorithm 3, line 4).

Algorithm 3 Response to the feedback f_m received at time t of neuron agent n where $f_m \in \mathcal{F}_{dr}$, $m \in \mathcal{M}(t)$ and $n \in \mathcal{N}(t)$.

1: □ evolution condition
2:      **if** $n \in \mathcal{N}^+$ **then** create(n, m) **endif**
3: □ propagation condition
4:      send($f_m, Temp_n(t_n)$)

3.3.2 Suppression of "bad temporal integration" NCS

When a feedback $f_m \in \mathcal{F}_{ti}$ is received by a neuron agent n, it first tries to tune the synaptic strength $\{n, m\}.\eta$ if $m \in Post_m$ (Algorithm 4, line 2). If n cannot help m by tuning, it tries reorganization: either by adding a synapse in between if there is no synapse, or removing the existing synapse. A neuron $n \in \mathcal{N}^+$ (respectively $n \in \mathcal{N}^-$) adds a new synapse (Algorithm 4, line 10) if $f_m \uparrow$ (resp. $f_m \downarrow$) and removes the existing synapse (Algorithm 4, line 6) if $f_m \downarrow$ (resp. $f_m \uparrow$). As a last resort, n creates a new neuron for m. A neuron $n \in \mathcal{N}^+$ (respectively $n \in helping\mathcal{N}^-$) creates a new neuron $n \in \mathcal{N}^+$ (resp. $n \in \mathcal{N}^-$) (Algorithm 4, line 15) if $f_m \uparrow$ (resp. $f_m \downarrow$) and creates a new neuron $n \in \mathcal{N}^-$ (resp. $n \in \mathcal{N}^+$) (Algorithm 4, line 19) if $f_m \downarrow$ (resp. $f_m \uparrow$).

Algorithm 4 Response to the feedback f_m received at time t of neuron n where $f_m \in \mathcal{F}_{ti}$ and $m, n \in \mathcal{N}(t)$.

1: □ tuning condition
2:      **if** $m \in Post_n$ **then** tune($\{n, m\}.\eta$, f_m) **endif**
3: □ reorganization condition
4:      **if** $m \in Post_n$ **then**
5:          **if** $(n \in \mathcal{N}^+ \wedge f_m \downarrow) \vee (n \in \mathcal{N}^- \wedge f_m \uparrow)$ **then**
6:             remove($\{n, m\}$)
7:          **endif**
8:      **else** // $m \notin Post_n$
9:          **if** $(n \in \mathcal{N}^+ \wedge f_m \uparrow) \vee (n \in \mathcal{N}^- \wedge f_m \downarrow)$ **then**
10:            add($\{n, m\}$)
11:          **endif**
12:      **endif**
13: □ evolution condition
14:      **if** $m \in Post_n$ **then**
15:          **if** $(n \in \mathcal{N}^+ \wedge f_m \uparrow) \vee (n \in \mathcal{N}^- \wedge f_m \downarrow)$ **then**
16:            create(Pre_n, m)
17:          **endif**
18:      **else** // $m \notin Post_n$
19:          **if** $(n \in \mathcal{N}^+ \wedge f_m \downarrow) \vee (n \in \mathcal{N}^- \wedge f_m \uparrow)$ **then**
20:            createInverse(Pre_n, m)
21:          **endif**
22:      **endif**
23: □ propagation condition
24:      send($f_m, Temp_n(t_n)$)

When n is unable to help, it propagates the feedback f_m to all its temporally closest friends $Temp_n(t_n)$ (Algorithm 4, line 24).[13] As mentioned before, the bad temporal integration NCS can be detected by both interneuron agents and the motoneuron agent. When detected by an interneuron agent, any synaptic strength change within a d_{psp} (4.0 ms) time range is welcome, since the objective is to activate the interneuron agent. Thus, when a neuron agent cannot suppress such a situation, it asks its presynaptic or postsynaptic neighbors. For the motoneuron agent, however, the objective is to have a synaptic strength change at a specific time. By contrast, when a neuron agent cannot suppress such a situation, it asks its temporally closest friend neurons. This way, the feedback propagates through the network.

3.4 Implementation

The model is implemented using RePast Symphony 2.0.0 beta, an agent-based simulation environment written in Java (North et al. 2006). The implemented model is then verified and validated by using the model testing framework given in (Gürcan et al. 2011, 2013). The dynamic tuning of the strengths of synapses, is implemented using the parameter evolution technique described in (Lemouzy et al. 2011).

For the statistical calculations, the SSJ[14] (Stochastic Simulation in Java) library is used. To increase the reliability of the motoneuron agent, the goodness-of-fit test for the tonic firing behavior is also performed using the aforementioned testing framework (Gürcan et al. 2011, 2013).[15]

4 Simulation results

In this section, we present the experimental setups and the results in order to evaluate the proposed computational model.

4.1 Ethical statement

The data are taken from experiments at Kemal S. Türker's laboratory at Ege University. Ethical approval for the procedures of these experiments are taken from the Human Ethics Committee, Izmir, Turkey.

4.2 General configurations

To test our model we have chosen to simulate the neuronal circuitry of single motor units using the data obtained from low-threshold stimulation experiments on human soleus and tibialis-anterior muscles. In these experiments, despite rhythmic stimulation, the stimuli are randomly distributed within the ISI due to the variability of motoneuron firing. The interstimulus intervals were chosen from 1000 ms to 3000 ms random intervals so that the after-effects of the synchronization induced by a stimulus disappears before the delivery of the next stimulus.

The exact information we used about the underlying pathways is that *sensory neurons make monosynaptic connections with the alpha motoneuron*. In this sense, we considered this path as the shortest path in the underlying network and defined its duration as l. Therefore, we initialized the simulations as $\mathcal{N}(0) = \{s, m\}$, $\mathcal{S}(0) = \{\{s, m\}, \{m, \varnothing\}\}$ and $d_{sm} = d_{m\varnothing} = l/2$ where $s \in \mathcal{K}$, $m \in \mathcal{M}$ and l is the latency of the estimated beginning of the pathway extracted from the PSF-CUSUM of the reference experimental data by the simulation scientist. In this sense, l_{begin} is set to l since the earliest stimulus-evoked change of the motoneuron behavior can be observed at l and l_{end} is set to 200.0 ms (see Section 2.2). The viewer agent ν stimulates all sensory neuron agents $s \in \mathcal{K}$ using from 1000 ms to 3000 ms random intervals as it is the case for the real experiments. Moreover, the tolerance value τ, which is used by ν to compare the ouputs, is set to 0.5.

Although it is well known that neurons need to receive thousands of synapses to be able to cross their firing threshold, interneurons are most of the time ready to react to disturbances that could raise a reflex (Capaday 2002; Lam and Pearson 2002; Misiaszek 2006). Consequently, we allowed the AHP level of interneurons a low value (1.0 mV), so that they can activate with a small number of synapses. This choice also increased the convergence speed of the model. Although, it is unclear that this should be the case for real single motor units, it does not affect the recruitment of presynaptic neuron agents of the motoneuron agent.

Besides, in several intracellular studies of tonically active motoneurons (e.g., (Calvin and Schwindt 1972; Schwindt and Crill 1982)), it has been reported that the amplitude of AHP is 10 mV. Consequently, we allowed the AHP level of interneurons to be 10.0 mV.

Since the minimum processing time, the synaptic processing time d^{den}, is 0.5 ms in the model, the simulation of the model proceeds 0.5 ms time steps. From the same reason, the bin size for the PSF diagrams is also set to 0.5 ms.

---

[13]The idea here is quite similar to spike-timing-dependent plasticity (Song et al. 2000) since neurons are trying to increase the temporal correlations between their spikes and the spikes of their presynaptic neurons.

[14]http://www.iro.umontreal.ca/~simardr/ssj/indexe.html, last access on 1 April 2013.

[15]The case study given in (Gürcan 2013) is comprehensively describing this test.

4.3 Simulation experiments

Before assessing the macro-level behaviors of simulations, we first need to be sure that the simulation of the motoneuron agent works fine. To this end, we plotted the ISI distributions for a human alpha-motoneuron and its simulation (Fig. 5). The distribution of ISIs has nearly identical mean and variance and the right-skewed shape occurs in both distributions. The distribution parameters were set to achieve these distributions. Once we ensure that the motorneuron agent is able to discharge at similar characteristics as the real human motor neuron, we can perform simulations for human single motor unit (SMU) pathways.

For illustration, we simulated the model by using 9 different human SMU pathways with different *macro-level* functional behaviors. As shown in Table 1, the data about these pathways are obtained with various numbers of trials (Trial N.) and various total number of consecutive empty bins (Tot. C.E.B.). In order to perform reliable simulation experiments, we separated these human data into training data and test data in terms of trials. In this sense, for each simulation experiment, we first divided each human data into *2n* trial segments and then select randomly *n* trial segments for forming training data and used the remaining *n* as test data. Training data are then given to the Viewer agent as reference data. In our simulation experiments, we have chosen *2n=10* and used the human data whose number of trials are more than 200 since the PSF better represents the profile of the postsynaptic potential caused by the stimulus when there are at least 100 trials (Türker and Powers 2005). After the simulations end, the results are analyzed in order to ensure that the generated networks are functionally equivalent to the reference real networks. To calculate the similarity of two networks, a Pearson-correlation analysis is performed between the simulated and the test PSF-CUSUMs. This correlation is a proxy for evidence that reflects the accuracy of the model in terms of its coefficient of determination. The correlation r yields a 0 when there is no correlation (totally uncorrelated) and a 1 for total correlation (totally correlated). The degree of similarity is then calculated as r^2 (M. Sim. in Table 1). This information is sufficient to claim that the two underlying networks are *functionally equivalent* in the sense of PSF analysis since our objective was to obtain a response pattern from an artificial single motor unit that is comparable with the response of the real human single motor unit.

For each human neuronal pathway (HNP), simulation experiments were repeated 10 times (Table 1). The 6 outputs of these experiments are shown in Fig. 6. Similarities of PSF-CUSUM diagrams show that the model can mimic HNPs with different macroscopic patterns. In other words, an initial network successfully converges to a solution network whose macroscopic behavior conforms to its reference biological network.

When a network finishes organizing itself, the correct organization of neurons is said to be found. Now the question is, are we sure that the emergent neural network continues adequately generating the macro-level functional behavior with its reference HNP? In our case, where stimulus-evoked changes on motoneuron discharge rates are monitored, we fed the emergent network with the similar random input during its learning process (with an inter-stimulus interval of 1000 ms to 3000 ms as described in Section 4.2). What we expected was a fairly similar *macroscopic* output (not exactly the same of course because of the noise) to its reference biological network. We performed such tests by continuously stimulating the final networks for a long time. An illustration of this test for the SS-1-1-1 data is shown in Fig. 7.

The detailed results for one of the simulation experiments for SS-1-1-1 data are shown in Fig. 8. At the end of the simulation, there is a strong positive correlation between PSF-CUSUMs (Fig. 8a). To provide this similarity the model creates excitatory and inhibitory synapses on the
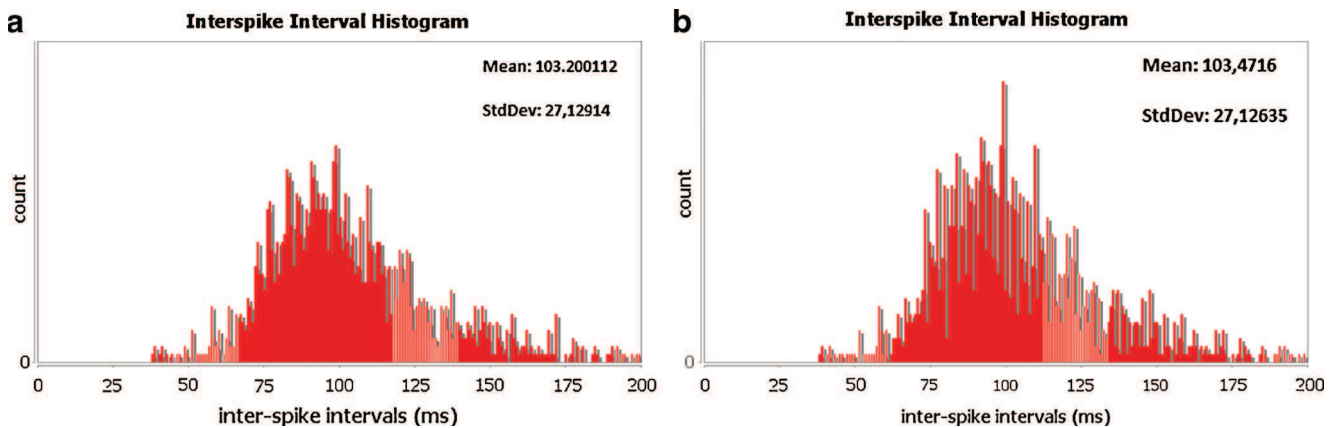

Fig. 5 Inter-spike-interval (ISI) probability distributions of (**a**) a human alpha-motoneuron and (**b**) its simulation

Table 1 Results for simulation experiments of 9 different human neuronal reflex pathways (HNP) obtained from soleus and tibialis-anterior muscles with various numbers of trials (Trial N.) and total number of consecutive empty bins (Tot. C.E.B.). For each HNP, simulations were conducted 10 times and, the mean similarity obtained (M. Sim.), the mean number of excitatory (Exc. N.) and inhibitory neurons (Inh. N.), and the mean total number of neurons (Tot. N.) are presented

HNP	Trial N.	Tot. C.E.B.	M. Sim. (%)	Exc. N.	Inh. N.	Tot. N.
AO-2-3-1	303	13	77.62	411	138	549
BU-1-1-1	323	50	93.49	366	6	372
BU-1-3-1	232	43	96.14	348	0	348
NK-3-2-1	697	28	96.16	355	159	514
OS-3-1-1	510	88	83.78	298	0	298
OS-3-3-1	510	88	86.84	340	7	347
OS-4-2-1	987	25	94.69	317	19	336
OS-4-3-1	986	45	91.51	359	13	372
SS-1-1-1	783	0	93.81	391	169	560
mean	592.3	42.2	90.44	353.8	56.7	410.6

motoneuron agent. The number and temporal distribution of these synapses are plotted in Fig. 8b. Another striking feature of the model, illustrated in Fig. 8c, is that it is able to extract the presynaptic activity on a motoneuron. This activity is not uniform, but consists of multiple temporal PSPs that appear and disappear at various times.

Besides, we are able to plot and analyze generated neuronal networks by exporting their graph representation via GraphML language.[16] Then Gephi (Bastian et al. 2009) is used to visualize this graph[17] as shown in Fig. 8d.

5 Discussion and conclusions

5.1 Biological interpretation

The model presented here uses integrate-and-fire neurons and PSF reflex recordings together with artificial self-organization, to the best of our knowledge for the first time, for generating an artificial neuronal network that mimics its reference biological neuronal network. We used reflex responses of human single motor units not only because we could record them during conscious contractions but also as they directly represent the discharges of motoneurons in the human spinal cord. To be able to show that the model is robust to biological variability, it has been rigorously tested as described in the results section.

The results show that the developed model is able to learn and simulate the functional behavior of human single motor units. The mean similarity observed is between 77.62 % and 96.16 %, the mean number of neuron agents is between

298 and 560 (Table 1). For most of the data sets the mean similarity is more than 90 %. This success indicates that the proposed computational neuronal network model is a potential candidate for mimicking neuronal networks. However, for some data sets it is contrary and the model is unable to mimick the functional behavior in that success. The worst one is AO-2-3-1 with an average similarity of 77.62 % and its ouput is shown in Fig. 6a. This situation is most probably caused by higher level of synaptic noise in those motoneuron membranes. This noise makes much harder for the viewer agent to give the right feedback. In such a case, the underlying SMU must be stimulated much more while collecting data from the human subject since noise is inversely proportional to the stimulation count in PSF-CUSUM diagrams. The other relatively worse simulation results belong to OS-3-1-1 with an average similarity of 83.78 % and OS-3-3-1 with an average similarity of 86.84 % (see Fig. 6b and c respectively). In these experiments, the numbers of stimulations are higher and thus there is less noise. However, unlike AO-2-3-1 and other HNPs, the experimental data contain the highest number of consecutive empty bins. Since these bins provide no information to the viewer agent, it is harder to make the correct evaluations.

After attaining functional adequacy – in terms of predicting empirical responses, the structure of the resulting networks can be analyzed to see to what extent they are biologically plausible. In this sense, the first criterion can be the number of neurons. As seen from Table 1, the average number of neurons observed at the end of simulations is not an implausible value for a human single motor unit pathway (SMU). One may think that it is necessary to justify these numbers – because such numbers can be judged too far from real values. However, the biological constraints given in the model do not currently prevent creation of such a number of neurons. On the other hand, we know that each neuron

[16] http://graphml.graphdrawing.org/, last access on 1 April 2013.

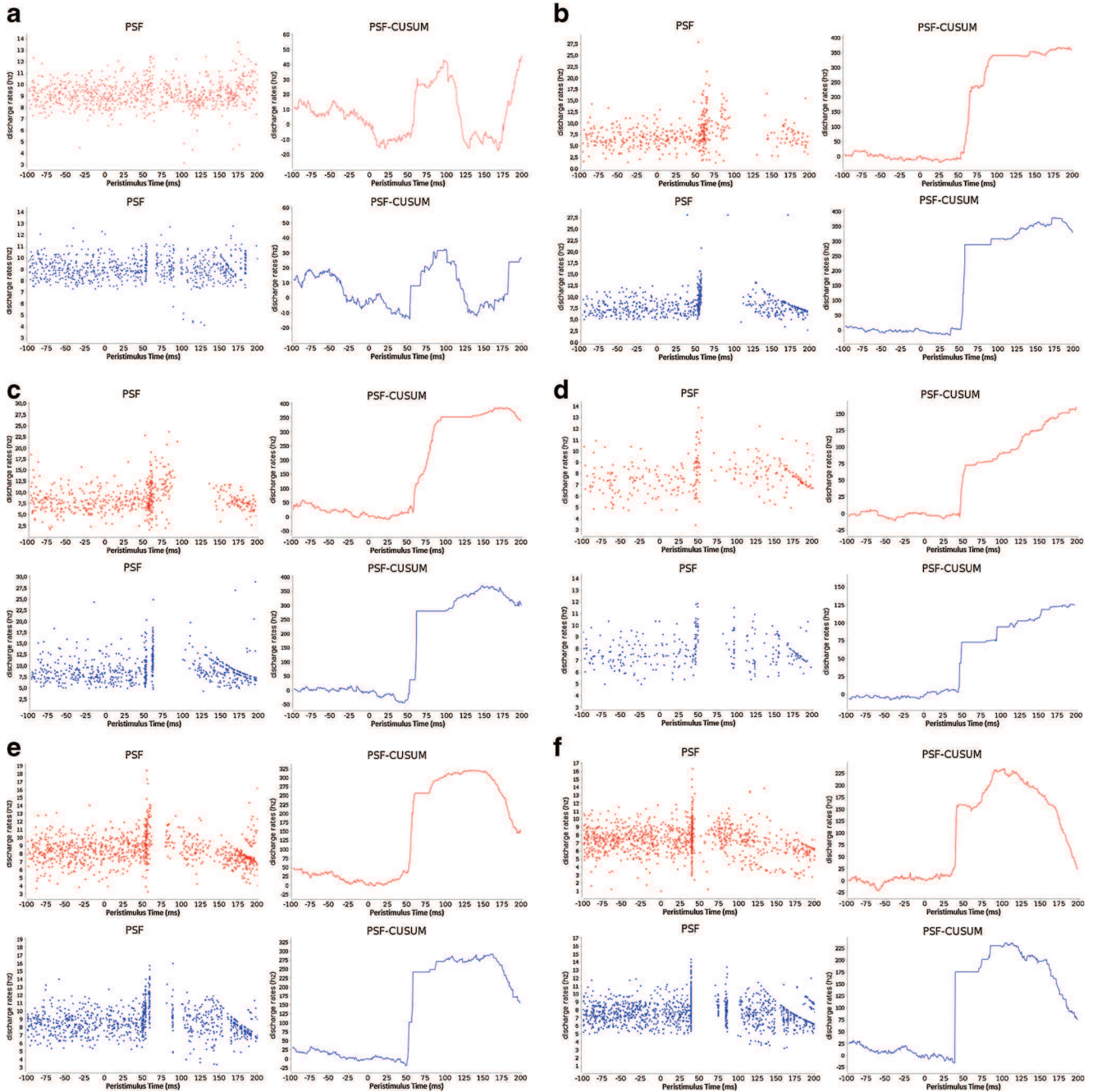[17] https://gephi.org/, last access on 1 April 2013.

Fig. 6 The reference (*red*) and resulting simulated (*blue*) PSF and PSF-CUSUM diagrams for 6 out of 9 simulation experiments of human neuronal pathways. Reference data are compiled from the reflex response of the soleus muscle motor units. The simulated data are compiled from the motoneuron agent responses of a simulation experiment. The results (**a**), (**b**) and (**c**) present the three worst similarities obtained, and the result (**d**), (**e**) and (**f**) show the three best similarities obtained. They belong to human data AO-2-3-1, OS-3-1-1, OS-3-3-1, BU-1-3-1, NK-3-2-1 and SS-1-1-1 respectively

is created for a biological reason. Thus, we think that these numbers may relate to the possible long loops in the reflex pathways since it is known that some of the reflex responses of the human motor units occur at longer latencies as it is shown in Fig. 6. In this sense, what is remarkable here is that

the simulations give us such possible long loops (in Fig. 8d they are represented as interneuronal loops made from yellow dots) that could account for these longer latency reflex events, even though they were not a priori defined in the model (initially there was only a sensory neuron agent and
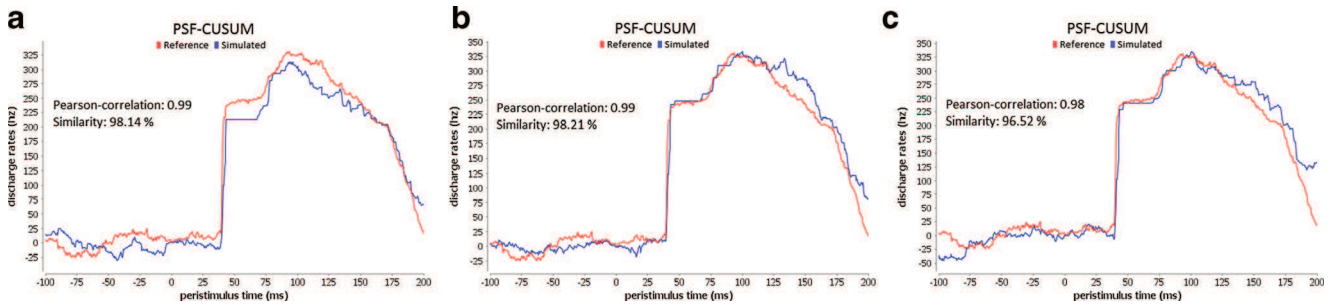
Fig. 7 (**a**), (**b**) and (**c**) illustrate there separate tests performed after the convergence of SS-1-1-1 data. These tests show that after the network finishes organizing itself, it retains the same macro-level behavior

a motoneuron agent). Lengths of these loops possibly represent (1) the latencies of the reflex responses as the stimulus induced action potentials have to go through many interneurons to elicit the long latency reflex responses, or (2) the latency of an action potential travelling on one long axon to the cortex and on another long axon back from the cortex. If the latter is happening, the number of interneurons would decrease dramatically. However, at present this cannot be known without further human experiments.

Remembering that going across each synapse takes about 0.5 ms, for a reflex response that occurs at for example 20 ms after the first phase of the reflex action potential will have to jump through 40 synapses involving that many interneurons. In Fig. 8d therefore, there must be five separate reflex responses after the first reflex event.

Conceptually, an instant significant increase (and decrease) of PSF-CUSUM value indicates instant EPSPs (and IPSPs respectively) on motoneuron membrane. Moreover, since we restrict the PSP weights of individual synapses to the biological evidences on human studies, the number of excitatory and inhibitory synapses on the motoneuron are self-extracted by the simulation model (see Fig. 8b). However, we note that these numbers are only estimations and their correctness cannot be directly proven. For example, for some pathways the simulations do not find any inhibitory interneurons (see Table 1), which is biologically impossible. A possible explanation is that, there is a high level of synaptic noise in their data. The model is therefore unable to distinguish whether it is an inhibition or a noise when there is a decrease in the CUSUMs (see e.g., Fig. 6b and d).

In PSF technique, it is common practice to keep the number of trials as high as possible to ensure that the empty bins are not due to chance and really empty because of inhibitions. In other words, we can say that as the number of trials increases, the information about the PSF also increases. The results show that there is no correlation between the number of trials and the average success of the simulations so long as the number of trials is more than 116 ($232/2 = 116$) in training data (see BU-1-3-1 dataset Table 1).

5.2 Artificial self-organization

Artificial self-organization of the agents proceeds in a completely unsupervised way thanks to the AMAS theory. The NCSs of the agents is at the heart of this mechanism. Hence, giving the accurate feedback to the *right* agents is important. Without such a feedback mechanism, the networks would not converge and their internal representation would keep changing. Consequently, since the moving average of the PSF-CUSUM derivative is used as the feedback, the calculation of the PSF-CUSUM derivative must be performed carefully as it affects the correct modification procedure of the network. The calculation should be done without displacing the peaks and throughs significantly. During our preliminary investigations, it has been observed that when the PSF-CUSUM derivative is smoothed too much, the information about the dynamics of the system was lost. On the other hand, when the smoothing was not sufficient, the noise of the motoneuron agent prevented the interneuron agents to learn the right dynamics. Besides, the tolerance value τ is also an important parameter as it is closely related to the correct feedback. Similar to smoothing, when τ is higher, the information about the dynamics of the system is lost, and when τ is smaller, it is harder to detect good outputs.

5.3 Comparison with existing models

Our computational model can be contrasted to self-organizing and evolutionary network models that aim to find the right network structure, and existing effective connectivity investigations in the literature. On the one hand, comparisons with self-organizing and evolutionary neural network models are given to show what led us to design a new emergent neural network model since *emergence* is crucial to be able to define the transition from *micro-level* to the *macro-level* without relying on mathematical arguments. On the other hand, comparisons with existing effective connectivity studies are given to show that these techniques cannot be used for estimating the effective connectivity in *human* neuronal pathways using motor unit analysis.
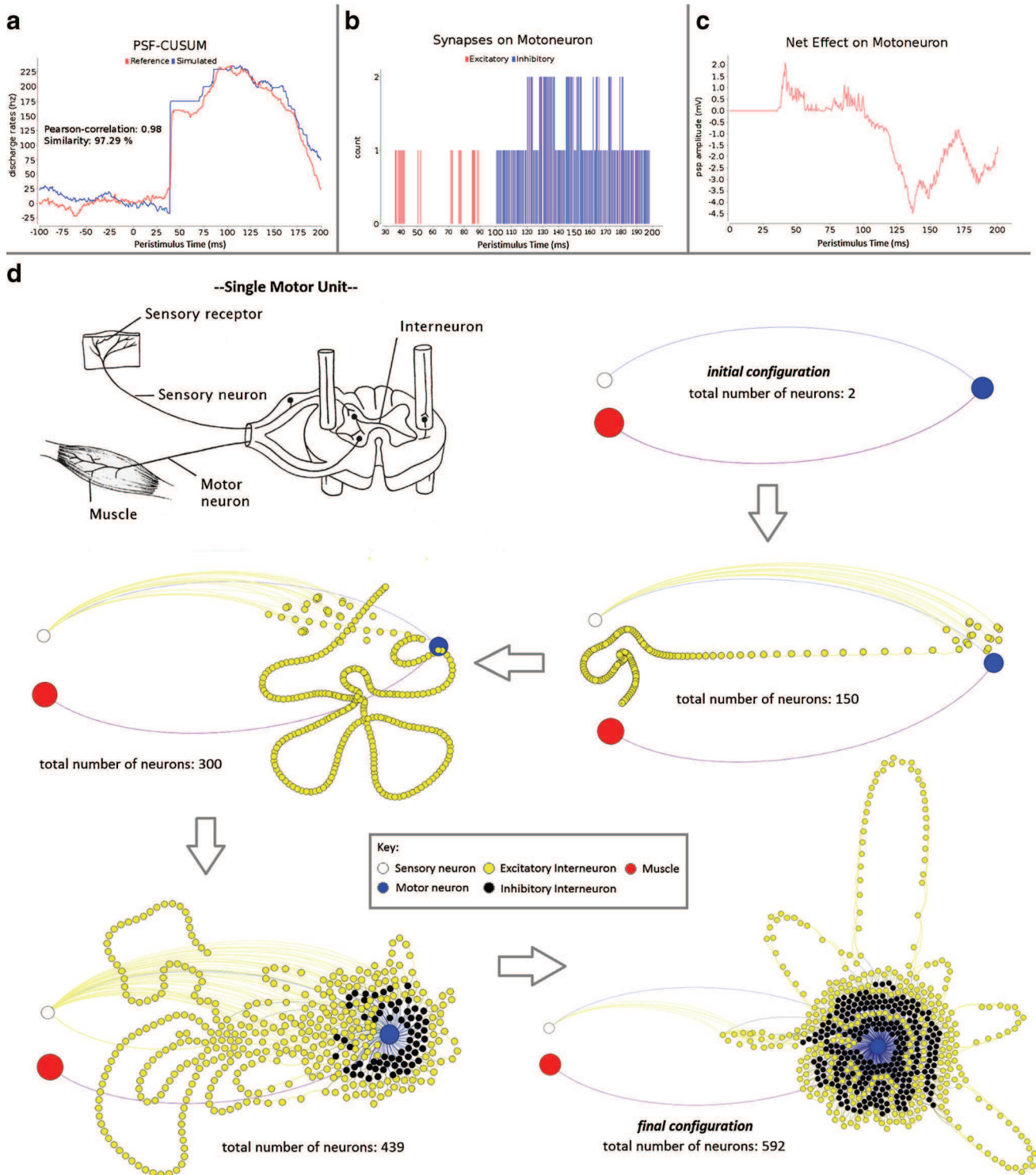
Fig. 8 This figure illustrates the results that came out of a simulation run at the end of its effort to learn the global pattern obtained from the human reflex experiment SS-1-1-1. (**a**) PSF-CUSUM diagrams of the reference data (*red line*) and its simulated replication (*blue line*). Pearson-correlation of these lines is 0.98 and thus their similarity is 97.29 %. (**b**) The temporal distribution of created excitatory (*red*) and inhibitory (*blue*) synapses on the motoneuron. (**c**) The net PSP on motoneuron caused by its presynaptic connections given in (**b**). (**d**) The cinematic representation of the evolution of the neural network from the initial configuration towards the final configuration together with the number and the sign (excitatory or inhibitory) of neurons that came throughout the simulation run. Note also that, in the final configuration, the extent of the pathways that represent the long latency reflex responses are emerging as neuronal loops in the figure. The big *red dot*, the big *white dot*, the *blue dot*, the small *black dots* and the small *yellow dots* illustrate respectively the muscle, the sensory receptor, the motor neuron, the inhibitory interneurons and the excitatory interneurons

5.3.1 Self-organizing neural networks

There are various self-organizing neural network models in the literature that do not have a predefined structure and size. Most of these models add new neuron(s) to support the neuron that has accumulated the highest error during previous iterations or to support topological structures (Villmann et al. 1997; Fritzke 1994; Fahlman and Lebiere 1990). In these models, new neuron(s) are added every λ iterations, where λ is a constant. Apart from these models, (Marsland et al. 2002) proposed another network model that grows when required (GWR). In GWR, both the nodes (neurons) and edges (synapses) can be created and destroyed during the learning process. Rather than adding a new neuron after every λ inputs, new neurons can be added at any time. The new neurons are positioned dependent on the input and the current best neuron, rather than adding them where the accumulated error is the highest. In this sense, although GWR is an unsupervised algorithm, it needs access to global information unlike our model (since selecting the best matching neuron is not possible from the point of view of a single neuron). Such a global view prevent the system being emergent, unlike our model. New synapses, on the other hand, are added if there is not a connection between the best matching neuron, and the second best. In GWR, for removal of synapses, they have an associated "age". This is originally set to zero, and is incremented at each time step for each synapse that is connected to the winning neuron. Synapses that exceeds some constant age_{max} are removed from the network. The removal of synapses seems quite similar to our model in the sense that there is a threshold, however both addition and removal of synapses in GWR requires a global view of the system.

Besides, there are self-organizing neural network models that use biological rules for changing the organization of the network. (Lazar et al. 2009) propose a self-organizing recurrent neural network model, SORN, that combines three distinct forms of biological local plasticity[18] to learn spatio-temporal patterns. In SORN, initially connections are sparse and random with a degree of probability. Moreover, direct connections between inhibitory neurons are not present. (Martin and Reggia 2010), on the other hand, focus on the self-assembly of recurrent neural network architectures using swarm intelligence. Their method involves network growth in a three-dimensional space, and cell migration and division, along with axon-axon interactions, play an important role in the growth process. However, rather than functionality, they are more focused on self-assembly in physical space involving geometrical relations.

---

[18]STDP (Song et al. 2000), synaptic scaling of the excitatory-excitatory connections (Turrigiano et al. 1998), and intrinsic plasticity regulating the thresholds of excitatory units.

5.3.2 Evolutionary neural networks

Evolutionary systems consist of machine learning optimization and classification paradigms that are roughly based on evolution mechanisms such as biological genetics and natural selection. In a typical neuroevolutionary system (Yao 1999; Floreano et al. 2008; Stanley 2004), the weights of a neural network are strung together to form an individual genome. A population of such genomes is then evolved by evaluating each one and selectively reproducing the fittest individuals through crossover and mutation. Most neuroevolutionary systems require the designer to manually determine the networks topology (i.e. how many hidden nodes there are and how they are connected). In contrast, (Stanley and Miikkulainen 2002; Stanley 2004) proposed NeuroEvolution of Augmenting Topologies (NEAT) approach in order to automate the search for appropriate topologies and initial weights of neural network function approximators. NEAT automatically evolves the topology to fit the complexity of the problem by combining the usual search for network weights with evolution of the network structure. However, NEAT does not attempt to learn a value function. Instead, it finds good policies directly by training *action selectors*, which map states to the action the agent should take in that state.

However, to create networks that will adapt to the problem at hand, autonomously, without knowing the capacity (in terms of neurons) required to handle the complexity of the problem, we need *evolutionary function approximation* methods. To this end, there are various methods in the literature (Chatzidimitriou and Mitkas 2013; Roeschies and Igel 2010; Whiteson and Stone 2006). Basically, these methods select automatically function approximator representations that enable efficient individual learning. They begin with an initial network and *evolve* individuals that are better able to *learn*. This evolution takes place with some degree of probability. Neurons can be removed through crossover if for example the fittest network is chosen from the two parents to hold the offspring and it happens to be the smaller of the two parents. Synapses can also be removed through crossover or they can be removed in order to adjust the network to the density defined in the genome which is (number of active connections / N^2), where N is the number of neurons in the network.

Our model differs from NEAT studies in several aspects. First of all, there is neither randomness nor probability in our model. Each neuron and synapse are added and/or removed due to NCSs of agents. Secondly, although like NEAT models, our model begins with a simple initial network, unlike them, this is not for minimizing the dimensionality of the network. Rather, it is because of the lack of our knowledge about the underlying reference neural network. Minimizing the dimensionality in our model can

only be provided by defining NCSs in a better way. Finally, our work differs from existing NEAT works in that (a) it applies cooperation rules to make the structure of the network evolve, (b) our focus is mainly on biological neural networks rather than artificial neural networks and thus (c) it uses an established neuroscience method (PSF) to extract local information. So, at least to the best of our knowledge, such a synergy has not been examined before.

5.3.3 Dynamic causal modeling

In the neurosciences, a common way to model neuronal circuits - given metabolic or electrophysiological data - is to use Dynamic Causal Modeling (DCM) (Friston et al. 2003). In DCM, biologically plausible generative models are optimized to best explain observed neuronal responses. For example, DCM for event-related potentials (ERP) tries to explain ERPs that are formally similar to the peristimulus frequency-grams used in our work (Garrido et al. 2007; David et al. 2006; Kiebel et al. 2006). Crucially, the objective of DCM is to find the model with the greatest evidence; where evidence is the probability of the data given a model. This usually involves exploring large model spaces by adding or subtracting nodes and edges from graphical models of neuronal circuits generating observed responses. At present, this exploration of model space is usually specified by hand. In contrast, the *artificial self-organization* procedures we have described in this article provide an automatic and principled way to perform greedy searches over model space. It is also possible that the same objective function used in DCM – the evidence for a particular model – could be employed in *artificial self-organization*.

5.3.4 Effective connectivity investigations

Buibas and Silva (2011) present a formal modeling framework for using real-world data to map the functional topology of complex dynamic networks. The framework formally defines key features of cellular neural network signalling and experimental constraints associated with observation and stimulus control, and can accommodate any appropriate model intracellular dynamics. They claim that, the framework is particularly well-suited for estimating the functional connectivity in biological neural networks from experimentally observable temporal data. However, the framework is unable to estimate and map the effective topology of complex networks with unknown connectivities. There are also approaches for estimating parameters and dynamics of small groups of neurons (Eldawlatly et al. 2010; Makarov et al. 2005), Eldawlatly et al. (2010) use dynamic bayesian networks in reconstructing functional neuronal networks from spike train ensembles. Their model can discriminate between mono- and polysynaptic links between spiking

cortical neurons. Makarov et al. (2005) present a deterministic model for neural networks whose dynamic behavior fits experimental data. They also use spike trains and their model permits to infer properties of the ensemble that cannot be directly obtained from the observed spike trains. However, these studies are far from establishing a complete effective connectivity of larger networks.

5.4 Limitations of the model

The threshold-crossing model, used for the simulations presented in this paper, is a simple one and does not include all membrane mechanisms, which may influence motor neuron excitability. The AHP time course is described simply as a straight line although in reality the membrane potential reaches the threshold in the random walk process where noise increases with time (Warren et al. 1992). In this sense, for increasing the success, the reality and the reliability, the motoneuron models which are verified by the experimental results obtained from human experiments, like the one proposed in (Kuraszkiewicz et al. 2012), should be used.

Besides, in this work, we have simply used the PSF-CUSUM correlation between the observed and predicted responses as a proxy for the quality of the model produced. Although PSF analysis is certainly 'a' measure of network behavior, there are lots of other possibilities as well, and just computing a correlation on this value does not allow us to claim that the emergent model and its underlying HNP are functionally equivalent. This does not accommodate model complexity that is an inherent part of model evidence. In other words, in future work, we could consider not just a measure of accuracy or correlation but a penalized measure such as the Akaike information criteria (Akaike 1974) or free energy bounds on Bayesian model evidence (Friston 2009).

In addition, the model presented in this article is only based on temporal constraints: no steric regulation of proliferation due to physical contact between membrane, neither any local chemical regulation of neurons activity due to their proximity with glial cells (Takata and Hirase 2008). First mechanism, inspired from morphogenesis modelling as in (Marsland et al. 2002), should help neurons in finding right accointancies and so increase significantly the number of cycles necessary for the network to get a correct functional structure. Moreover, *in vivo*, neurons are not the only cells implied in information treatment (Perea and Araque 2010). Glial cells are known to regulate synaptic plasticity, this regulation can be integrated in modeled synapses, what is done today, but what is still to construct is local transmission of regulation from synapses to groups of neurons in a glial neighborhood that may be different from a strict functional neuronal neighborhood (Fellin 2009).

5.5 Conclusions

So far, our simulations display dynamics with strong *macro-* and *micro-level* functional similarities to real human single motor unit pathways. Since the theory used at the *micro-level* is not mathematically related to the *macro-level* functionality, all resulting networks generated by our model can be qualified as *emergent*. Furthermore, these networks can be regarded as biologically plausible, because the nominal behaviors of the agents conform to their corresponding biological elements (they are all verified and validated). Hence, the results allowed us to make predictions about the underlying neuronal pathways. However, these findings do not constitute a proof that the simulated neural networks are exactly as their reference biological networks.

Consequently, it is not currently possible to generalize our model as a *generic facility that could allow mimicking any part of neuronal pathways*. Such generalization can only be obtained when a particular simulation scenario does not fit the data at hand. This is why model optimization based upon model evidence (as in Dynamic Causal Modeling) penalizes overly complicated models with too many degrees of freedom. Clearly, the next step in our work is to assess the generalization of our computational model. This would normally involve seeing how well the model predicted responses to new patterns of stimulation. We have not really addressed this issue in the current work. We hope to be able to use other data in future analyses to address (and possibly optimize) generalization; for example, optimizing one simulation scenario given the observed responses to a single stimulation pulse (like we did in this article) and then seeing how well it predicted responses to another scenario; for example a pair of stimulation pulses that were separated by a short interval. In terms of scenario selection, this is known as establishing predictive validity and usually goes hand in hand with simulation scenarios that have high evidence.

Moreover, more complex pathways (like multiple motor unit pathways) have not yet been validated. In other words, it is not yet clear whether our computational model can perform the same practical functions as the real larger-scale pathways they model. To this end, we need two things: (1) the discharge rates of multiple motor units obtained from the same human subject and (2) more NCSs related to biological situations to increase biological reality (e.g., introducing spatial constraints). Apparently, the more biological information is provided to the model, the more biologically correct the resulting neuronal networks would be. The envisioned next steps involve expansion of the project in multiple directions, some of which are related to surmounting this limitation.

As a result of this perspective, we are planning to achieve a *generic model that could allow mimicking any part of (human) neuronal pathways, if the PSF data can be measured and provided according to specifications*. The facility can then be used to construct the artificial replication of any neuronal pathway, consisting of real neurons.

Conflict of Interest The authors declare that they have no conflict of interest.

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, *19*(6), 716–723.

Bastian, M., Heymann, S., Jacomy, M. (2009). Gephi: an open source software for exploring and manipulating networks. In *International AAAI conference on weblogs and social media (ICWSM 2009)*.

Bernon, C., Capera, D., Mano, J.-P. (2009). Engineering self-modeling systems: application to biology. In A. Artikis, G. Picard, L. Vercouter (Eds.)*, Engineering societies in the agents world IX. Lecture notes in computer science* (Vol. 5485, pp. 248–263). Berlin Heidelberg: Springer.

Bonabeau, E., Dorigo, M., Theraulaz, G. (1999). *Swarm intelligence: from natural to artificial systems*. Santa Fe Institute Studies in the Sciences of Complexity. OUP USA.

Brax, N., Andonoff, E., Gleizes, M.-P. (2012). A self-adaptive multi-agent system for abnormal behavior detection in maritime surveillance (regular paper). In *KES international conference (KES-AMSTA), Dubrovnik* (Vol. 7327, pp. 174–185). New York: Springer.

Brinkworth, R.S.A., & Türker, K.S. (2003). A method for quantifying reflex responses from intra-muscular and surface electromyogram. *Journal of Neuroscience Methods*, *122*(2), 179–193.

Buibas, M., & Silva, G.A. (2011). A framework for simulating and estimating the state and functional topology of complex dynamic geometric networks. *Neural Computation*, *23*(1), 183–214.

Calvin, W., & Schwindt, P. (1972). Steps in production of motoneuron spikes during rhytmic firing. *Journal of Neurophysiology*, *35*, 297–310.

Camps, V., Gleizes, M.P., Glize, P. (1998). A self-organization process based on cooperation theory for adaptive artificial systems. In *1st international conference on philosophy and computer science: processes of evolution in real and virtual systems*. Krakow, Poland.

Capaday, C. (2002). The special nature of human walking and its neural control. *Trends in Neurosciences*, *25*(7), 370–376.

Capera, D., George, J.-P., Gleizes, M.-P., Glize, P. (2003). The amas theory for complex problem solving based on self-organizing cooperative agents. In *Proceedings twelfth IEEE international workshops on enabling technologies: infrastructure for collaborative enterprises (WETICE 2003)* (pp. 383–388).

Capera, D., Gleizes, M.-P., Glize, P. (2004). Mechanism type synthesis based on self-assembling agents. *Journal of Applied Artificial Intelligence*, *18*(9–10), 921–936.

Chatzidimitriou, K.C., & Mitkas, P.A. (2013). Adaptive reservoir computing through evolution and learning. *Neurocomputing*, *103*, 198–209.

Colquhoun, D., & Hawkes, A.G. (1990). Stochastic properties of ion channel openings and bursts in a membrane patch that contains two channels: evidence concerning the number of channels present when a record containing only single openings is observed. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, *240*(1299), 453–477.

Combettes, S., Sontheimer, T., Rougemaille, S., Glize, P. (2012). Weight optimization of aircraft harnesses (short paper). In *International conference on practical applications of agents and multiagent systems (PAAMS), Salamanca* (pp. 229–232). Springer-Verlag.

David, O., Kiebel, S.J., Harrison, L.M., Mattout, J., Kilner, J.M., Friston, K.J. (2006). Dynamic causal modeling of evoked responses in eeg and meg. *NeuroImage*, *30*(4), 1255–1272.

Dayan, P., & Abbott, L. (2005). *Theoretical neuroscience: computational and mathematical modeling of neural systems*. Computational Neuroscience. MIT Press.

de Haan, J. (2006). How emergence arises. *Ecological Complexity*, *3*(4), 293–301.

Eldawlatly, S., Zhou, Y., Jin, R., Oweiss, K.G. (2010). On the use of dynamic bayesian networks in reconstructing functional neuronal networks from spike train ensembles. *Neural Computation*, *22*(1), 158–189.

Ellaway, P. (1978). Cumulative sum technique and its application to the analysis of peristimulus time histograms. *Electroencephalography and Clinical Neurophysiology*, *45*(2), 302–304.

Fahlman, S.E., & Lebiere, C. (1990). *Advances in neural information processing systems 2. chapter the cascade-correlation learning architecture* (pp. 524–532). San Francisco: Morgan Kaufmann Publishers Inc.

Fellin, T. (2009). Communication between neurons and astrocytes: relevance to the modulation of synaptic and network activity. *Journal of Neurochemistry*, *108*(3), 533–544.

Floreano, D., Dürr, P., Mattiussi, C. (2008). Neuroevolution: from architectures to learning. *Evolutionary Intelligence*, *1*(1), 47–62.

Friston, K. (2009). The free-energy principle: a rough guide to the brain? *Trends in Cognitive Sciences*, *13*(7), 293–301.

Friston, K.J. (2011). Functional and effective connectivity: a review. *Brain Connectivity*, *1*(1), 13–36.

Friston, K., Harrison, L., Penny, W. (2003). Dynamic causal modelling. *NeuroImage*, *19*(4), 1273–1302.

Fritzke, B. (1994). A growing neural gas network learns topologies. In *NIPS* (pp. 625–632).

Garrido, M.I., Kilner, J.M., Kiebel, S.J., Stephan, K.E., Friston, K.J. (2007). Dynamic causal modelling of evoked potentials: a reproducibility study. *NeuroImage*, *36*(3), 571–580.

Georgé, J.P., Edmonds, B., Glize, P. (2004). *Making self-organizing adaptive multi-agent systems work, chapter 16* (pp. 321–340). Kluwer Publishing.

Georgé, J.-P., Gleizes, M.-P., Camps, V. (2011). Cooperation. In G. Serugendo, M.-P. Gleizes, A. Karageorgos (Eds.), *Self-organizing software: from natural to artificial adaptation. Natural computing series,* (chap. 9, pp. 347–377). Springer.

Gerstner, W., Sprekeler, H., Deco, G. (2012). Theory and simulation in neuroscience. *Science*, *338*(6103), 60–65.

Goldstein, J. (1999). Emergence as a construct: history and issues. *Emergence : A Journal of Complexity Issues in Organizations and Management. The New England Complex Systems Institute*, *1*(1), 49–72.

Grande, G., & Cafarelli, E. (2003). Ia afferent input alters the recruitment thresholds and firing rates of single human motor units. *Experimental Brain Research*, *150*, 449–457.

Guivarch, V., Camps, V., Péninou, A. (2012). Amadeus: an adaptive multi-agent system to learn a user's recurring actions in ambient systems. *Advances in Distributed Computing and Artificial Intelligence Journal, Special Issue no 3*.

Gürcan, Ö., Dikenelli, O., Türker, K.S. (2010a). Agent-based exploration of wiring of biological neural networks: position paper. In R. Trumph (Ed.)*, 20th european meeting on cybernetics and systems research (EMCSR 2010)* (pp. 509–514). Vienna.

Gürcan, Ö., Dikenelli, O., Türker, K.S. (2010b). Understanding wiring of human cns using agent-based modeling and simulation. *Turkish Journal of Neuroscience*, *16*(1), 219–220.

Gürcan, Ö., Dikenelli, O., Bernon, C. (2011). Towards a generic testing framework for agent-based simulation models. In *5th international workshop on multi-agent systems and simulation (MAS&S'11)* (pp. 635–642). Szczecin.

Gürcan, Ö., Bernon, C., Türker, K.S., Mano, J.-P., Glize, P., Dikenelli, O. (2012). Simulating human single motor units using self-organizing agents. In *Self-adaptive and self-organizing systems (SASO), 2012 IEEE sixth international conference on* (pp. 11–20).

Gürcan, Ö., Dikenelli, O., Bernon, C. (2013). A generic testing framework for agent-based simulation models. *Journal of Simulation (JOS)*, 1–19. http://www.palgrave-journals.com/jos/journal/vaop/ncurrent/full/jos201226a.html.

Gydikov, A., Tankov, N., Gerilovsky, L., Radicheva, N. (1977). Motor unit activity upon polysynaptic reflex in man. *Agressologie*, *18*(2), 103–108.

Hay, E., Hill, S., Schürmann, F., Markram, H., Segev, I. (2011). Models of neocortical layer 5b pyramidal cells capturing a wide range of dendritic and perisomatic active properties. *PLoS Computational Biology*, *7*(7), e1002107.

Heylighen, F. (1999). The science of self-organization and adaptivity. In *Knowledge management, organizational intelligence and learning, and complexity, in: the encyclopedia of life support systems, EOLSS* (pp. 253–280). Publishers Co. Ltd.

Hodgkin, A.L., & Huxley, A.F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, *117*(4), 500–544.

Iansek, R., & Redman, S. (1973). The amplitude, time course and charge of unitary post-synaptic potentials evoked in spinal motoneurone dendrites. *Journal of Neurophysiology*, *234*, 665–688.

Izhikevich, E.M., & Edelman, G.M. (2008). Large-scale model of mammalian thalamocortical systems. *Proceedings of the National Academy of Sciences*, *105*(9), 3593–3598.

Jenner, J.R., & Stephens, J.A. (1982). Cutaneous reflex responses and their central nervous pathways studied in man. *The Journal of Physiology*, *333*(1), 405–419.

Kaddoum, E., & Georgé, J.-P. (2012). Collective self-tuning for complex product design (short paper). In *IEEE international conference on self-adaptive and self-organizing systems (SASO), Lyon - France*, page (electronic medium). CPS (Conference Publishing Services).

Kandel, E.R., Schwartz, J., Jessell, T.M. (2000). *Principles of neural science, 4th edn*. Mc Graw Hill.

Kiebel, S.J., David, O., Friston, K.J. (2006). Dynamic causal modelling of evoked responses in eeg/meg with lead field parameterization. *NeuroImage*, *30*(4), 1273–1284.

Koch, C., & Segev, I. (1989). *Methods in neuronal modeling: from synapses to networks*. Computational Neuroscience Series. Mit Press.

Koestler, A. (1967). *The ghost in the machine*. London, the danube ed. edition: Hutchinson and Co.

Kuraszkiewicz, B., Mlozniak, D., Piotrkiewicz, M. (2012). Threshold-crossing model of human motoneuron. In E. Pitka, & J. Kawa (Eds.), *Information technologies in biomedicine. lecture notes in computer science* (Vol. 7339, pp. 209–218). Berlin Heidelberg: Springer.

Lacouture, J., Rodriguez, I., Arcangeli, J.-P., Chassot, C., Desprats, T., Drira, K., Garijo, F., Noel, V., Sibilla, M., Tessier, C. (2011). Mission-aware adaptive communication for collaborative mobile entities. In M.M. Cruz-Cunha, & F. Moreira (Eds.), *Handbook of research on mobility and computing: evolving technologies and ubiquitous impacts* (chap. 64, pp. 1056–1076). IGI Global.

Lam, T., & Pearson, K. (2002). The role of proprioceptive feedback in the regulation and adaptation of locomotor activity. *Advances in Experimental and Medical Biology*, *508*, 343–355.

Lazar, A., Pipa, G., Triesch, J. (2009). SORN: a self-organizing recurrent neural network. *Frontiers in Computational Neuroscience*, *3*, 23.

Lemouzy, S., Camps, V., Glize, P. (2010). Real time learning of behaviour features for personalised interest assessment. In Y. Demazeau, F. Dignum, J. Corchado, J. Pérez (Eds.), *Advances in practical applications of agents and multiagent systems. Advances in soft computing* (Vol. 70, pp. 5–14). Berlin/Heidelberg: Springer.

Lemouzy, S., Camps, V., Glize, P. (2011). Principles and properties of a mas learning algorithm: a comparison with standard learning algorithms applied to implicit feedback assessment. In *Proceedings of the 2011 IEEE/WIC/ACM international conferences on web intelligence and intelligent agent technology - volume 02, WI-IAT '11* (pp. 228–235). Washington: IEEE Computer Society.

Mainen, Z.F., Joerges, J., Huguenard, J.R., Sejnowski, T.J. (1995). A model of spike initiation in neocortical pyramidal neurons. *Neuron*, *15*(6), 1427–39.

Makarov, V.A., Panetsos, F., de Feo, O. (2005). A method for determining neural connectivity and inferring the underlying network dynamics using extracellular spike recordings. *Journal of Neuroscience Methods*, *144*(2), 265–279.

Marsland, S., Shapiro, J., Nehmzow, U. (2002). A self-organising network that grows when required. *Neural Networks*, *15*(8–9), 1041–1058.

Martin, C., & Reggia, J. (2010). Self-assembly of neural networks viewed as swarm intelligence. *Swarm Intelligence*, *4*, 1–36.

Misiaszek, J. (2006). Neural control of walking balance: if falling then react else continue. *Exercise & Sport Sciences Reviews*, *34*(3), 128–134.

Misiaszek, J.E. (2003). The h-reflex as a tool in neurophysiology: Its limitations and uses in understanding nervous system function. *Muscle & Nerve*, *28*(2), 144–160.

Nicolis, G., & Prigogine, I. (1977). *Self-organization in nonequilibrium systems: from dissipative structures to order through fluctuations*. Hoboken: Wiley (Wiley-Interscience Publication).

North, M., Collier, N., Vos, J. (2006). Experiences creating three implementations of the repast agent modeling toolkit. *ACM Transactions Modelling Computing Simulation*, *16*(1), 1–25.

Ottens, K., Gleizes, M.-P., Glize, P. (2007). A multi-agent system for building dynamic ontologies. In *International joint conference on autonomous agents and multiagent systems (AAMAS), Hawaii* (pp. 1278–1284). ACM Press.

Perea, G., & Araque, A. (2010). Glia modulates synaptic transmission. *Brain Research Reviews*, *63*(1–2), 93–102.

Picard, G., & Gleizes, M.P. (2005). Cooperative self-organization to design robust and adaptive collectives. In *2nd international conference on informatics in control, automation and robotics (ICINCO'05), 14-17 september 2005, Barcelona, Spain, volume I* (pp. 236–241). INSTICC Press.

Powers, R.K., Robinson, F.R., Konodi, M.A., D., B.M. (1992). Effective synaptic current can be estimated from measurements of neuronal discharge. *Journal of Neurophysiology*, *68*(3), 964–968.

Roeschies, B., & Igel, C. (2010). Structure optimization of reservoir networks. *Logic Journal of the IGPL*, *18*(5), 635–669.

Rossi, P., Pierelli, F., Parisi, L., Perrotta, A., Bartolo, M., Amabile, G., Serrao, M. (2003). Effect of painful heterotopic stimulation on the cutaneous silent period in the upper limbs. *Clinical Neurophysiology*, *114*(1), 1–6.

Schwindt, P., & Crill, W. (1982). Factors influencing motoneuron rhytmic firing: results from a voltage-clamp study. *Journal of Neurophysiology*, *48*, 875–890.

Serugendo, G., Gleizes, M.-P., Karageorgos, A. (Eds.) (2011). Natural Computing Series. Springer.

Song, S., Miller, K.D., Abbott, L.F. (2000). Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nature Neuroscience*, *3*(9), 919–926.

Stanley, K.O. (2004). *Efficient evolution of neural networks through complexification*. Ph.D. thesis, Department of Computer Sciences, The University of Texas at Austin.

Stanley, K.O., & Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary Computation*, *10*(2), 99–127.

Stephens, J., Usherwood, T., Garnett, R. (1976). Technique for studying synaptic connections of single motoneurones in man. *Nature*, *263*, 343–344.

Takata, N., & Hirase, H. (2008). Cortical layer 1 and layer 2/3 astrocytes exhibit distinct calcium dynamics *in vivo*. *PLoS ONE*, *3*(6), e2525.

Türker, K.S., & Miles, T.S. (1991). Threshold depolarization measurements in resting human motoneurones. *Journal of Neuroscience Methods*, *39*(1), 103–107.

Türker, K.S., & Cheng, H.B. (1994). Motor-unit firing frequency can be used for the estimation of synaptic potentials in human motoneurones. *Journal of Neuroscience Methods*, *53*(2), 225–234.

Türker, K.S., & Powers, R.K. (1999). Effects of large excitatory and inhibitory inputs on motoneuron discharge rate and probability. *Journal of Neurophysiology*, *82*(2), 829–840.

Türker, K.S., & Powers, R.K. (2003). Estimation of postsynaptic potentials in rat hypoglossal motoneurones: insights for human work. *Journal of Physiology*, *551*, 419–431.

Türker, K.S., & Powers, R.K. (2005). Black box revisited: a technique for estimating postsynaptic potentials in neurons. *Trends in Neurosciences*, *28*(7), 379–386.

Türker, K.S., Yang, J., Brodin, P. (1997). Conditions for excitatory or inhibitory masseteric reflexes elicited by tooth pressure in man. *Archives of Oral Biology*, *42*(2), 121–128.

Turrigiano, G.G., Leslie, K.R., Desai, N.S., Rutherford, L.C., Nelson, S.B. (1998). Activity-dependent scaling of quantal amplitude in neocortical neurons. *Nature*, *391*(6670), 892–896.

Videau, S., Bernon, C., Glize, P., Uribelarrea, J.-L. (2011). Controlling bioprocesses using cooperative self-organizing agents. In *PAAMS* (pp. 141–150).

Villmann, T., Der, R., Herrmann, M., Martinetz, T. (1997). Topology preservation in self-organizing feature maps: exact definition and

measurement. *IEEE Transactions on Neural Networks*, 8(2), 256–266.

Warren, J.D., Miles, T.S., Türker, K.S. (1992). Properties of synaptic noise in tonically active human motoneurons. *Journal of Electromyography and Kinesiology*, 2(4), 189–202.

Welcomme, J.-B., Gleizes, M.-P., Redon, R. (2009). A self-organising multi-agent system managing complex system design application to conceptual aircraft design. *International Transactions on Systems Science and Applications, Self-organized Networked Systems*, 5(3), 208–221.

Whiteson, S., & Stone, P. (2006). Evolutionary function approximation for reinforcement learning. *Journal of Machine Learning Research*, 7, 877–917.

Wooldridge, M. (2002). *An introduction to multiagent systems*. Hoboken: Wiley.

Yao, X. (1999). Evolving artificial neural networks. *Proceedings of the IEEE*, 87(9), 1423–1447.