ReScience

[Re] Fast-Activating Voltage- and Calcium-Dependent Potassium (BK) Conductance Promotes Bursting in Pituitary Cells: A Dynamic Clamp Study

Simen Tennøe[1, 2], **Kjetil Hodne**[3], **Trude M. Haug**[4], **Finn-Arne Weltzien**[3], **Gaute T. Einevoll**[1, 5, 6], **and Geir Halnes**[1, 5]

1 Centre for Integrative Neuroplasticity, University of Oslo, Oslo, Norway **2** Department of Informatics, University of Oslo, Oslo, Norway **3** Department of Basic Sciences and Aquatic Medicine, Norwegian University of Life Sciences, Campus Adamstuen, Norway **4** Institute of Oral Biology, University of Oslo, Oslo, Norway **5** Faculty of Science and Technology, Norwegian University of Life Sciences, Ås, Norway **6** Department of Physics, University of Oslo, Oslo, Norway

simenten@student.matnat.uio.no

Editor
Benoît Girard

Reviewers
Georgios Detorakis
Andrew P. Davison

Received Nov, 9, 2018
Accepted Mar, 11, 2019
Published Mar, 27, 2019

Competing Interests:
The authors have declared that no competing interests exist.

Article repository

Code repository

A reference implementation of

→ *Fast-Activating Voltage- and Calcium-Dependent Potassium (BK) Conductance Promotes Bursting in Pituitary Cells: A Dynamic Clamp Study*, J. Tabak, M. Tomaiuolo, A. Gonzalez-Iglesias, L. Milescu and R. Bertram, Journal of Neuroscience 31.46 (2011), 10.1523/JNEUROSCI.3235-11.2011

Introduction

As part of the dynamic clamp study by Tabak et al. 2011 [7], a computational model was developed for the voltage dynamics of endocrine pituitary cells in rats. The model captured the spontaneous activity of these cells, including the generation of Ca^{2+}-channel mediated spikes and pseudo-plateau bursts. As an important achievement, the model explained the paradoxical role that big conductance K^+ (BK) channels had in prolonging spike duration and sometimes promoting burst firing in these cells [9], contrary to what one would expect from a hyperpolarizing current. The original model was implemented in XPP [2]. The code for the model was made available online at https://www.math.fsu.edu/~bertram/software/pituitary/JNS_11b.ode, while the code used in the analysis of the model outcome was not made available.

In the current paper, we have reimplemented the computational model by Tabak et al. [7] using the Python interface for the NEURON simulator [5], a widely used simulator for multicompartmental neurons. In addition, we have performed an uncertainty quantification and sensitivity analysis of the model using the Uncertainpy Python package [8], version 1.1.4 (Zenodo: 10.5281/zenodo.1473453). The model implementation works with Python 2 and 3. The results in this paper were created using Python 3.7.0 within a Docker (https://www.docker.com/) environment.

The reimplemented model reproduced the characteristic firing patterns seen in the original publication, and we thus confirmed the original study. The sensitivity analysis further presented a systematic overview of the model in terms of how its characteristic response features depended on the various model parameters. Supporting the main conclusion from the original work, the sensitivity analysis showed that the bursting

propensity of the model was highly sensitive to the BK conductance. However, the analysis also revealed that the bursting propensity was sensitive to additional parameters (conductances), and thus that BK is not the sole determinant for whether the cell is bursty.

Methods

When reimplementing the model by Tabak et al. [7] we followed the descriptions in the original publication, using the original implementation for verification purposes. We also had a brief communication with the original authors to obtain details on the analysis part of the model.

Model

The model by Tabak et al. [7] was defined by the equation:

$$C\frac{dV}{dt} = -(I_{\mathrm{Ca}} + I_{\mathrm{K}} + I_{\mathrm{BK}} + I_{\mathrm{SK}} + I_{\mathrm{leak}} + I_{\mathrm{noise}}), \tag{1}$$

where C is the membrane capacitance, V is the membrane potential, and I_X the current through a specific ion channel X. The model included six different currents:

- I_{Ca} – Voltage gated Ca^{2+} current.
- I_{K} – Voltage gated K^{+} current.
- I_{BK} – Big conductance K^{+} current.
- I_{SK} – Small conductance K^{+} current.
- I_{leak} – Leak current.
- I_{noise} – Stochastic current representing channel noise.

A current through an ion channel X was given by the simplified relation:

$$I_X = G_X Y_X (V - E_X), \tag{2}$$

where G_X denotes the maximum ion channel conductance, and E_X denotes the reversal potential of the ion species conducted by channel X. Y_X denotes an ion channel specific gating function, which was unity for I_{leak}, an instantaneous function of V for I_{Ca}, an instantaneous function of the cytosolic Ca^{2+} concentration for I_{SK}, and a dynamic function of V and t for the remaining ion channels I_{K}, I_{BK}, and I_{K}.

The original implementation used the total membrane capacitance (units pF) and total membrane conductances (units nS), while the NEURON simulator requires these entities to be specified per membrane area with units $\mu\mathrm{F/cm}^2$ and $\mathrm{S/cm}^2$, respectively. NEURON also requires that the membrane area is defined. To get the parameters on the form required by NEURON we defined an arbitrary membrane area (A), and divided the capacitance C and ion channel conductances G_X by A:

$$g_{X,\,\mathrm{NEURON}} = \frac{G_X}{A}, \qquad c_{\mathrm{NEURON}} = \frac{C}{A}. \tag{3}$$

Combining Equation 1, 2 and 3 shows that the model is independent of the choice of A:

$$\frac{C}{A}\frac{dV}{dt} = -\left(\frac{G_X}{A} Y_X (V - E_X) + \ldots\right). \tag{4}$$

The original model further included an equation for handling the intracellular Ca^{2+} concentration, which is relevant for the gating of SK channels:

$$\frac{d[Ca]}{dt} = -f_c(\alpha I_{\mathrm{Ca}} + k_c[Ca]), \tag{5}$$

where f_c denotes the fraction of free Ca^{2+} in the cytoplasm, k_c denotes the extrusion rate, and the constant α converts an incoming current to a molar concentration. α was converted to NEURON units by taking:

$$\alpha_{\mathrm{NEURON}} = A\alpha. \tag{6}$$

Combining Equation 3, 5 and 6 shows that this choice keeps the model independent of the choice of A:

$$\frac{d[Ca]}{dt} = -f_c(A\alpha\frac{G_{\mathrm{Ca}}}{A}(V - E_{\mathrm{Ca}}) + k_c[Ca]). \tag{7}$$

We arbitrarily chose a cell body with a membrane area of $\pi \cdot 10^{-6}$ cm^2, i.e. with a diameter of 10 μm. We used all equations from the original publication, substituting G_X with $g_{X,\,\mathrm{NEURON}}$, C with c_{NEURON}, and α with α_{NEURON}. The parameter values from the original publication and the converted parameter values are summarized in Table 1. Parameters not listed in this table were kept unchanged from the original publication. To make the discussion and results easier to compare to the original publication, we will refer to the original conductance values through the rest of this paper.

The noise was added by using a current clamp that injected a random current at each time step in the simulation, as described by the original publication. Simulations with noise were run with a fixed time step of `dt = 0.01` ms, which is the same time step used in the original publication. When performing the sensitivity analysis, the noise amplitude was set to zero ($A_{\mathrm{noise}} = 0$), and the simulations were run using adaptive time steps.

We found one discrepancy between the parameters listed in the original publication and the values found in the original source code. The maximum conductance of K^+ channels (G_{K}) was listed as 3.2 nS in the original publication, while the value used in the original source code was 3 nS. Both values were tested and $G_{\mathrm{K}} = 3$ nS gave results most similar to the results in the original publication. We therefore decided to use $G_{\mathrm{K}} = 3$ nS instead of the value listed in the original publication.

Table 1: The parameter values in Tabak et al. [7] that were converted from currents and capacitance to currents and capacitance per membrane area due to requirements by the NEURON simulator. The original model parameter values are denoted Tabak while the parameter values in the reimplemented model are denoted NEURON, with names as used in the model implementation.

Tabak	Value	Unit	NEURON	Value	Unit
			`A`	$3.14 \cdot 10^{-6}$	cm^2
C	10	pF	`c`	3.18	μF/cm^2
G_{Ca}	2	nS	`g_Ca`	$6.37 \cdot 10^{-4}$	S/cm^2
G_{K}	3	nS	`g_K`	$9.55 \cdot 10^{-4}$	S/cm^2
G_{BK}	2	nS	`g_BK`	0	S/cm^2
G_{SK}	2	nS	`g_SK`	$6.37 \cdot 10^{-4}$	S/cm^2
G_{l}	0.2	nS	`g_l`	$6.37 \cdot 10^{-5}$	S/cm^2
α	0.0015	μM/fC^2	`alpha`	$4.71 \cdot 10^{-3}$	mM $\cdot$ cm$^2/\mu$C
A_{noise}	4	pA	`noise_amplitude`	0.004	nA

Event detection

In the analysis, we ran the model for 60000 ms and discarded the first 10000 ms of the voltage trace to eliminate the transient initial response.

The first step of the model analysis was to detect events (spikes or bursts) in the model voltage trace. To do this, the voltage was normalized so that its minimum value was set to 0 and its maximum was set to 1. The start of an event was specified to be when the voltage crossed an onset threshold (defined to be 0.55), and the end of an event to be when it next descended below another, lower termination threshold (defined to be 0.45). An event includes the first point before it crossed the onset threshold and the first point after it descended below the termination threshold.

The difference in onset and termination threshold was necessary to prevent random fluctuations around the threshold (during upstroke or downstroke) to be considered as independent events. If the voltage trace started above the onset threshold, we discarded the first part of the voltage trace until we got below the termination threshold. Similarly, if an event did not fall below the termination threshold before the simulation ended, that event was discarded. Additionally, we required that events have an amplitude of at least 10 mV. This prevents the problem where the normalization step leads to detecting false events with an amplitude less than 1 mV in cases where the model does not generate any events and instead exhibits small (much less than 1 mV) fluctuations around a steady state.

We used Uncertainpy to detect events, as the described threshold-detection algorithm is available to us by using the `uncertainpy.Spikes` object with the arguments `normalize=True`, `trim=False` and `min_amplitude = 10`. Note that in Uncertainpy the `end_threshold` is given relative to the onset threshold, so to get a termination threshold = 0.45 we set `end_threshold = -0.1`.

An event was defined as a burst when its duration was longer than a given threshold (60 ms). The burstiness factor was defined as the fraction of the total number of events that were considered as bursts. All parameters used in the analysis are summarized in Table 2.

The description of the threshold-detection algorithm for detecting events (bursts or spikes) was incomplete in the original publication. We contacted the original authors, who were helpful in describing the threshold-detection algorithm, but who did not recall the exact numerical values of all threshold choices. The onset threshold, termination threshold and burst-duration threshold used (Table 2) were therefore set to the values we found to give the best agreement between our analysis outcome and that in the original publication.

Table 2: The parameters used in the analysis of the model.

Parameter	Value	Unit
Simulation time	60000	ms
Discard	10000	ms
Event onset threshold	0.55	
Event termination threshold	0.45	
Burst threshold	60	ms
Minimum event amplitude	10	mV

Uncertainty quantification and sensitivity analysis

We used Uncertainpy to further examine the model through an uncertainty quantification and sensitivity analysis. This enabled us to quantify how sensitive salient response properties of the model is to changes in the various parameters. In the sensitivity analysis, the four conductances G_{Ca}, G_{K}, G_{SK}, and G_{l} were assigned uniform

distributions within ±50% of their original values. G_{BK}, which had no default value in the original model, was given a uniform distribution between 0 and 1 nS as this was the parameter range explored in the original study. We use polynomial chaos with the point collocation method (the default of Uncertainpy) and a polynomial order of eight. In the sensitivity analysis, we wanted all the variance in the simulation outcome to reflect parameter variations, and the random noise was therefore turned off by setting $A_{noise} = 0$.

We calculated the uncertainty and sensitivity of the five features of the model:

- Event rate, which is the event firing rate (named `spike_rate` in Uncertainpy).
- Average event peak, which is the average event peak voltage (named `average_AP_overshoot` in Uncertainpy).
- Average AHP (afterhyperpolarization) depth, which is the average minimum voltage between events.
- Burstiness factor, the fraction of events with a duration longer than 60 ms.
- Average duration, the average duration of an event.

Some of these features are not defined for all parameter combinations (for example average AHP depth is not defined when there are no events). The point collocation method still gives reliable results, as long as the features are defined for a sufficiently large fraction of the parameter combinations (in our case the lowest was ~91.5%) [1].

Some of the outcomes from the sensitivity analysis were unexpected and were explored further by varying selected parameters and documenting how these variations affected the average event duration and burstiness factor of the model. The parameters varied in this additional analysis were G_{BK}, G_{SK}, and G_K, all of which were varied within the range used in the uncertainty analysis.

Results

We repeated all simulations and qualitatively reproduced Figure 1 and Figure 2 in Tabak et al. [7]. The remaining results in the original publication were experimental results, and therefore outside the scope of this reproduction. The results shown in Figure 1 correspond well to those in Figure 1 of the original publication. The original model and the reproduced version showed the same behavior when increasing G_{BK}.

The results shown in Figure 2 correspond well to those in Figure 2 of the original publication. In this figure, G_{BK} was fixed at a given value, while the remaining model parameters were sampled randomly (see Methods). When G_{BK} was set to zero (Figure 2**A**), most model parameterizations had a low burstiness factor (between 0 and 0.1). Oppositely, when G_{BK} was fixed at the maximum value (Figure 2**C**), most model parameterizations had a high burstiness factor (between 0.9 and 1). For any value of G_{BK}, the model evaluations tended to be either predominantly bursting (i.e. most events were bursts) or predominantly spiking (few events were bursts), so that the number of model evaluations with an intermediate burstiness factor between 0.1 and 0.9 (events changed between being bursts or not) was always low (less than 20 evaluations).

Albeit qualitatively similar, the presented results were not strictly identical to those in the original publication. As random noise was added to the simulations, exact replications were unattainable, and some discrepancies between the current and the original study were expected. The largest deviations between the current analysis and the original work were seen in Figure 1**B** (right panel), where the original work found a burstiness factor of 0.34, while our analysis found a burstiness factor of 0.42, and in Figure 2**C**, where the number of model evaluations with a low burstiness factor was much higher (approximately 75) in the original work compared to what we found in the current analysis (16). Below, we analyze whether the observed discrepancies

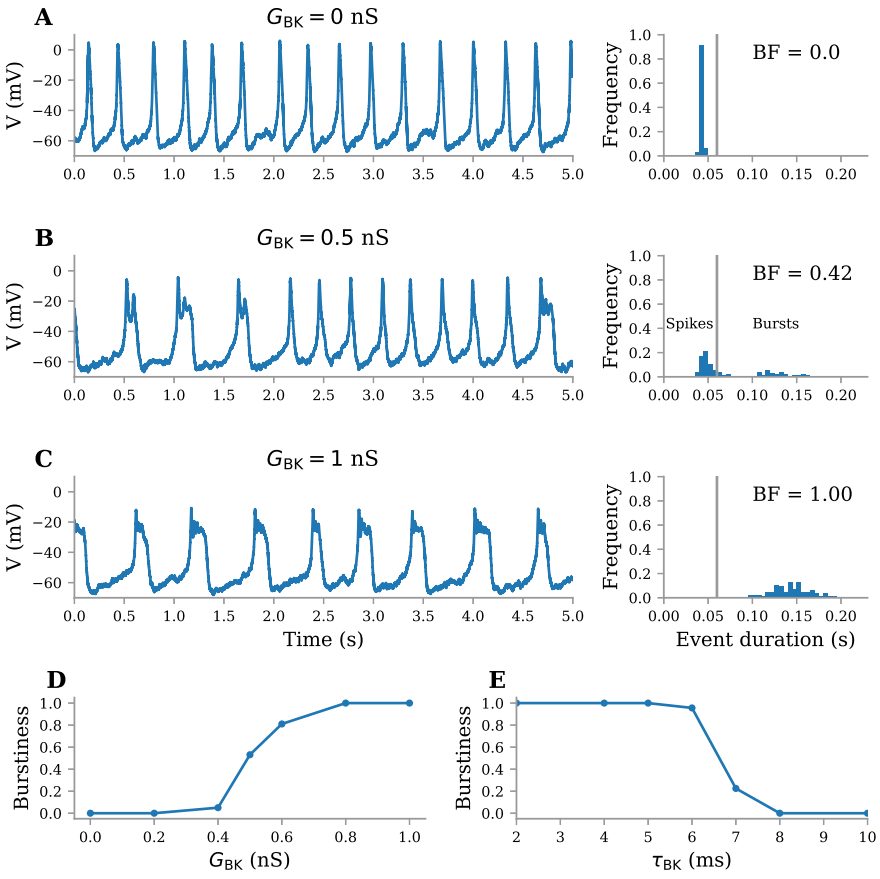

Figure 1: Model predictions for the effect of various G_{BK} conductances on burstiness. **A**-**C** Left, membrane potential of the model. Right, distribution of event durations in the time interval from 1 to 5 s (of the 50 s simulated). The grey line indicates the threshold for what is considered a spike and what is considered a burst, and BF denotes the burstiness factor. **D** The burstiness factor increased with G_{BK}. **E** The burstiness factor decreased with τ_{BK}.

can be ascribed to noise, or if they may reveal other differences in implementation details. We limit the analysis to only consider the above mentioned two cases.

Examining the discrepancies between the replicated and original simulations

Below, we examine whether the discrepancies between the current and original work, as reflected in Figure 1**B** and Figure 2**C**, can be explained by (i) the random noise added in the simulations, (ii) differences in integration methods, as reflected by the simulation time step dt, (iii) numerical floating point errors introduced when converting from total conductances to conductance per-unit-area by choosing a membrane area A, or if (iv) the discrepancies may reflect unintended differences in the algorithms for the model analysis.

We start by exploring the burstiness factor (BF) in Figure 1**B**, which reflects the fraction of events that were bursts in a single simulation. As the burstiness factor varies from simulation to simulation (due to noise), we calculated it for 100 reruns of the model, which allowed us to calculate its mean and standard deviation (Figure 3). We calculated the mean and standard deviations for three different values of the area ($A = \pi \cdot 10^{-9}; \pi \cdot 10^{-6}; \pi \cdot 10^{-3}$) , and three values of the timestep ($\mathrm{dt} = 0.05; 0.005; 0.001$), and the obtained statistics did not vary much with these model choices. In all cases, the burstiness factor had a mean of about 0.4 and a standard deviation of about 0.04. The burstiness factor of 0.34 found in the original work was thus roughly a standard deviation lower than the mean found here, and it is thus not too unlikely that the observed discrepancies could be due to random noise (i).

Next, we explored how the results presented in Figure 2**C** depended on the choice of membrane area and simulation time step. Figure 4 shows the burstiness of the model for three values of the area ($A = \pi \cdot 10^{-9}, \pi \cdot 10^{-3}, 1$) and three values of the timestep ($\mathrm{dt} = 0.05, 0.005, 0.001$), changing one of them at the time. When comparing Figure 4 to Figure 2**C** we see that changes in either the timestep or the area causes very little changes in the overall results. The small variations between the different panels in Figure 4 are thus most likely due to the random noise. In this case, neither noise (i), integration method differences (ii), or floating point errors (iii) seem like a likely explanation of the discrepancies between the original simulation and the current results.

The above indicates that there might be some small differences between the current and original implementation that are not due to noise or numerical issues (iv). We were not able to detect the precise cause of the observed difference. We believe that the model itself is identical in the two cases (we have compared our code with the XPP code from the original work [7]) and that the differences are more likely to reflect differences in the choices made in the analysis part. These could be choices regarding event detection (such as the definition of onset and termination thresholds and burst definition), or criteria for which simulations that were included in and discarded from the analyses. These choices were imprecisely described in the original work, and it is possible that we have made some minor misinterpretations, and that our criteria deviate slightly from the original ones. In the original work, the robustness results (Figure 2) in the original publication were calculated using custom developed software that used CUDA to run on GPUs, and the code for performing this analysis was not available.

Uncertainty quantification and sensitivity analysis

The uncertainty quantification and sensitivity analysis of the model is shown in Figure 5. The sensitivity was given as the total-order Sobol indices, which quantify how much of the variance of the model each parameter (accounting for all of its interactions with other parameters) is responsible for [6].

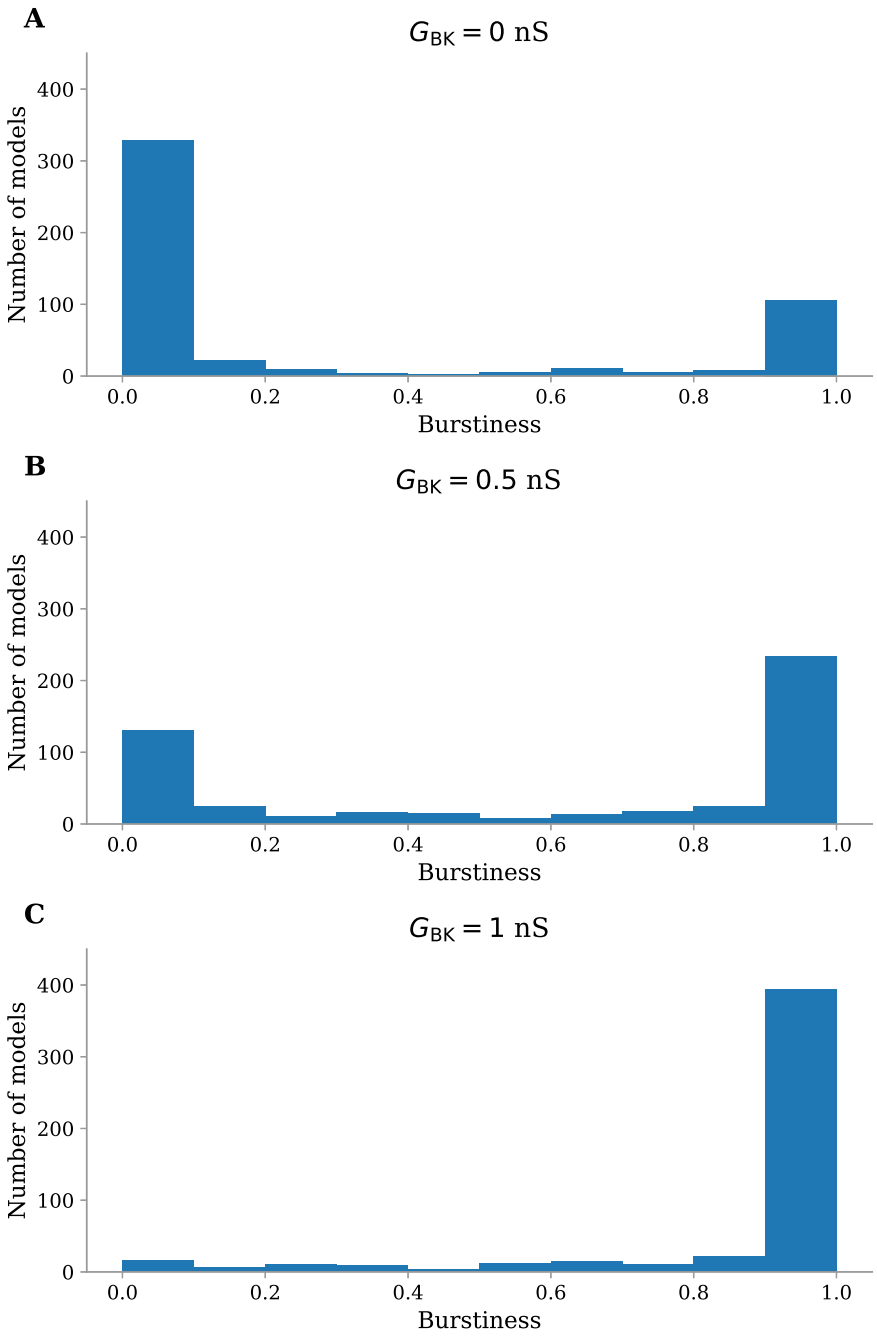

Figure 2: Robustness of the burstiness of the model for three values of G_{BK} when changing G_{Ca}, G_{K}, G_{SK}, and G_{l} uniformly within $\pm 50\%$ of their original values. **A** For $G_{\mathrm{BK}} \to 0$ nS, 67.5% of the active models were spikers (burstiness factor < 0.3). **B** For $G_{\mathrm{BK}} \to 0.5$ nS, 33.8% were spikers. **C** For $G_{\mathrm{BK}} \to 1$ nS, only 4.4% were spikers.

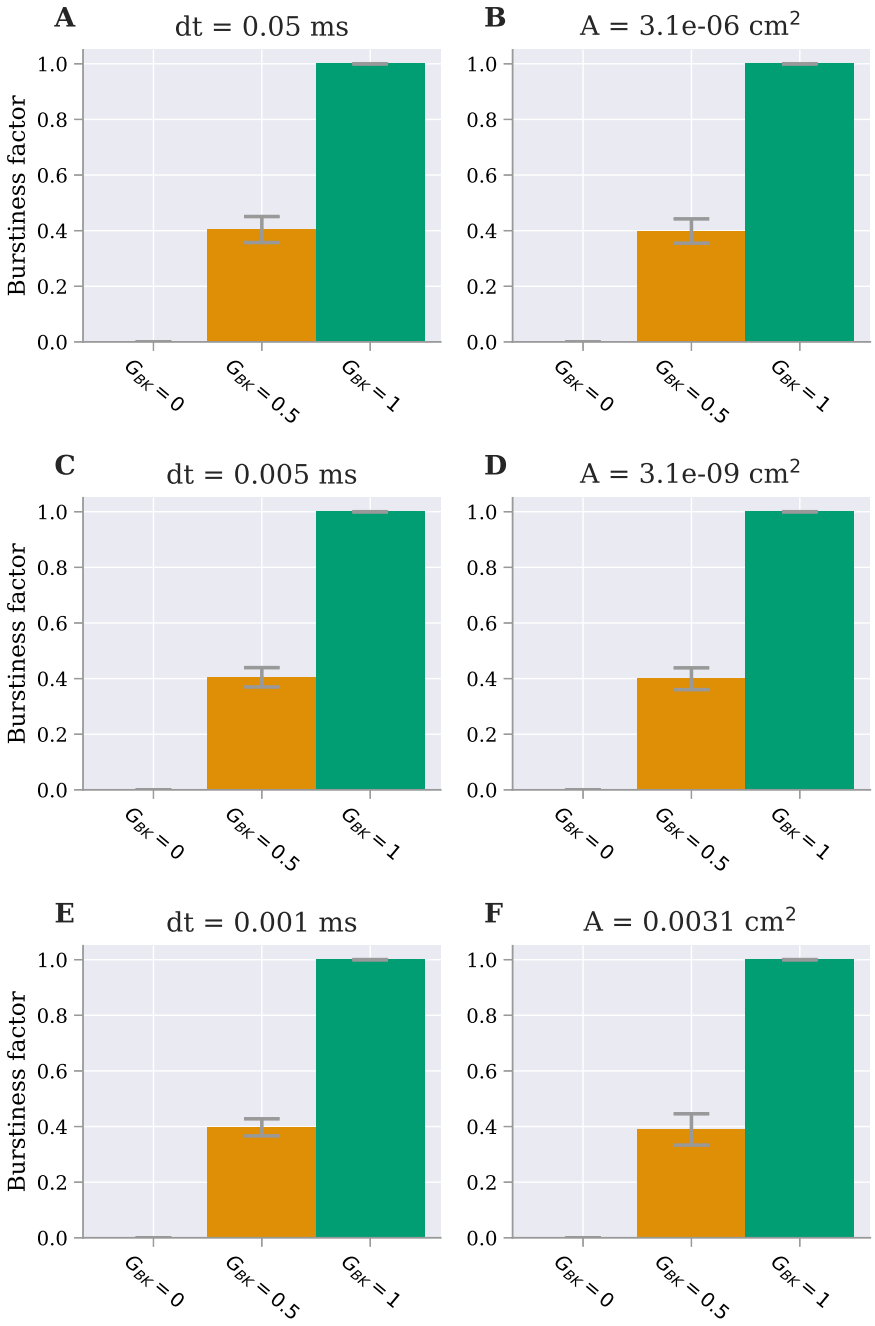

Figure 3: Mean and standard deviation of the burstiness factor of 100 reruns of the model for three values of the area (A), three values of the timestep ((dt), and three values of G_{BK}. Panel **B** shows the results for model using the original parameters.

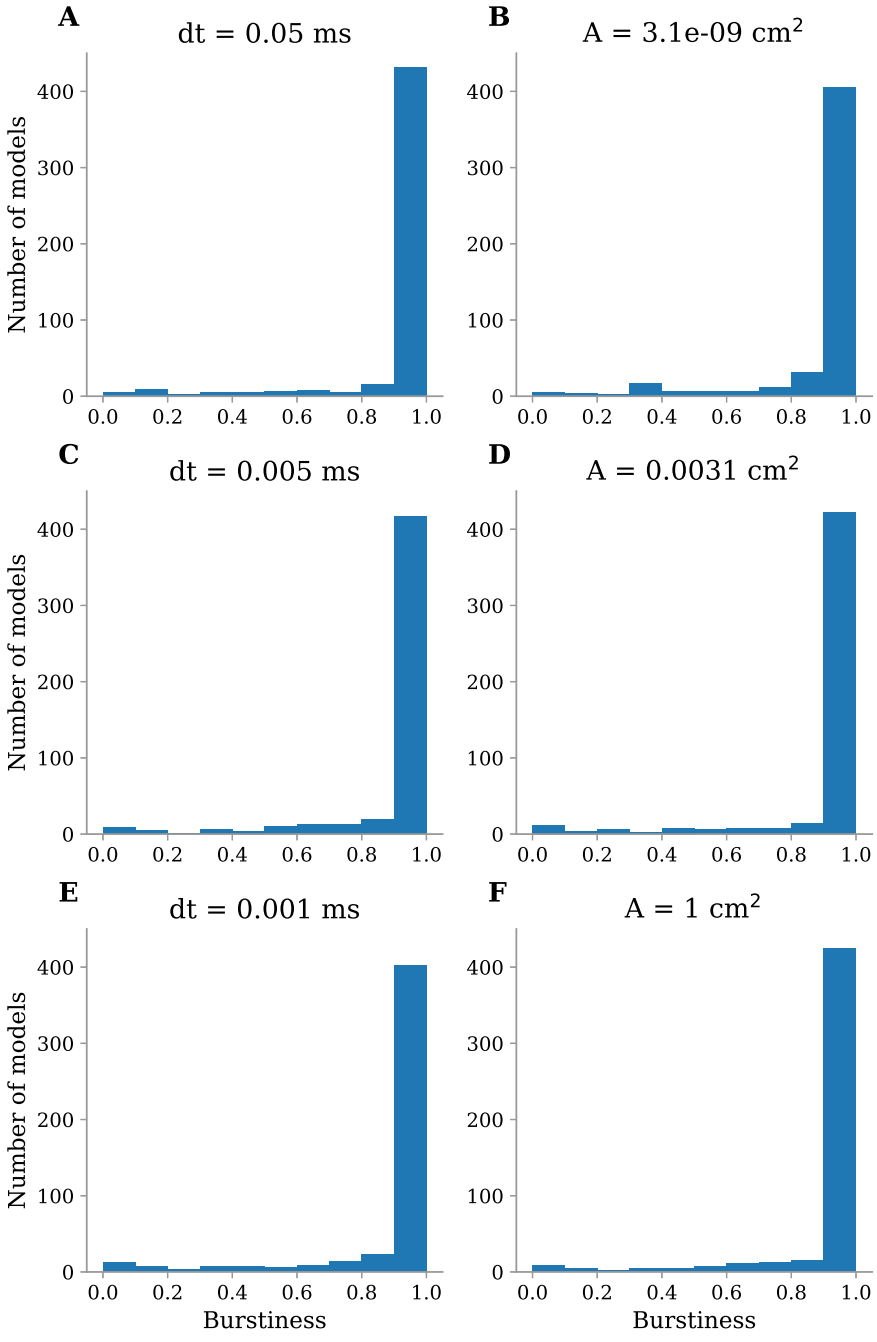

Figure 4: Burstiness of the model for three values of the area and three values of the timestep when changing G_{Ca}, G_{K}, G_{SK}, and G_{l} uniformly within $\pm 50\%$ of their original values. When changing the area the timestep was kept unchanged at $\mathrm{dt} = 0.01$ and when changing the timestep the area was kept unchanged at $\pi \cdot 10^{-6}$. $G_{\mathrm{BK}} = 1$ as that is where we observed the difference.

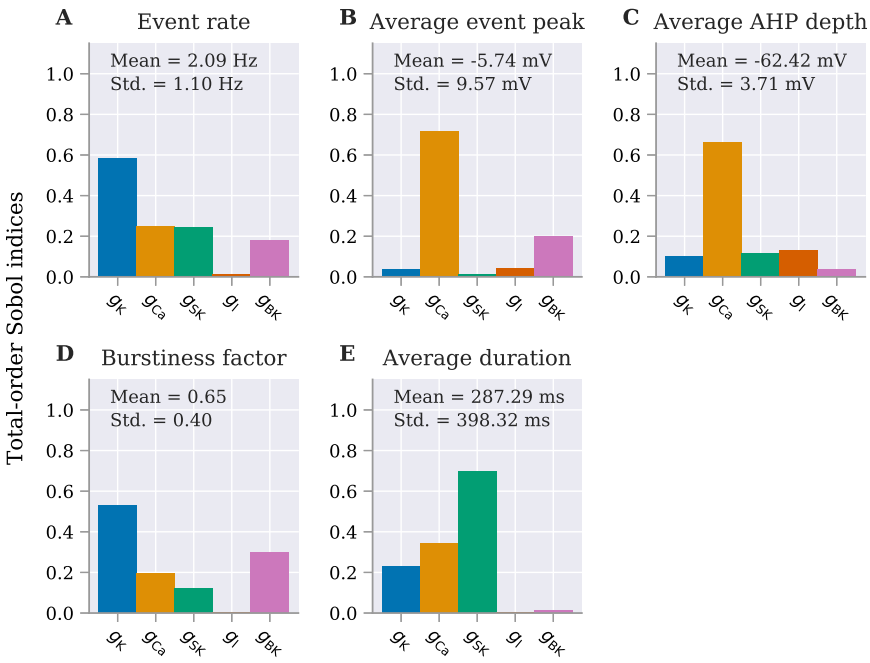

Figure 5: Uncertainty quantification and sensitivity analysis of a selected set of response features of the model. **A** Event rate denotes the event firing rate. **B** Average event peak denotes is the average event peak voltage. **C** Average AHP (afterhyperpolarization) depth denotes the average minimum voltage between two consecutive events. **D** Burstiness factor denotes the fraction of events with duration longer than 60 ms. **E** Average duration denotes the average duration of the events.

The sensitivity analysis showed that the spike rate was sensitive to almost all ion channel conductances, but most so to G_K (Figure 5**A**). Such a role of the delayed rectifying K^+ channel in controlling the firing rate has been seen in other studies [3].

The event amplitude was mainly sensitive to G_{Ca} (Figure 5**B**), which is not surprising given that the events are generated by I_{Ca}. However, it also had a relatively high sensitivity to G_{BK}, in line with what was found in the previous study [7].

The average afterhyperpolarization depth was in turn most sensitive to G_{Ca} (Figure 5**C**). This may seem counterintuitive, as I_{Ca} is not a hyperpolarizing current. However, I_{Ca} is responsible for triggering all the three hyperpolarizing currents (I_K, I_{BK} and I_{SK}) that generate the afterhyperpolarization depth. I_K and I_{BK} are activated by the voltage deflection caused by I_{Ca}, while I_{SK} is activated by the Ca^{2+} entering through I_{Ca}.

The burstiness factor of the model was mainly sensitive to G_K and G_{BK} (Figure 5**D**). The sensitivity to G_{BK} confirms the findings in the original publication, i.e. that BK channels promote bursting. However, the large sensitivity to G_K is a novel insight for the current study and indicates that also G_K was important for determining if the model produced bursts or spikes. This observation is tightly related to the explanation for how BK can act as a burst promoter in the first place, which is contrary to what one would expect from a hyperpolarizing current. The explanation, proposed by both Tabak et al. [7] and the experimental studies they were inspired by [9], was that G_{BK} promoted bursting by reducing the peak amplitude of events (as reflected in Figure 5**B**), thereby preventing full activation of the otherwise more strongly hyperpolarizing delayed rectifier current (I_K). In this context, the sensitivity analysis simply shows that the indirect effect on I_K obtained by varying G_{BK} was smaller than the direct effect on I_K obtained by varying G_K (Figure 5**D**).

Surprisingly, the average event duration had a very low sensitivity to G_{BK} (Figure 5**E**), and was instead most sensitive to G_{SK}. This was unexpected since the burstiness was highly sensitive to G_{BK}, and a burst was defined as an event exceeding a certain duration. An exploration of the counterintuitive relationship between Figure 5**D** and **E** is presented below.

Parameter exploration

To explore the relationship between the results in Figure 5**D** and **E**, we examined the effects of varying G_{BK}, G_K, and G_{SK} on the burstiness and the average duration of events (Figure 6). It should be noted that this figure only shows how the model responds when changing two parameters at the time, so the higher-order interactions included in the total-order Sobol sensitivity indices are absent.

Figure 6**A** shows the regions in the G_{BK}/G_K parameter plane where the model produced regular spikes (yellow) and bursts (green). For low (< 2 nS) values of G_K, the cell was bursting regardless of the value of G_{BK}. Hence, for low values of G_K, the burstiness of the cell was insensitive to G_{BK}. In comparison, a sufficiently large change in G_K could switch the cell between a regular and bursty state for any (fixed) value of G_{BK}. These results thus fit well with the sensitivity analysis in Figure 5**D**, which showed that the burstiness was more sensitive to G_K than to G_{BK}.

We next fixed G_K at the default value 3 nS, and explored how it could be the case that burstiness was sensitive to G_{BK} but not so much to G_{SK}, while event duration was sensitive to G_{SK} but not so much to G_{BK} (Figure 6**B**). As the figure shows, for $G_{BK} < 0.2$ nS the cell was always regularly spiking, while for $G_{BK} > 0.8$ nS, the cell was always bursting, irregardless of the values of G_{SK}. In comparison, changing G_{BK} (keeping G_{SK} fixed) could always switch the cell between a regular and bursty state. In equivalence with the analysis of Figure 6**A**, this explains why the burstiness was less sensitive to G_{SK} than G_{BK}. However, although changes in G_{BK} more often led to changes in burstiness, the effects on the event duration was modest. That is, for most (fixed) values of $G_{SK} >$, a change in G_{BK} could push the event duration

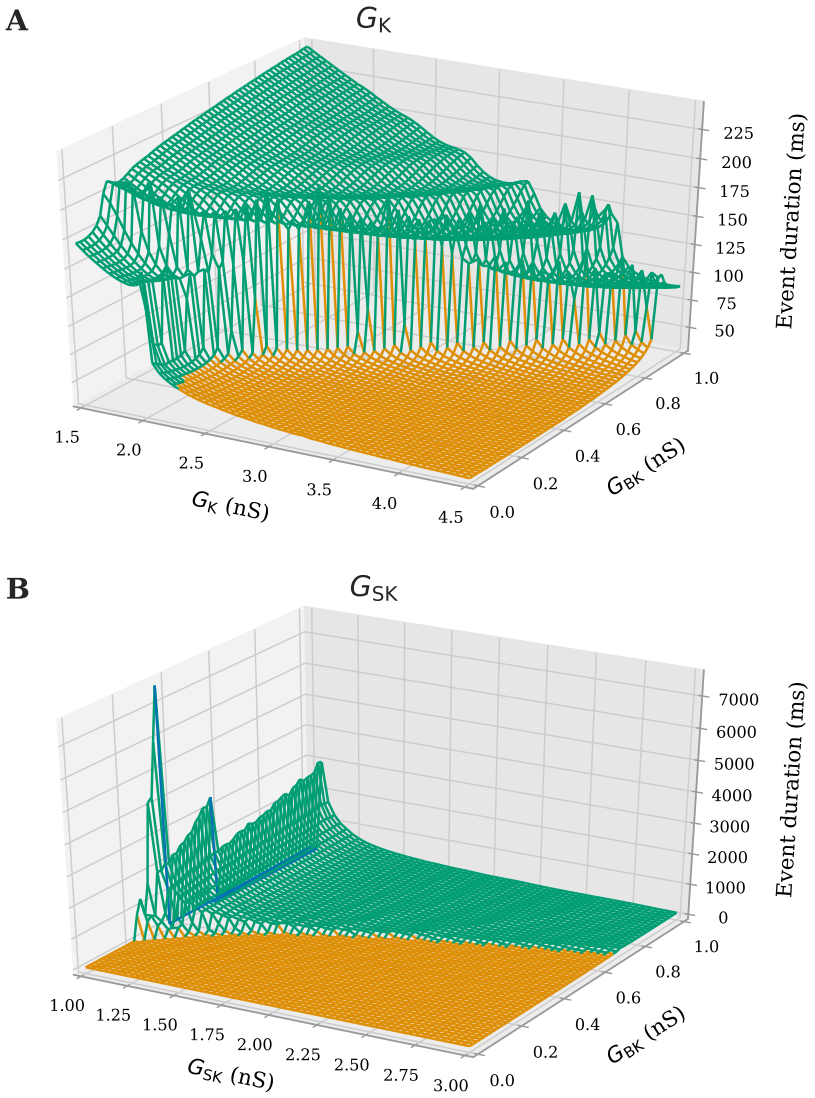

Figure 6: The average duration of events while varying G_{BK} and either **A** G_{K} or **B** G_{SK}. The areas in parameter space where the average duration of the events is longer than the burstiness factor threshold are in green, while the areas where the average duration is below this threshold are in yellow. Areas in blue produce no events and the average duration is then set to -1 for visualization purposes.

from slightly below to slightly above the burst-duration threshold, but did not lead to larger changes in burstiness. Oppositely, reducing G_{SK} to the lower values in the explored range resulted in burst durations of several thousands of milliseconds (as long as $G_{BK} > 0.2$ nS). Hence, while G_{BK} was important for achieving a burst in the first place, G_{SK} had a much larger impact on the duration of the burst. This explains the difference in sensitivity between the average duration and burstiness factor observed in Figure 5**D** and **E**.

Conclusion

We were able to qualitatively reproduce all the computational results in Tabak et al. [7]. By performing an uncertainty quantification and sensitivity analysis we confirmed the key conclusions in the original publication using a different simulator and different analysis methods, which provided additional insight into how different membrane mechanisms interact to produce the characteristic response features of the model. Overall, the reproduction effort went smoothly, with a little help from the original authors in describing the threshold-detection algorithm used in the analysis of the model. The original model now exists as a model using the Python interface for NEURON, which hopefully makes it accessible to a wider audience. Our personal motivation for reproducing the model by Tabak et al. was that we needed it in a computational study where we compared the dynamical properties of pituitary cells in rats versus fish [4].

References

[1] Vinzenz Gregor Eck et al. "A guide to uncertainty quantification and sensitivity analysis for cardiovascular applications". In: *International Journal for Numerical Methods in Biomedical Engineering* 32.8 (2016), e02755–n/a. ISSN: 20407947. DOI: 10.1002/cnm.2755. arXiv: NIHMS150003. URL: http://dx.doi.org/10.1002/cnm.2755.

[2] B. Ermentrout. *Simulating, Analyzing, and Animating Dynamical Systems*. Society for Industrial and Applied Mathematics, 2002. DOI: 10.1137/1.9780898718195. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9780898718195. URL: https://epubs.siam.org/doi/abs/10.1137/1.9780898718195.

[3] Dongxu Guan, William E. Armstrong, and Robert C. Foehring. "Kv2 channels regulate firing rate in pyramidal neurons from rat sensorimotor cortex". In: *The Journal of Physiology* 591.19 (2013), pp. 4807–4825. DOI: 10.1113/jphysiol.2013.257253. eprint: https://physoc.onlinelibrary.wiley.com/doi/pdf/10.1113/jphysiol.2013.257253. URL: https://physoc.onlinelibrary.wiley.com/doi/abs/10.1113/jphysiol.2013.257253.

[4] Geir Halnes et al. "BK channels have opposite effects on sodium versus calcium-mediated action potentials in endocrine pituitary cells." In: *bioRxiv* (2018). DOI: 10.1101/477976. eprint: https://www.biorxiv.org/content/early/2018/11/23/477976.full.pdf. URL: https://www.biorxiv.org/content/early/2018/11/23/477976.

[5] M. L. Hines and N. T. Carnevale. "The NEURON Simulation Environment". In: *Neural Computation* 9.6 (1997), pp. 1179–1209. ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.6.1179. arXiv: 1501.01854. URL: http://www.mitpressjournals.org/doi/10.1162/neco.1997.9.6.1179.

[6] Toshimitsu Homma and Andrea Saltelli. "Importance measures in global sensitivity analysis of nonlinear models". In: *Reliability Engineering & System Safety* 52.1 (1996), pp. 1–17. ISSN: 09518320. DOI: 10.1016/0951-8320(96)00002-6. URL: http://www.sciencedirect.com/science/article/pii/0951832096000026.

[7] Joël Tabak et al. "Fast-Activating Voltage- and Calcium-Dependent Potassium (BK) Conductance Promotes Bursting in Pituitary Cells: A Dynamic Clamp Study". In: *Journal of Neuroscience* 31.46 (2011), pp. 16855–16863. ISSN: 0270-6474. DOI: 10.1523/JNEUROSCI.3235-11.2011. eprint: http://www.jneurosci.org/content/31/46/16855.full.pdf. URL: http://www.jneurosci.org/content/31/46/16855.

[8] Simen Tennøe, Geir Halnes, and Gaute T. Einevoll. "Uncertainpy: A Python Toolbox for Uncertainty Quantification and Sensitivity Analysis in Computational Neuroscience". In: *Frontiers in Neuroinformatics* 12 (2018), p. 49. ISSN: 1662-5196. DOI: 10.3389/fninf.2018.00049. URL: https://www.frontiersin.org/article/10.3389/fninf.2018.00049.

[9] Fredrick Van Goor, Yue-Xian Li, and Stanko S. Stojilkovic. "Paradoxical Role of Large-Conductance Calcium-Activated K+ (BK) Channels in Controlling Action Potential-Driven Ca2+ Entry in Anterior Pituitary Cells". In: *Journal of Neuroscience* 21.16 (2001), pp. 5902–5915. ISSN: 0270-6474. DOI: 10.1523/JNEUROSCI.21-16-05902.2001. eprint: http://www.jneurosci.org/content/21/16/5902.full.pdf. URL: http://www.jneurosci.org/content/21/16/5902.