Biol Cybern (2008) 99:371–379
DOI 10.1007/s00422-008-0269-2

ORIGINAL PAPER

Biological Cybernetics

Evaluating automated parameter constraining procedures of neuron models by experimental and surrogate data

Shaul Druckmann · Thomas K. Berger · Sean Hill · Felix Schürmann · Henry Markram · Idan Segev

Received: 17 May 2008 / Accepted: 23 September 2008

Abstract Neuron models, in particular conductance-based compartmental models, often have numerous parameters that cannot be directly determined experimentally and must be constrained by an optimization procedure. A common practice in evaluating the utility of such procedures is using a previously developed model to generate surrogate data (e.g., traces of spikes following step current pulses) and then challenging the algorithm to recover the original parameters (e.g., the value of maximal ion channel conductances) that were used to generate the data. In this fashion, the success or failure of the model fitting procedure to find the original parameters can be easily determined. Here we show that some model fitting procedures that provide an excellent fit in the case of such model-to-model comparisons provide ill-balanced results when applied to experimental data. The main reason is that surrogate and experimental data test different aspects of the algorithm's function. When considering model-generated surrogate data, the algorithm is required to locate a perfect solution that is known to exist. In contrast, when considering experimental target data, there is no guarantee that a perfect solution is part of the search space. In this case, the optimization procedure must rank all imperfect approximations and ultimately select the best approximation. This aspect is not tested at all when considering surrogate data since at least one perfect solution is known to exist (the original parameters) making all approximations unnecessary. Furthermore, we demonstrate that distance functions based on extracting a set of features from the target data (such as time-to-first-spike, spike width, spike frequency, etc.)—rather than using the original data (e.g., the whole spike trace) as the target for fitting—are capable of finding imperfect solutions that are good approximations of the experimental data.

Keywords Neuron · Model · Compartmental · Multi-objective Optimization · Firing pattern · Automated fitting · Parameter constraining

S. Druckmann (✉) · I. Segev
Interdisciplinary Center for Neural Computation
and the Department of Neurobiology, Institute of Life Sciences,
The Hebrew University of Jerusalem, Edmond J. Safra Campus,
Givat Ram, 91904 Jerusalem, Israel
e-mail: drucks@lobster.ls.huji.ac.il

I. Segev
e-mail: idan@lobster.ls.huji.ac.il

T. K. Berger · S. Hill · F. Schürmann · H. Markram
Brain Mind Institute, Ecole Polytechnique Fédérale de Lausanne
(EPFL), 1015 Lausanne, Switzerland
e-mail: thomas.berger@epfl.ch

S. Hill
e-mail: sean.hill@epfl.ch

F. Schürmann
e-mail: felix.schuermann@epfl.ch

H. Markram
e-mail: henry.markram@epfl.ch

1 Introduction

Model fitting procedures designed to constrain the parameters of a neuron model in accordance with a set of target data, typically the spike firing of the neuron, are complex, multi-faceted algorithms. Several different such procedures have been suggested and tested in different contexts in recent years (Achard and De Schutter 2006; Bush et al. 2005; Druckmann et al. 2007; Gerken et al. 2006; Keren et al. 2005; Prinz et al. 2003; Vanier and Bower 1999; Weaver and Wearne 2006). Each of these procedures is composed of distinct, closely interacting elements. One such element is the neuron model itself whose parameters are to be constrained. Other elements include the distance functions used

to compare between model results and target data, and the optimization algorithm that, given a measure of the success of a set of solutions, suggests further solutions to be evaluated. Each of these elements is a subject of intensive research. Dozens of different optimization algorithms exist, many different distance functions have been suggested, and the range of different neuron models that can be explored is of course limitless. Hence, the number of possible combinations of the aforementioned elements is staggering. Thus, a reliable test of the effectiveness of different model fitting procedures is of great interest.

A direct way to evaluate the effectiveness of a model fitting procedure is to test the method on the task it is ultimately to face. Namely, constraining the parameters of a given model in accordance with a defined set of experimental data. However, one possible problem with such an approach is that failure of the method to arrive at an acceptable model might have more than a single cause. First and foremost, it could be due to shortcomings in the model fitting procedure. However, it could also be due to an unfortunate choice of the model itself and the set of functions associated with it (e.g., the specific set of ion channel kinetics in the model). The case may be that there is no set of parameters (e.g., of the value of the maximal channel conductance) that is capable of generating an output that matches the target data. Clearly, in this case the model fitting procedure is not responsible for the failure to generate an acceptable model.

In order to circumvent this problem, one may alternatively challenge the fitting method to constrain the parameters of the model according to data for which we know a priori that an acceptable result is available. One way to do so is to use surrogate data. Indeed, such an approach has been used by several previous studies (Achard and De Schutter 2006; Keren et al. 2005; Weaver and Wearne 2006). Specifically, an existing compartmental model with a pre-selected set of ion channels kinetics and a given set of parameters (e.g., the value of their maximal conductance) is used to generate surrogate target data (e.g., in response to a step current pulse of a given amplitude). The specific parameter values that generated the data are not disclosed to the model fitting algorithm and it must search the parameter space anew in order to find solutions that match the target data. In this case, an acceptable solution (a perfect solution in fact) is guaranteed, namely the parameter values that were originally used to generate the surrogate data. We term this approach the "reference model test" as a previously developed, reference, model with a reference set of parameters is used to generate the target data.

The reference model test challenges the capability to find a perfect solution if one exists. While this is an aspect that should indeed be evaluated, we show in this study that it is only a necessary, not sufficient, criterion for evaluating the success of a parameter constraining method. Specifically, we show that the exact same method is deemed a success in the reference model test but then fails to obtain well-balanced acceptable solutions in the more realistic case of constraining the model in accordance with experimental data. In order to ascertain that this is not due to a failure of the chosen model, we modify the method only by changing the distance function used to compare simulation and target data. We show that with the new distance function, a multiple objective feature-based distance function (Druckmann et al. 2007), the model fitting procedure now succeeds in obtaining successful solutions. We then discuss the reason for the failure of the forgetful model to predict the applicability of a distance function (or model fitting procedure) to the real task of constraining neuronal models in accordance with experimental data.

2 Methods

2.1 Cell model

Full details of the cell model can be found in Druckmann et al. (2007). In brief, the cell was represented as a compartmental, conductance-based model using a detailed reconstructed morphology of a nest basket cell from layer 2/3 of the rat primary somatosensory neocortex. Ten ion channels were incorporated in the model cell soma (for details see the article mentioned above). For the sake of simplicity the dendrite was kept passive. The maximal conductances for the different channels as well as the leak conductance in the soma and dendrite were chosen as free model parameters, for a total of 12 free parameters. All simulations were performed in NEURON (http://www.neuron.yale.edu, Carnevale and Hines 2005).

The simulation of the application of the stimulus protocol to a detailed neuron model (i.e., the evaluation of a single parameter set) requires a quite substantial time (typically minutes). As we would like to explore many thousands of parameter sets within a reasonable amount of time, we implemented the algorithm in a fashion that is compatible with use on parallel clusters running message passing interface (MPI). The optimization procedure ran on two types of parallel machines: a cluster consisting of 28 Sun $\times$4100, dual AMD 64 bit Opteron 280 dual core (total of 112 processors), running Linux 2.6 and the EPFL BlueGene/L supercomputer (Adiga et al. 2002). Average run time of a single model fitting procedure on the cluster was approximately one day. Roughly equivalent runtime was observed when running on 256 processors of the EPFL BlueGene. By parallelizing to the level of individual stimulus steps, nearly linear speedup was also achieved for 512 processors.

2.2 Distance functions

In this study, we employ two different classes of distance functions. First, we consider trace-to-trace comparisons,

distance functions that aim to have a given model response perfectly match a given experimental trace; second, feature-based comparisons, distance functions that aim to bring a model to accord with a set of values extracted from experimental data. We chose to use the phase plane trajectory distance function to represent the former class as it is an advanced form of such comparison designed to overcome the sensitivity to phase shifts that is typical of trace-to-trace comparisons (LeMasson and Maex 2001). We chose to employ the multiple objective optimization feature-based distance functions (Druckmann et al. 2007) to represent the latter class as it has been shown to successfully deal with experimental data.

Here, we present a brief overview of these distance functions. A detailed description of the distance functions can be found in the above references. For a given voltage trace, the phase plane trajectory method is based upon the construction of a two-dimensional (phase-plane) histogram pitting the voltage (X-axis) versus its time derivate (Y-axis). For each point in a trace, a count is added to the bin in the 2D histogram that corresponds to the voltage and voltage-derivative value of the point. The distance between two traces is calculated via the difference in bin counts of the two-dimensional histograms. As we compare between three levels of step currents simultaneously, there are two simple options. First, the three values of the individual comparisons can be summed to end up with a single distance value between model and target data. Second, each individual comparison distance value can be considered separately resulting in three distance values.

Feature-based distance functions were designed to deal with the variability of experimental responses to identical stimuli (e.g., sequential repetitions of the same current step). Averaging the raw traces is usually not a viable option as they typically contain action potentials and thus do not lend themselves to averaging. Traditional distance functions use single trace-to-trace comparisons. Yet, as the various voltage responses to the same stimulus differ amongst themselves, it is somewhat arbitrary to select a given voltage trace and force the model to be in full accordance with it and not the other equally valid voltage traces. However, features of the voltage trace (such as firing rate, time to first action potential, action potential width, etc.) can be readily extracted along with their variability from the target experimental data. Thus, we chose to use these features as the basis of our distance functions. Hence the term feature-based distance functions. Specifically, we measure the distance between a given model trace and the experimental data by extracting the feature values and calculating their distance from the experimental mean in units of experimental variability. The different features are combined using multiple objective optimization techniques (see the following section). As the surrogate data contains no variability, we use a surrogate variability of one percent of the mean for each feature.

For an example of the application of phase plane trajectory distance functions to constraining the parameters of neuron models see Achard and De Schutter (2006). For an example of the application of feature-based distance functions see Druckmann et al. (2007).

2.3 Multiple objective optimization

It seems to us extremely useful (if not essential) to be able to employ more than one distance function in the model fitting procedure, as we often want to employ a given distance function for more than a single stimulus and/or use multiple distance functions for each stimulus. In standard single objective optimization, the resulting values for different distance functions must be summed in order to arrive at a single scalar distance value. Unless, by some chance, the ensuing multiple distance values are scaled relative to each other for every possible type of comparison, one will end up with unequal contributions of different distance functions in different contexts (for a more detailed exposition of the problem see Druckmann et al. 2007). In order to address this issue, we opt to use the technique of multiple objective optimization (Cohon 1985; Hwang and Masud 1979). In brief, the main difference between single objective and multiple objective optimizations lies in the fact that the greater than, lesser than relationship between two scalar distance function values in single objective optimization is replaced by the relationship of *dominance*. If there are M distance functions $f_j(x), j = 1, \ldots, M$ (referred to as objective functions in the multiple objective optimization literature) then a solution x_1 is said to *dominate* a solution x_2 if both the following conditions hold:

$$\forall j : f_j\left(x^1\right) \leq f_j\left(x^2\right) \tag{1}$$

$$\exists k \in \{1, 2\ , \ldots, M\} \quad \text{so that} \quad f_k\left(x^1\right) < f_k\left(x^2\right) \tag{2}$$

Thus, solution A may dominate solution B, solution B may dominate solution A, or importantly neither solution dominates the other.

2.4 Optimization algorithms

We employ a class of global optimization algorithms termed genetic algorithms (Holland 1975) due to their original inspiration from abstracted notions of genetic evolution. These algorithms are a thriving field of research and have been put to extensive use in many different fields. Indeed, one can find a staggering number of algorithm variants and different heuristics (see for example Bäck et al. 1998) making it almost impossible to perform a comprehensive comparison of all variants on a specific task. In our exploration of the different algorithms, we found one specific variant named NSGA-II (Deb et al. 2002) to be highly effective.

Each iteration of the optimization procedure tests 300 parameter sets in parallel. By trial and error, we found that the algorithm typically converges after approximately 1,000 iterations (testing 3×10^5 parameter sets). In order to better assure convergence we perform 1,500 iterations. As the entire optimization procedure is stochastic, we repeat the full procedure 50 times and combine all solutions. The best solutions are then chosen from this pool.

The feature-based distance functions are used within a multiple objective optimization framework. We initially implemented the phase plane trajectory distance functions both in a single objective version (all three current steps equally summed) and in a multiple objective version (each current step as an individual objective). Though we found that the multiple objective version converged slightly faster, the large safety factor of 1,500 iterations caused this difference to be merely quantitative, not qualitative. We thus chose to keep the phase plane distance functions as originally employed in their relevant studies and used the single objective optimization version.

2.5 Model selection

Feature-based distance functions are expressed in readily interpreted units of experimental standard deviations from the target data mean. Thus, a threshold for the acceptance of models can be easily set to, for example, two standard deviations in each feature. In contrast, the phase plane trajectory distance functions are expressed in arbitrary units. Thus, choosing a reasonable threshold is rather difficult. Moreover, the specific value of this threshold is difficult to interpret as high or low without detailed knowledge of the specific modeling study (e.g., average error values for best solutions, sample traces with a given error, etc.).

For the phase plane trajectory distance functions we chose a value of 0.01 in the case of model-to-model comparisons and 0.1 in the case of model-to-experimental data comparisons. For comparison, the distance value between the middle intensity experimental trace and the experimental responses to the weaker and stronger step stimuli are 0.31 and 0.39, respectively. As these values are in arbitrary units that depend on the nature of the target data it is difficult to compare these values to those of other studies. However, we note that the values correspond approximately to the top one percent of all solutions. For the feature-based distance functions we chose a threshold of two standard deviations in the case of experimental and surrogate data.

2.6 Model-to-model comparison (the reference model test)

A common practice used to assure that a successful solution for the target data can be found in the search space is using controlled surrogate data. Specifically, several studies have generated surrogate data by utilizing a previously found model in the fashion detailed below. First, a given reference model (typically obtained by a previous manual-tuning of model parameters) is used to generate the surrogate data. Next, the specific reference parameter values (e.g., membrane capacitance, channel conductances) of the model are discarded and the model fitting procedure must search for these values anew within a given allowed range. With this method a perfect solution is known to exist, namely, the parameter values that generated the data in the first place.

2.7 Electrophysiological recordings

Acute brain slices were prepared from the primary somatosensory neocortex of 14- to 18-day old Wistar rats. Cells were targeted under infrared differential intereference contrast microscopy and recorded in the current clamp mode using the patch clamp technique. For full experimental procedure see Markram et al. (1997). Voltage responses current injection were recorded from the cells. In this study we present data from one cell that has been stimulated with 2 s depolarizing current steps of three different amplitudes (150, 200, and 250 pA).

3 Results

We begin by implementing the reference model test. In brief, we use a previously obtained model to generate surrogate data by applying a simulated current clamp injection and recording the model voltage response (see Sect. 2). The model fitting procedure is then applied in order to constrain the model in accordance with the surrogate data (see Sect. 2).

Figure 1 portrays the relevant voltage responses to the mid level intensity step current. The surrogate data voltage trace is plotted in (Fig. 1a). We apply the model fitting procedure with a phase plane trajectory distance function in order to constrain model parameters in accordance with the target surrogate data. The response of a typical example of models found at the end of the parameter constraining procedure to pass the error threshold (see Sect. 2) is shown in (Fig. 1b). We then repeat the same parameter constraining procedure with the exact same elements save that the distance function is changed from the phase plane trajectory distance function to a feature-based distance function. The response of a typical representative of the models that passed the feature-based distance function error threshold is shown in (Fig. 1c). As can be seen, the models generated by using both the phase plane trajectory distance function and the feature-based distance function (Fig. 1b and c, respectively) both closely resemble the target surrogate data (Fig. 1a).

We now employ the parameter constraining procedure to generate models in accordance with experimental, not

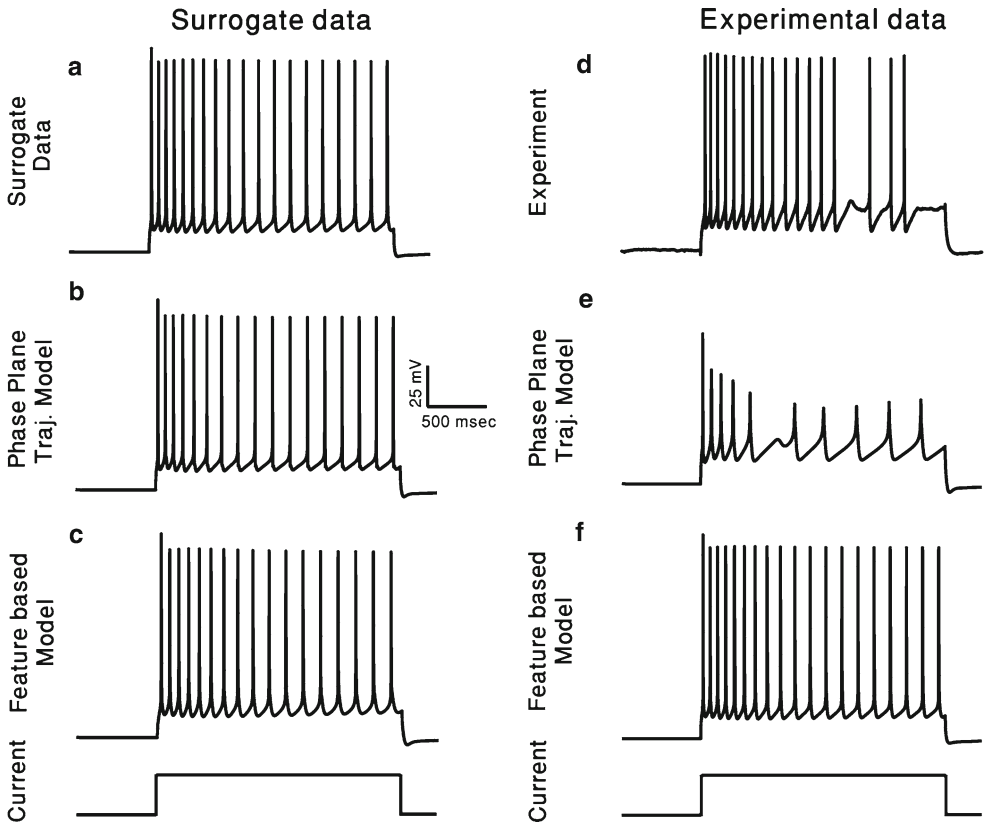

Fig. 1 Fitting model to model-generated surrogate data and to experimental data. **a–c** Fitting model to model-generated data. **a** Surrogate data. Shown is the voltage response of the compartmental neuron model to a two second long, 200 pAmp step current (*bottom* most trace displays current). **b** Voltage response of the same current step applied to one of the low error solutions of the phase plane trajectory method (for precise error values see Sect. 3). **c** The response of the feature based distance functions method for one of its low error solutions. Note that both methods, represented by traces b and c, generate models that are in close accord with the target data trace a. **d–f** Fitting model to experimental data. The exact same procedures applied to generate the models displayed in **b** and **c** are repeated only now with experimental data as the target for fitting. **d** Experimental voltage response. **e** Phase plane trajectory model. **f** Feature based distance function. Note that the phase plane trajectory solution is quite dissimilar to the target data while the feature based distance function solution is in closer accord with the target data

surrogate data. We apply the exact same parameter constraining procedure (including all parameters such as parameter search ranges, iteration number, etc.) using the same model and ion channel kinetics. Figure 1d–f displays the relevant voltage responses to the mid level intensity step current. The experimental data voltage trace is shown in (Fig. 1d). The voltage response of the model obtained by the phase plane trajectory distance function with the lowest distance between target data and model over all fifty repetitions of the parameter constraining procedure (see Sect. 2) is shown in (Fig. 1e). The voltage response of the model obtained by the feature-based distance function with the lowest distance, assuming equal weighting of features, is shown in (Fig. 1f). Note that neither voltage trace perfectly replicates the target experimental data. The afterhyperpolarization is better matched by that of the phase plane trajectory model (Fig. 1e). Yet, the model obtained by the feature-based distance function (Fig. 1f) is in much closer general accord with the target data. Both examples are typical representations of low error models of the respective distance functions. These results are consistent across nearly all of the 50 repetitions of the parameter constraining procedure.

The phase plane trajectory distance value between the traces shown in Fig. 1e and Fig. 1d (in arbitrary units) is 0.09. For comparison, the distance value between the experimental trace (Fig. 1d) and the experimental responses to the weaker and stronger step stimuli are 0.31 and 0.39, respectively. One can also measure the distance between the model obtained by the feature-based distance functions and the experimental trace (Fig. 1f to Fig. 1d) as calculated by the phase plane trajectory distance function. The resulting value is 0.7. Thus, although the result found by the feature-based distance function is in better accord with the experimental data it is considered poorer by the phase plane trajectory distance function

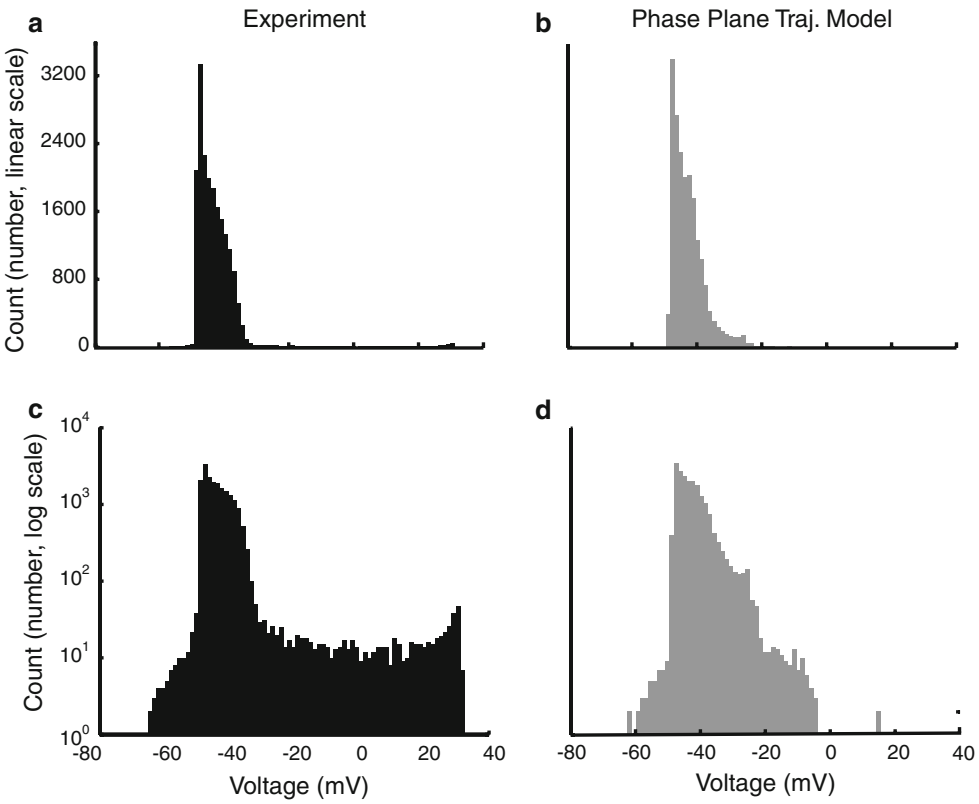

Fig. 2 Voltage histograms comparing experimental data to the response of the phase plane trajectory model. **a** The histogram of voltage values of the response shown in Fig. 1d. The *Y*-axis denotes number of voltage points (of 20,000) in a linear scale. **b** Same as in a for the voltage response of the best phase plane trajectory model. Note that they appear quite similar. **c**, **d** Histograms shown in (**a**, **b**) but with a log scaled *Y*-axis. Note that the number of counts of voltage values corresponding to action potentials (e.g., >−30 mV) is dominated by the number of counts in the sub threshold voltage regime and that in the supra-threshold regime the experimental and phase plane trajectory model differ to a great degree

Hence, if encountered during the parameter search it would be rejected in favor of (Fig. 1e). In summary, Figure 1 shows that a given method can do extremely well on the reference model test yet fail to obtain a well-balanced model when attempting to fit experimental data.

Figure 2a shows the voltage value histogram of the experimental trace depicted in (Fig. 1d). As can be seen, the histogram is dominated by the values between approximately −50 and −30 mV, corresponding to sub-threshold voltages. The action potentials themselves, however, can be noticed as a minute peak in the right hand tail of the histogram. Figure 2b shows the same histogram on a log scale. In this depiction, the voltage values corresponding to action potentials can be clearly seen. As shown by the log scale the voltage points corresponding to action potentials constitute a fraction nearly two orders of magnitude smaller than those corresponding to sub threshold values. Thus, any trace-to-trace comparison that gives equal precedence to all voltage points will end up implicitly favoring the subthreshold regime. Figure 2c displays the linear scale voltage value histogram of the trace shown in (Fig. 1e) that represents a low error solution of the phase plane trajectory distance function. Figure 2d depicts the log scale histogram of the same trace. As can be seen, the linear scale histogram seems to be quite similar to that of the experimental trace yet the differences are obvious in the log scale histogram, mainly in the range of voltage pertaining to action potentials.

Figure 3 depicts the matching phase plane trajectories (see Methods). Each point in the phase plane is represented by a grey scale color code based on the number of corresponding data points. Figure 3a portrays the phase plane trajectory of the experimental trace depicted in (Fig. 1d) in a strongly scaled fashion. Note that elliptic trajectories corresponding to action potentials can be clearly seen but many points have saturated black values. Figure 3b depicts the same form of plot for the best model found with the phase plane trajectory distance function whose voltage trace is plotted in (Fig. 1e). Note that with this scaling the trajectories are markedly dissimilar especially in the form of the action potential trajectory. Figure 3c depicts the phase plane trajectory in a more naturally scaled color code. Namely, white still corresponds to a count of zero whereas black now corresponds to the maximum histogram value (∼0.05). The inset portrays an expanded view of the region of the phase plane with the higher count values. Figure 3d depicts the model phase plane trajectory with the same color code. Note that the trajectories corresponding to action potentials can no longer be readily differentiated from the zero value. Employing this scale, those bins of the experimental trace that can clearly be distinguished from the background (also see inset) appear quite similar to those of the model trace. The dissimilarity of the scaled phase plane trajectories and similarity of the unscaled versions is akin to the similarity of the linear and dissimilarity of the log scale one dimensional histograms in (Fig. 2).

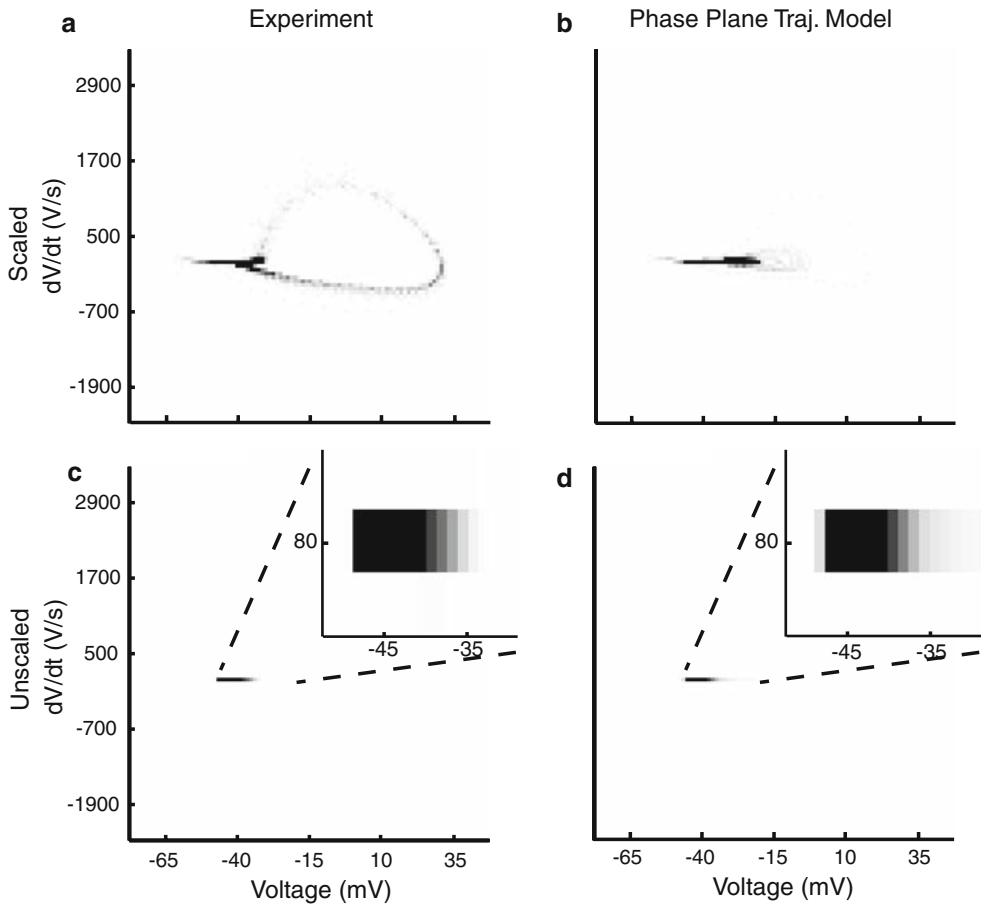

Fig. 3 Phase plane trajectory plots for the experimental data and for the response of the phase plane trajectory model. **a** Scaled phase plane trajectory plot for experimental response shown in Fig. 1d. Each bin in the 100×100 two dimensional histogram is coded in grey scale according to the number of voltage points in the bin; white color for no points and black for any value exceeding 0.1% of the total voltage points. Intermediate values are linearly interpolated. The elliptic trajectories correspond to action potentials. **b** Same form of plot as in a, but for the best phase plane trajectory model. Note that at this scale, a and b are quite dissimilar. **c** Unscaled phase plane trajectory plot for experimental response shown in Fig. 1d. Here the grey scale code corresponds to no points (*white*) whereas black corresponds to the maximum histogram value (∼0.05). Inset represents a zoom into the region with the highest bin counts. **d** Same as in **c**, but for the best phase plane trajectory model. Note that in this resolution the phase plane trajectories seem quite similar

4 Discussion

4.1 Surrogate data versus experimental data

The main difference between the reference model test and the challenge of constraining a model with respect to experimental data is that in the former case a perfect solution is guaranteed to exist (namely, the parameter set which generated the surrogate data). In stark contrast, when considering data obtained from experiment, in order for a perfect solution to be part of the search space each of the possibly dozens of ion channels must be present in the neuron model, with their dynamics perfectly described and their distributions across the membrane faultlessly characterized. Even if such exquisite experimental precision becomes available some day, many modeling efforts would still choose to use reduced models for reasons of tractability. This would again dispel the possibility of a perfect solution existing in the searchable parameter space.

Clearly, if such a perfect solution does exist, it is guaranteed to generate traces that are in exact point-to-point agreement with the target data set. Thus, any distance function, no matter what form of comparison it embodies, will recognize it as an ideal solution. Even basic distance functions such as mean square error, which clearly suffer from grave problems when used for voltage traces that contain action potentials as discussed by (LeMasson and Maex 2001), will recognize it as a perfect solution.

In contrast, if no perfect solution exists, then the distance function will play a much more critical role. As all solutions are imperfect approximations, each will have a certain, non-zero error value. This value is determined by the nature of

the distance function. Different distance functions will be sensitive to different discrepancies (for instance, the phase plane trajectory ignores shifts in time). Thus, all approximations are ranked *according to the values determined by the nature of the distance functions employed*. Ultimately, only a subset of the possible solutions, those with the lowest distance values, will be selected. Thus, a poor choice of a distance function might cause a good approximation to be rejected or a poor approximation to be accepted.

The nature of the imperfect solutions found to be the best approximations by the distance function is thus clearly a crucial test for the applicability of a distance function. However, the reference model test does not challenge this aspect at all. As at least one perfect solution can be found, the problem of choosing amongst imperfect solutions should never arise (unless the convergence of the algorithm is terminated prematurely). Rather, the reference model test only evaluates the more technical (albeit still useful) question of whether the fitting method can pick out a perfect solution when such is known to exist. One way to extend the surrogate data approach to cases were some approximation must be chosen, while still maintaining knowledge of the parameter values that generated the target data, is to generate surrogate data from a given model yet employ the parameter constraining procedure on a reduced model lacking some of the channels, or on a model with inaccuracies introduced into the channel dynamics as performed by Hobbs and Hooper (2008). The parameter constraining strategy adopted by these authors is shown to be able to recover model parameters very well in the reference model test and is shown to be capable of dealing with the aforementioned inaccuracies. Thus, it holds promise to be successfully applied to experimental data.

4.2 Trace-to-trace comparison heavily emphasizes subthreshold regime

For the set of experimental data we considered in this study we find that, in our hands, those solutions chosen as good approximations of a given experimental trace by trace-to-trace comparison distance functions, as represented by the phase plane trajectory distance functions, are typically choices that heavily emphasize the subthreshold regime while more balanced choices between subthreshold and suprathreshold accuracy are rejected. We show that this is mainly due to the fact that the voltage data is dominated by the subthreshold regime and that trace-to-trace comparison assigns equal weight to every point in the voltage trace. Indeed, as action potentials are very brief events in time (in the order of milliseconds), if a cell fires at a rate of approximately 10 Hz then the voltage points pertaining to action potential will amount to one or two percent of the total voltage points. Thus, they are only a marginal contribution to the bulk of the voltage data. A distance function attempting to match a spiking trace point-by-point runs the risk of having the resulting models accurate in the subthreshold range, but in poor agreement with the experimental data in the suprathreshold range. In fact, due to this effect, a model that agrees with the experimental data in only the sub-threshold range, with no action potentials at all, can be potentially considered as more than 95% correct.

Clearly, these distance functions can be modified to at least partly overcome the above difficulties by different heuristics such as non-equal voltage point weighting, breaking up of the full voltage trace into different segments (for instance containing only a single spike), etc. Furthermore, using multiple objective optimization both feature-based and trace-to-trace comparisons can be combined and potentially the best of both worlds could be enjoyed. However, most importantly, the main point of the present study is not to discuss the utility of different distance functions but rather to highlight the fact that some methods may successfully pass the reference model test, yet produce poorer results in the real test of constraining model parameters in accordance with experimental data. Thus, the reference model test should be viewed as a necessary, not sufficient, test of a parameter fitting method.

4.3 Automated model parameter constraining in accordance with experimental data

Only a few studies concerning the automated parameter constraining of compartmental neuron models have actually applied their method directly to experimental data (Druckmann et al. 2007; Prinz et al. 2003; Vanier and Bower 1999) whereas other studies have applied the methods to surrogate model-generated data (Achard and De Schutter 2006; Bush et al. 2005; Gerken et al. 2006; Keren et al. 2005; Weaver and Wearne 2006). Intuitively, it makes perfect sense to test a parameter constraining procedure by applying it to surrogate data that was generated by a model with known parameters and then testing whether the procedure is able to recover the correct parameters. Indeed, this is a useful step in determining the utility of a parameter constraining procedure as it is a basic requirement from such a method. However, it is by no means the decisive step.

Indeed, as demonstrated in the present study, the main difficulty faced by the optimization procedure when dealing with experimental data resides in the fact that, given a set of assumed membrane channels, no perfect solution is likely to exist. Thus, the distance function(s) must rank the models, that all are but imperfect approximations, according to how well they approximate the data. The challenge is to construct a parameter constraining procedure that will result in a ranking of the models akin to what an experienced experimentalist/modeler would decide when comparing the experiments and the model results, thus offering alternative to hand-tuning parameter values. This capability is indeed

far more challenging than merely finding a perfect solution within a given search space.

It is interesting to note that the three studies mentioned above that compared model performance to experimental data relied on error functions that used multiple criteria. The studies of (Druckmann et al. 2007; Prinz et al. 2003) also incorporated the fact that neuronal responses to a sequentially repeated identical input are variable. Finally, the application of multiple objective optimization suggested by Druckmann et al. (2007) allows the seamless integration of multiple distance functions into a single optimization framework that addresses neuronal variability by using feature-based distance functions. Importantly, the same feature-based method could be also used to automatically classify the various firing patterns of neurons (work in progress).

Acknowledgments This work was supported by the Blue Brain Project and by the Israeli Science Foundation.

References

Achard P, De Schutter E (2006) Complex parameter landscape for a complex neuron model. PLoS Comput Biol 2:e94

Adiga NR, Almasi G, Almasi GS, Aridor Y, Barik R, Beece D, Bellofatto R, Bhanot G, Bickford R, Blumrich M, et al (2002) An Overview of the BlueGene/L Supercomputer. Paper presented at: Supercomputing, ACM/IEEE 2002 Conference

Bäck T, Fogel DB, Michalewics Z (1998) Handbook of evolutionary computation. Institute of Physics Publishing, Bristol

Bush K, Knight J, Anderson C (2005) Optimizing conductance parameters of cortical neural models via electrotonic partitions. Neural Netw 18:488–496

Carnevale NT, Hines ML (2005) The NEURON book. Cambridge University Press, Cambridge

Cohon JL (1985) Multicriteria programming: Brief review and application. In: Gero JS (ed) Design optimization. Academic Press, New York

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6:182–197

Druckmann S, Banitt Y, Gidon A, Schuermann F, Markram H, Segev I (2007) A novel multiple objective optimization framework for constraining conductance-based neuron models by experimental data. Front Neurosci 1:7–18

Gerken WC, Purvis LK, Butera RJ (2006) Genetic algorithm for optimization and specification of a neuron model. Neurocomputing 69:1039–1042

Hobbs KH, Hooper SL (2008) Using complicated, wide dynamic range driving to develop models of single neurons in single recording sessions. J Neurophysiol 99:1871–1883

Holland JH (1975) Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence. University of Michigan Press, Ann Arbor

Hwang CL, Masud ASM (1979) Multiple objective decision making, methods and applications : a state-of-the-art survey. Springer, Berlin

Keren N, Peled N, Korngreen A (2005) Constraining compartmental models using multiple voltage recordings and genetic algorithms. J Neurophysiol 94:3730–3742

LeMasson G, Maex R (2001) Introduction to equation solving and parameter fitting. In: De Schutter E (ed) Computational Neuroscience: Realistic Modeling for Experimentalists. CRC Press, London pp 1–21

Markram H, Lubke J, Frotscher M, Roth A, Sakmann B (1997) Physiology and anatomy of synaptic connections between thick tufted pyramidal neurones in the developing rat neocortex. J Physiol 500(Pt 2):409–440

Prinz AA, Billimoria CP, Marder E (2003) Alternative to hand-tuning conductance-based models: construction and analysis of databases of model neurons. J Neurophysiol 90:3998–4015

Vanier MC, Bower JM (1999) A comparative survey of automated parameter-search methods for compartmental neural models. J Comput Neurosci 7:149–171

Weaver CM, Wearne SL (2006) The role of action potential shape and parameter constraints in optimization of compartment models. Neurocomputing 69:1053–1057