

Using Probabilistic Dependencies Improves the Search of Conductance-Based Compartmental Neuron Models

Roberto Santana, Concha Bielza, and Pedro Larrañaga

Abstract. Conductance-based compartmental neuron models are traditionally used to investigate the electrophysiological properties of neurons. These models require a number of parameters to be adjusted to biological experimental data and this question can be posed as an optimization problem. In this paper we investigate the behavior of different estimation of distribution algorithms (EDAs) for this problem. We focus on studying the influence that the interactions between the neuron model conductances have in the complexity of the optimization problem. We support evidence that the use of these interactions during the optimization process can improve the EDA behavior.

1 Introduction

The intrinsic electrophysiological properties of neurons condition the activity of neuronal circuits and ultimately determine biological responses to multiple stimuli. Conductance-based compartmental neuron models [6,7] have been very useful to study different aspects of neuronal dynamics. The electrical activity of this type of neuron model is mostly influenced by its ionic current conductances. Single compartmental models are particularly suitable for investigating the way in which different ionic currents act on the neuronal subthreshold behavior and spike generation.

A common method used in the creation of conductance-based compartmental neuron models is to record the *in vitro* response of the neuron to a set of simple current stimuli and then attempt to replicate the response on a detailed compartmental model of that cell [2]. Usually the general form of the model (determined by a set of differential equations) is known but it is necessary to find a choice of the model parameters that guarantees a good match between the model behavior and the experimental data. This identification process can be posed as the problem of finding the optimal (given a predefined measure) values for the set of parameters.

However, the parameter optimization problem is not straightforward. Disparate parameter combinations can lead to similar neuron electrophysiological properties, interactions between the parameters can be highly nonlinear and the simulation of the models can be very costly. Therefore, it is an important issue to conceive optimization algorithms than can deal with this type of problems. We use EDAs, [9,10], a class of evolutionary algorithms that construct probabilistic models of the set of selected solutions. Our analysis focuses on the influence that the interactions between the neuron model parameters have in the behavior of the different EDAs. We support evidence that the use of these interactions during the optimization process can help to obtain better sets of neuron parameters.

As a case study we use a database of about 1.7 million model neurons that were generated by independently varying the 8 maximal conductances of a realistic conductance-based model [13]. This information allows us to know the function landscape and optima, permitting us to accurately evaluate the performance of different variants of the optimization algorithms. It has already been shown [5,13,19] that databases of model neurons are particularly convenient to study the way in which the coordinated regulations (correlations) of ionic currents influence different aspects of the neurons dynamics. We speculate that correlations are translated into interactions between the variables of the optimization problem. Therefore, the databases can also be useful to investigate the way the search is affected by the strong interactions between the conductance parameters and to design more appropriate optimization methods for these problems.

2 Neurons and Models of Neurons

Neurons can display different types of spontaneous electrical activity. A neuron is *silent* when no electrical activity is displayed. On the contrary, *tonically active neurons* display different patterns of electrical activity and can be classified accordingly based on these patterns. *Spiking neurons* are those that display narrow spikes while *bursting neurons* have a broad shoulder after each spike. Neuron can also show an *irregular* activity, where none of the previous classifications can be given. The same classification is used to describe the behavior of neurons under electrical stimulation, i.e. when a current is input to the neurons. In general, different input current stimuli may determine changes in the electrical activity of the neuron.

To characterize the electrophysiological activity of neurons, measures that describe their electrical behavior have been defined. The *burst duration* is the interval in which there are a significantly higher number of spikes as compared to other intervals in the spike train. The *burst period* is the burst duration plus the largest interspike interval. The *number of maxima per period* is the number of maxima comprised in a bursting period. The *after-hyperpolarization (AHP) potential* is the trough voltage between bursts.

For tonically active and bursting neurons, the *spike amplitude* is the difference between the maximum value of the membrane potential and the value at the onset of the action potential. The average *inter-maximum interval (IMI)* is the

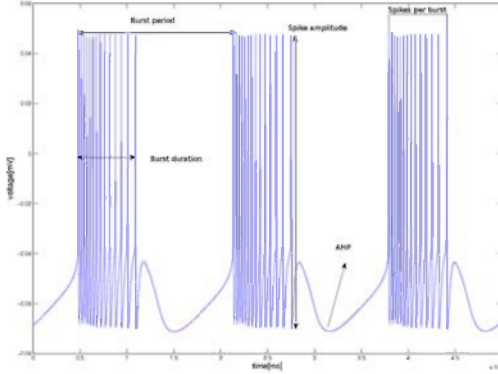


Fig. 1. Different measures that characterize the electrophysiological activity of neurons

average of the distances between two subsequent voltage maxima. The *discharge frequency* is the inverse of the IMI. Figure 1 illustrates some of these measures.

Neuron models are indispensable tools for understanding the neuron structural organization and testing different hypotheses about their behavior. A neuron model should display an appropriate balance between its accuracy to reproduce the neuron behavior and its computational complexity and tractability. We use a single compartmental model. This type of model neglects the neuron's spatial structure and focuses on how its various ionic currents contribute to subthreshold behavior and spike generation.

3 Model Description

The model was constructed based on experimental data obtained from lobster stomatogastric neurons [13,21]. The 8 currents in the single-compartmental model are based on those of the lobster stomatogastric ganglion neurons (STG) [21,14] and include a Na^+ current, I_{Na} ; two Ca^{2+} currents, I_{CaT} and I_{CaS} ; a transient K^+ current, I_A ; a Ca^{2+} dependent K^+ current, I_{KCa} ; a delayed rectifier K^+ current, I_{Kd} ; a hyperpolarization-activated inward current, I_H ; and a leak current, I_{leak} . In what follows we present the basic details about the model. More details can be found in [14].

Each of the model membrane currents is described by

$$I_i = \bar{g}_i(m_i)^p h_i(V - E_i)A \quad (1)$$

where E_i is the reversal potential, $A = 0.628 \times 10^{-3} \text{ cm}^2$ is the membrane area and \bar{g}_i is the maximal conductance. The database of models was constructed varying the maximal conductances of all 8 currents independently. Information

about the way in which the reversal potentials E_i were computed, as well the equations for the activation and inactivation variables m_i and h_i can be obtained from [14].

I_{input} , being a given input current, the neuron potential V is modeled by

$$C \frac{dV}{dt} = - \sum_i I_i - I_{input} \quad (2)$$

where $C = 0.628$ nF is the membrane capacity. In [14], some experiments were conducted to observe the model electrical activity with different input currents. For current step simulations, the input current was stepped from zero to a depolarizing DC current of 3 or 6 nA.

Different assignments of the model maximal conductance parameters will influence the model electrical activity. Therefore, it is possible to investigate which are the particular characteristics of a set of models determined by different parameter combinations. In [14], the database of 1, 679, 626 models was generated by varying the 8 parameters of the model previously described over 6 equidistant values.

Let X_i and x_i respectively represent a discrete random variable and a possible assignment to X_i . Similarly, we use $\mathbf{X} = (X_1, \dots, X_n)$ to represent an n -dimensional random variable and $\mathbf{x} = (x_1, \dots, x_n)$ to represent one of its possible values. In our problem representation, each variable will represent one of the neuron model parameters, $n = 8$ and $x_i \in \{0, \dots, 5\}$. Table 1 shows the variable codification of the neuron model parameters and the conductance values assigned to each of the membrane currents which were used in the construction of the database.

Each set of conductances represented by a vector $\mathbf{x} = (X_1, \dots, X_8)$ defines a “model neuron” or simply a *neuron*. The spontaneous activity of each neuron was simulated and classified into 4 categories: silent, tonically active, bursting, and non-periodic. Several features were extracted from the neurons. Different descriptors of the electrical activity were computed.

Table 1. Parameter representation and Conductance values assigned to each of the membrane currents and used in the construction of the database

<i>Var</i>	<i>Current</i>	0	1	2	3	4	5
X_1	I_{Na}	0	100	200	300	400	500
X_2	I_{CaT}	0	2.5	5.0	7.5	10.0	12.5
X_3	I_{CaS}	0	2.0	4.0	6.0	8.0	10.0
X_4	I_A	0	10.0	20.0	30.0	40.0	50.0
X_5	I_{KCa}	0	5.0	10.0	15.0	20.0	25.0
X_6	I_{Kd}	0	25.0	50.0	75.0	100.0	125.0
X_7	I_H	0	0.01	0.02	0.03	0.04	0.05
X_8	I_{leak}	0	0.01	0.02	0.03	0.04	0.05

4 Optimization Approach

Our goal is to find, from a large set of candidates, a neuron model that resembles the electrical activity of a given target neuron as described by recorded experimental data. One simplification is to use a search of candidate solutions amenable for tractable exhaustive enumeration (neuron database). Since the neuron model database provides a description of all the neurons, we can carefully design a fitness function based on this information and know a priori the function values for all the solutions of the search space.

For each optimization problem, we will use as a target neuron, one neuron model selected from the database according to some predefined criterion related to its electrophysiological activity. This choice of the target neuron guarantees that there is at least a solution of the search space that optimizes the fitness function. However, the question of how to measure the similarity between the dynamics of the target neuron and any other neuron has to be solved.

There are three main types of functions used to find optimal neuron model parameters [22]: feature-based functions, point-by-point comparison of voltage traces, and multi-objective functions. In this paper we use the first approach and the frequency of voltage maxima as the feature that will characterize the dynamics of the neuron models. The spontaneous frequency of voltage maxima contains information about the neurons dynamics but this information does not support much details about the neuron behavior. Therefore, in addition to the spontaneous frequency we use the steady state maximal frequencies as computed during 3 nA and 6 nA current injections.

Let \mathbf{x}^* and \mathbf{x} respectively be the target neuron and any other neuron of the search space. The fitness function $f(\mathbf{x})$ is defined as:

$$f(\mathbf{x}) = -((f_{q_s}(\mathbf{x}) - f_{q_s}(\mathbf{x}^*))^2 + (f_{q_3}(\mathbf{x}) - f_{q_3}(\mathbf{x}^*))^2 + (f_{q_6}(\mathbf{x}) - f_{q_6}(\mathbf{x}^*))^2)^{\frac{1}{2}} \quad (3)$$

where f_{q_s} , f_{q_3} and f_{q_6} respectively represent the spontaneous frequency and the steady state maximal frequencies as computed during during 3 nA and 6 nA current injections. When $\mathbf{x}^* = \mathbf{x}$, $f(\mathbf{x}) = 0$ which is the maximum of the function. Therefore, we transform the search for a neuron model with similar electrical activity to the target neuron in the maximization of function $f(\mathbf{x})$. Notice that the fitness landscape of this function will depend on the choice of \mathbf{x}^* .

5 Estimation of Distribution Algorithms

EDAs [9,10] are optimization algorithms that can learn and exploit the search space regularities in the form of probabilistic dependencies. They are very similar to genetic algorithms, but instead of using genetic operators, they construct an explicit probability model of a set of selected solutions. The model is used to generate new promising solutions. One characteristic that serves to distinguish different types of EDAs is the probabilistic model used by the algorithm. The

models may differ in the order and number of the probabilistic dependencies that they represent.

Let $p(\mathbf{x})$ denote a positive probability distribution. In this paper, we use three different types of models: A univariate marginal model, a tree model and a Bayesian network model. In a univariate model, the joint probability distribution can be factorized as the product of the univariate probabilities of the variables, i.e. $p(\mathbf{x}) = \prod_i p(x_i)$. This is the model used by the univariate marginal distribution algorithms (UMDA) [10].

A probability distribution $p_{Tree}(\mathbf{x})$ that conforms to a tree is defined as $p_{Tree}(\mathbf{x}) = \prod_{i=1}^n p(x_i | pa(x_i))$ where $Pa(X_i)$ is the parent of X_i in the tree, and $p(x_i | pa(x_i)) = p(x_i)$ when $Pa(X_i) = \emptyset$, i.e. X_i is the root of the tree. Probabilistic trees are represented by acyclic connected graphs. In this paper we use the Tree-EDA [18], an EDA that uses trees to represent the probability distributions. A Bayesian network can be seen as a generalization of a tree where each variable can have multiple parents. In this paper we use the estimation of Bayesian networks algorithm (EBNA) [3], one of the EDAs based on the use of Bayesian networks.

Pseudocode for EBNA is shown in Algorithm 1. The algorithm was implemented in Matlab using the MATEDA-2.0 software [15]. The implementations of the UMDA and Tree-EDA follow the same scheme but the learning and sampling steps are modified accordingly. These were implemented using the Matlab Bayes Net (BNT) toolbox [11]. The scoring metric used for the Bayesian network was the Bayesian metric with uniform priors, and each node was allowed to have a maximum number of 5 parents. The truncation parameter was $T = 0.5$. Best elitism, in which the selected population is passed to the next population, was used.

Algorithm 1. EBNA

```

1  Generate an initial population  $D_0$  of individuals and evaluate them
2   $t \leftarrow 1$ 
3  do {
4      $D_{t-1}^{S_e} \leftarrow$  Select  $N$  individuals from  $D_{t-1}$  using truncation selection
5     Using  $D_{t-1}^{S_e}$  as the data set, apply local search to find one BN structure
       that optimizes the scoring metric
6     Calculate the parameters of the BN using  $D_{t-1}^{S_e}$  as the data set
7      $D_t \leftarrow$  Sample  $M$  individuals from the BN and evaluate them
8  } until Stopping criterion is met

```

6 Related Work

There is considerable work on the application of optimization methods to neuron model parameter optimization. Usually, single objective functions that measure a particular aspect of the model performance given the parameters are used.

Among the most commonly employed optimization methods are [12,22]: hand tuning, gradient descent, evolutionary algorithms, bifurcation analysis and hybrid methods.

We review some of the work that seems to corroborate the convenience of modeling the interactions between the conductances in the search of neuron models that exhibit a desired electrophysiological behavior pattern. In the next section we present empirical results that show that this is indeed the case.

Achard and De Schutter [1] use an evolutionary strategy to obtain a set of different good models of the cerebellar *Purkinje* cells. The authors investigate different hypotheses that could explain the large diversity models with similar good conductance density values. Although probabilistic models of the neuron model parameter space are not constructed, the correlations between pairs of parameters are computed and used to investigate the relationship between the parameters.

In [4], Golowasch et al. generated a set of neuron models by randomly sampling the maximal conductance of the STG neuron [21]. The models were used to identify sets of maximal conductances that generate one-spike bursters. A model using the means of the maximal conductance of this set was constructed. It turned out that the model was not itself a one-spike burster. The authors concluded that averages over multiple samples can fail to characterize a system whose behavior depends on interactions involving a number of highly variable components.

In [5], a neuron database is used to investigate spiking variability in *Globus pallidus* neurons. The authors acknowledge that the effect of each conductance in the neuron electrophysiological properties was highly dependent on the background context of other present conductances. The fact that every conductance in the model could show opposite effects on spike rate when it was increased depending on the background of other present conductances [5] suggests that fitness functions that use the spike rate as a feature may have malign interactions between the variables [8]. Malign interactions can deceive evolutionary algorithms that do not take interactions into account.

In [19], Smolinski and Prinz analyze a different database of neuron models. From the analysis of the subset of models that resemble the electrophysiological behavior of natural neurons the authors identify three types of corregulations: 1) No significant interactions. 2) Expression of co-preference for specific ranges of values and 3) Corregulation, expressed by a characteristic “ridge” in the plot of pair-by-pair co-occurrence of specific parameter values. We could expect that variables with no significant interactions are less likely to appear together in the graphical model structures learned by EDAs.

7 Experiments

The main objective of our experiments is to investigate whether the use of interactions, represented by the graphical models used by EDAs, improves the results achieved when no interactions are taken into account. We assume that,

Table 2. Target neuron models selected from the database and a number of properties and measures determined from their simulation

<i>Index</i>	I_{Na}	I_{CaT}	I_{CaS}	I_A	I_{KCa}	I_{Kd}	I_H	I_{leak}	T_s	T_3	T_6	f_{q_s}	f_{q_3}	f_{q_6}
720973	100	7.5	4	40	5	125	0	0.01	2	1	1	1.1632	54.6364	60.1949
1522117	500	5.0	6	40	10	125	0	0.01	2	1	1	1.1559	37.7609	42.2354
833389	100	12.5	10	10	0	25	0.04	0.01	2	0	0	215.0538	0	0
965338	300	5.0	8	0	25	0	0.05	0.04	2	1	1	4.5382	5.8246	7.8225
436821	100	7.5	4	10	0	25	0.05	0.03	2	0	0	68.2173	0	0
1071411	300	10.0	10	40	20	25	0.02	0.03	2	2	2	3.9518	23.1826	27.9265
83317	0	2.5	8	40	5	100	0.02	0.01	2	0	0	2.4422	0	0
882103	300	0	10	20	15	100	0.05	0.01	2	1	1	18.0810	37.0142	42.6758
300566	100	0	4	30	25	75	0	0.02	2	1	1	4.8984	26.9808	35.5637
1374808	400	12.5	4	40	20	125	0	0.04	2	2	2	8.0523	35.9712	47.1328

if EDAs that represent probabilistic dependencies, i.e. Tree-EDA and EBNA, outperform those that do not represent such dependencies, i.e. UMDA, then the problem exhibits interactions between the variables and these interactions are important to solve the problems.

To test the algorithms, we select 10 functions defined by selecting different target neuron models from the data sets. The target neurons, shown in Table 2, are all bursting neurons during the spontaneous activity and have been selected trying to cover different patterns of electrical activity of the neurons. Under spontaneous activity, the first five neurons have the following relevant characteristics: for 720973: a high burst period and small burst duration; for 1522117: a high burst period and high burst duration; for 833389: the smallest burst period; for 965338: one of the smallest burst durations; for 436821: one of the highest number of maxima. The rest of the target neurons have been randomly selected from the set of bursting neurons.

In the table, *Index* is the index of the neuron in the database, T_s , T_3 and T_6 are the types of electrical activity under the different experimental conditions, coded as silent neuron (0), spiking neuron (1), bursting neuron (2). Similarly f_{q_s} , f_{q_3} and f_{q_6} represent the frequencies under the different experimental conditions.

We conducted 30 experiments of UMDA, Tree-EDA and EBNA for each of the 10 functions and with two different settings: I) Population size is 500 and 30 generations; II) Population size is 1000 and 60 generations are conducted. These settings were determined aiming to keep the number of function evaluations relatively small, and after preliminary experiments were conducted. The number of times each algorithm found the optimum for all the functions are shown in Table 3. It can be seen in the table that only one of the problems is relatively easier to solve by the methods. Instance 1071411 is particularly complex for all the methods. However, the best results are clearly achieved by EBNA that outperforms both the Tree-EDA and UMDA.

To determine whether differences between the algorithms are statistically significant, we have used the Kruskal-Wallis test to accept or reject the null hypothesis that the samples have been generated from the same probability distribution. The test significance level was 0.01. For all instances, except instances 833389 and 436821, significant statistical differences have been found between

Table 3. Number of times each algorithm found the optimum for each of the 10 functions

<i>Alg.</i>	720973		1522117		833389		965338		436821		1071411		83317		882103		300566		1374808		<i>Tot.</i>	
<i>Setting</i>	<i>I</i>	<i>II</i>	<i>I</i>	<i>II</i>	<i>I</i>	<i>II</i>	<i>I</i>	<i>II</i>	<i>I</i>	<i>II</i>	<i>I</i>	<i>II</i>	<i>I</i>	<i>II</i>	<i>I</i>	<i>II</i>	<i>I</i>	<i>II</i>	<i>I</i>	<i>II</i>		
<i>UMDA</i>	0	0	0	0	6	27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	33
<i>TREE – EDA</i>	0	0	0	0	13	19	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	36
<i>EBNA</i>	0	2	1	0	30	30	2	4	6	12	0	0	0	1	2	1	3	8	18	25	129	

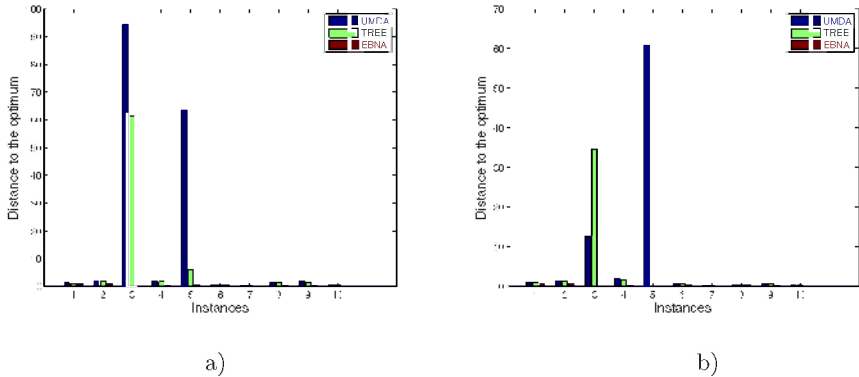


Fig. 2. Distance of the solutions found by the different EDAs to the best solutions for: a) EDA setting (500 – 30). b) EDA setting (1000, 60).

the EBNA and the other two EDAs for the two settings. There were not significant statistical differences between EBNA and Tree-EDA for instance 436821 for both EDA settings, and between EBNA and the other two EDAs for instance 833389, setting II.

It is interesting to note that the use of bivariate dependencies are not sufficient to improve the EDA results. To investigate the quality of the average solutions found by the algorithms, we also compute the average fitness distance of the best solutions found by the different EDAs to the optimal solutions. This information is shown in Figure 2. It can be appreciated that the increment in the number of allowed function evaluations, due to more individuals in the population and more generations, does not significantly improve the results of UMDA and Tree-EDA.

Finally, we investigate the structure of the dependencies learned by EBNA. For each problem, we computed the frequency of the arcs learned by the Bayesian network. The arc frequencies were calculated from the set of 900 Bayesian networks learned using the first EDA experimental setting. Figure 3 (a) shows the frequency matrix learned for instance 1071411. Lighter colors represent stronger dependencies between the variables. Note that not all the dependencies appear with the same frequency.

For each problem, frequency matrices were thresholded to leave those arcs that were in at least 550 of the 900 Bayesian networks. We then computed the thresholded dependencies that were present in at least 5 of the 10 instances. These dependencies are shown in Figure 3 (b). We hypothesize that some of them

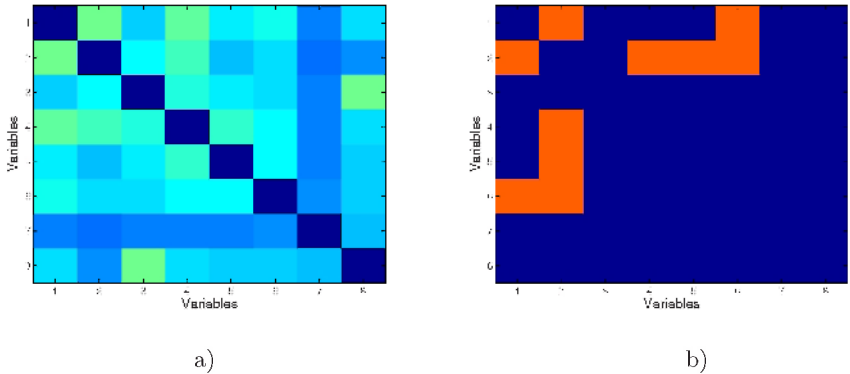


Fig. 3. Structure of the interactions captured by the Bayesian networks learned by EDAs. Lighter colors represent stronger dependencies between the variables. a) Most frequent structures learned for instance 1071411. b) Dependencies that were frequent for at least five of the 10 functions.

are due to the existence of important coregulations between the conductances of bursting neurons. To validate this hypothesis, we investigate previous studies of the conductance structure [20].

In [20], the construction of a dimensional stack image that visualizes the relationship between the spontaneous neuron activity and the conductance values allowed to make some conclusions about the structure of the conductance space. Of the four statements made about the “layout” of neuron models in the conductance space, only two refer to the interactions between conductances: 1) Many one-maxima bursters that have nonzero \bar{g}_{Na} have zero delayed-rectifier potassium conductance \bar{g}_{Kd} . 2) There seems to be a regular gradation of bursters from few maxima per burst to many maxima per burst as one increases g_{Kd} and decreases g_{CaT} . The interactions between the conductances pairs (g_{Na}, g_{Kd}) and (g_{Kd}, g_{CaT}) are both captured in the matrix shown in Figure 3 b). Further analysis, for instance the inspection of the corresponding marginal tables, should reveal the type of relationship between these pairs of parameter conductances as captured by the Bayesian networks. This remains as a subject of further work.

8 Conclusions

In this paper we have shown that interaction should be taken into account in the search of neuron models that have a predefined physiological activity. This is the first time, to our knowledge, that EDAs have been applied in the context of neuron modeling[1]. In contrast with previous EDA applications to problems from the biological domain [17], where bivariate interactions are sufficient to

¹ This is not the case in the field of artificial neural networks where several applications of EDAs have been reported.

remarkably improve the results achieved by univariate models, for the problems addressed in this paper, higher order interactions are needed.

Elucidating the relationship between the distribution of their intrinsic properties and dynamic activity of neurons is a crucial step in understanding larger-scale phenomena such as network oscillations and inter-nuclei synchronization. We speculate that further application of intelligent data analysis techniques [15,16] to the data generated by the EDAs can unveil additional information about the structure of the conductance space.

Acknowledgements

We thank to the Prinz laboratory, particularly to Amber Hudson and Tomasz Smolinski for providing useful information on neuron model databases, including the database used in this paper. This research is part of the CajalBlueBrain project. It has been partially supported by TIN-2008-06815-C02-02, TIN2007-62626 and Consolider Ingenio 2010 - CSD2007-00018 projects (Spanish Ministry of Science and Innovation).

References

1. Achard, P., De Schutter, E.: Complex parameter landscape for a complex neuron model. *PLoS Computational Biology* 2(7), 794–804 (2006)
2. Druckmann, S., Banitt, Y., Gidon, A., Schuermann, F., Markram, H., Segev, I.: A novel multiple objective optimization framework for constraining conductance-based neuron models by experimental data. *Frontiers in Neuroinformatics* 1(1) (2007)
3. Etxeberria, R., Larrañaga, P.: Global optimization using Bayesian networks. In: Ochoa, A., Soto, M.R., Santana, R. (eds.) *Proceedings of the Second Symposium on Artificial Intelligence (CIMAF 1999)*, Havana, Cuba, pp. 151–173 (1999)
4. Golowasch, J., Goldman, M.S., Abbott, L.F., Marder, E.: Failure of averaging in the construction of a conductance-based neuron model. *Journal of Neurophysiology* 87, 1129–1131 (2002)
5. Günay, C., Edgerton, J.R., Jaeger, D.: Channel density distributions explain spiking variability in the globus pallidus: A combined physiology and computer simulation database approach. *Journal of Neuroscience* 28(30), 7476–7491 (2008)
6. Herz, A.V.M., Gollisch, T., Machens, C.K., Jaeger, D.: Modeling single-neuron dynamics and computations: A balance of detail and abstraction. *Science* 314(5796), 80–85 (2006)
7. Hodgkin, A.L., Huxley, A.F.: A quantitative description of ion currents and its applications to conduction and excitation in nerve membranes. *Journal of Physiology* 117, 500–544 (1952)
8. Kallel, L., Naudts, B., Reeves, R.: Properties of fitness functions and search landscapes. In: Kallel, L., Naudts, B., Rogers, A. (eds.) *Theoretical Aspects of Evolutionary Computing*, pp. 177–208. Springer, Heidelberg (2000)
9. Larrañaga, P., Lozano, J.A. (eds.): *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*. Kluwer Academic Publishers, Dordrecht (2002)

