



Perfect Detection of Spikes in the Linear Sub-threshold Dynamics of Point Neurons

Jeyashree Krishnan^{1,2,3*}, PierGianLuca Porta Mana¹, Moritz Helias^{1,4}, Markus Diesmann^{1,4,5} and Edoardo Di Napoli^{2,3}

¹ Institute of Neuroscience and Medicine (INM-6) and Institute for Advanced Simulation (IAS-6) and JARA Institute Brain Structure Function Relationship (INM-10), Jülich Research Centre, Jülich, Germany, ² Aachen Institute for Advanced Study in Computational Engineering Science, RWTH Aachen University, Aachen, Germany, ³ Institute for Advanced Simulation, Jülich Research Centre, Jülich, Germany, ⁴ Department of Physics, Faculty 1, RWTH Aachen University, Aachen, Germany, ⁵ Department of Psychiatry, Psychotherapy and Psychosomatics, Medical Faculty, RWTH Aachen University, Aachen, Germany

OPEN ACCESS

Edited by:

Pedro Antonio Valdes-Sosa,
Joint China-Cuba Laboratory for
Frontier Research in Translational
Neurotechnology, China

Reviewed by:

Marc De Kamps,
University of Leeds, United Kingdom
Mikael Djurfeldt,
Royal Institute of Technology, Sweden

*Correspondence:

Jeyashree Krishnan
krishnan@ices.rwth-aachen.de

Received: 25 June 2017

Accepted: 15 December 2017

Published: 05 January 2018

Citation:

Krishnan J, Porta Mana P, Helias M, Diesmann M and Di Napoli E (2018) Perfect Detection of Spikes in the Linear Sub-threshold Dynamics of Point Neurons. *Front. Neuroinform.* 11:75. doi: 10.3389/fninf.2017.00075

Spiking neuronal networks are usually simulated with one of three main schemes: the classical time-driven and event-driven schemes, and the more recent hybrid scheme. All three schemes evolve the state of a neuron through a series of checkpoints: equally spaced in the first scheme and determined neuron-wise by spike events in the latter two. The time-driven and the hybrid scheme determine whether the membrane potential of a neuron crosses a threshold at the end of the time interval between consecutive checkpoints. Threshold crossing can, however, occur within the interval even if this test is negative. Spikes can therefore be missed. The present work offers an alternative geometric point of view on neuronal dynamics, and derives, implements, and benchmarks a method for perfect retrospective spike detection. This method can be applied to neuron models with affine or linear subthreshold dynamics. The idea behind the method is to propagate the threshold with a time-inverted dynamics, testing whether the threshold crosses the neuron state to be evolved, rather than vice versa. Algebraically this translates into a set of inequalities necessary and sufficient for threshold crossing. This test is slower than the imperfect one, but can be optimized in several ways. Comparison confirms earlier results that the imperfect tests rarely miss spikes (less than a fraction $1/10^8$ of missed spikes) in biologically relevant settings.

Keywords: state-space analysis, NEST, time-driven, event-driven, simulation, LIF neuron, differential geometry

1. INTRODUCTION

In the last decade considerable work has been devoted to improving the accuracy of efficient simulators of large networks of spiking neurons (Hansel et al., 1998; Mattia and Del Giudice, 2000; Shelley and Tao, 2001; Dehaene and Changeux, 2005; Brette, 2007; Morrison et al., 2007; D'Haene et al., 2009; van Elburg and van Ooyen, 2009; Zheng et al., 2009; Hanuschkin et al., 2010). This research field is driven by the ideal of combining two antagonistic goals: coping with a realistically high frequency of synaptic events arriving at a neuron, as in nature, and at the same time implementing in a mathematically accurate way the threshold process on which a wide class of neuron models is based.

Two classical schemes to simulate neuronal networks are the time-driven and the event-driven schemes (Ferscha, 1996; Fujimoto, 2000; Zeigler et al., 2000). Both schemes describe the state of the neurons by a set of variables and the action potentials as events that mediate the interaction between them.

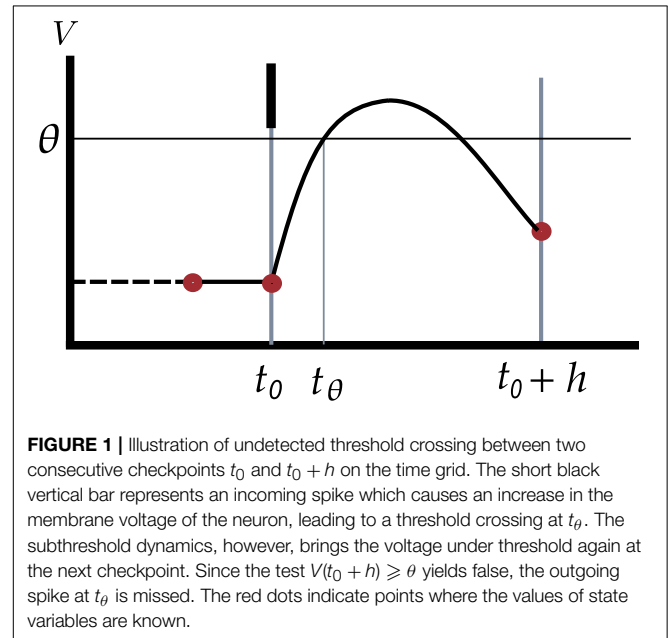
In a time-driven scheme, the state of a neuron is updated on a time grid defined by the simulation step (for a review see Morrison and Diesmann, 2008). After all neurons are updated, their membrane potential is checked for threshold crossings. If the membrane potential of a neuron is above the threshold at this checkpoint, a spike is delivered to all neurons it is connected to, and the potential is reset. Then a new iteration step begins. The step size stipulates how frequently the threshold-crossing inspections occur during the simulation. The choice of the step size trades off the spike-detection accuracy against the simulation speed (Morrison et al., 2007). Such grid-constrained simulations force each spike event to a position on the equispaced temporal grid spanned by the step size and therefore induce artificial synchronization of the network dynamics (Hansel et al., 1998; Shelley and Tao, 2001; Brette, 2007; Morrison et al., 2007; van Elburg and van Ooyen, 2009; Hanuschkin et al., 2010).

Event-driven schemes correct such artificial synchronization. In an event-driven scheme the state of a neuron is updated only when it receives a spike. A central queue of events is maintained and each spike is inserted into this queue with its own time stamp. Upon update a neuron predicts when its next spike will occur in the absence of further input. This preliminary event is inserted into the queue and confirmed if it becomes due or removed when invalidated by further input. Efficient and elegant spike prediction methods have been developed for classes of non-invertible neuron dynamics (Ferscha, 1996; Brette, 2007; D'Haene et al., 2009; van Elburg and van Ooyen, 2009; D'Haene and Schrauwen, 2010). Maintaining a central queue in a distributed simulation is challenging, however, and may compromise the time performance of the simulator (for a detailed review see Hanuschkin et al., 2010).

A hybrid scheme circumvents the shortcomings of time-driven and event-driven schemes by embedding a locally event-driven algorithm for each neuron into a globally time-driven scheme (Morrison et al., 2007). The arrival of a spike at a neuron introduces an additional update and checkpoint besides the global checkpoints. The motion of a given neuron in state space is then propagated from incoming spike to incoming spike and eventually to the end point of the global timestep. If the membrane potential of a neuron is above the threshold at a local or global checkpoint, the precise point of threshold crossing is determined in continuous time, and a spike is emitted. In this scheme spike events carry a floating point offset next to their location on the time grid. Thus, in contrast to an event-driven scheme, the hybrid scheme does not predict future spike times but identifies threshold crossings only retrospectively. Hanuschkin et al. (2010) demonstrate that the retrospective scheme is as accurate as the predictive one but at lower computational costs.

The hybrid scheme still has a loophole, however: spikes can be missed. The reason is that the crossing of the threshold voltage is tested by inspecting whether the membrane potential V of the neuron is above threshold θ at the end of a checkpoint, $t_0 + h$, as in the time-driven scheme:

$$V(t_0 + h) \geq \theta \Rightarrow \text{spike.} \quad (1)$$



See Figure 1 for an illustration of this scenario. However, the evolved membrane potential can be below threshold at the end of a checkpoint and yet have crossed the threshold two, four, $2n$ times between the initial and end checkpoints t_0 and $t_0 + h$. The first crossing constitutes a missed spike. Symbolically,

$$V(t_0 + h) \geq \theta \Rightarrow \text{threshold crossing, but} \\ \text{threshold crossing} \not\Rightarrow V(t_0 + h) \geq \theta \quad (2)$$

so the test $V(t_0 + h) \geq \theta$ is a sufficient but not necessary condition for the occurrence of a threshold crossing during the time interval $[t_0, t_0 + h]$. For future reference we call this test *standard test*.

The question remains whether a globally time-driven scheme can be formulated such that it detects every threshold crossing. Although Hanuschkin et al. (2010) argue that the loss of spikes of the standard test is not of practical relevance in natural parameter regimes, the availability of an absolutely lossless method would free the researcher from inquietude and costly controls when faced with previously unexplored neuron models or network architectures.

In this work, we propose a new spike-detection method based on a necessary and sufficient condition for threshold crossing to occur in a given time interval; we therefore call it *lossless method* or *lossless test*. The method is derived from state-space analysis and works especially well with neuron models with affine or linear subthreshold dynamics; i.e., of the form $\dot{s}(t) = As(t) + q$ or $\dot{s}(t) = As(t)$. It consists of a system of inequalities—some linear, some non-linear in the state-space variables—that together determine whether the initial state of a neuron will or will not reach threshold within the time interval until the next checkpoint. This lossless method is meant to replace the standard test (Equation 1) in the time-driven and the hybrid scheme. Alone it does not solve the problem of artificial synchronization characteristic of time-driven schemes. Hence this method is most

meaningful within a hybrid scheme. Thanks to its perfect spike detection the lossless method can in fact be used to benchmark hybrid schemes based on the standard test; Hanuschkin et al. (2010) use the method of D'Haene et al. (2009) for this purpose.

In section 3 we present the idea behind the lossless method for an affine or linear neuron dynamics, and develop its mathematical construction. Parts of this construction must be addressed on a case-by-case basis; therefore in section 4 we provide a concrete implementation of the lossless method for the leaky integrate-and-fire model with exponential synaptic currents (Fourcaud and Brunel, 2002), within the hybrid scheme. Readers interested only in the implementation are encouraged to proceed directly to section 4.

The method can be algorithmically expressed in different ways. We explore two alternative sequences of inequalities and assess their costs in terms of time-to-completion relative to each other and to the hybrid scheme based on the standard test. For the latter scheme we also assess the number of missed spikes in commonly considered network regimes. The hybrid scheme based on our lossless method delivers the desired exact implementation of the mathematical definition of the neuron model without any further approximation up to floating point precision.

Preliminary results have been published in abstract form (Kunkel et al., 2011; Krishnan et al., 2016). The technology described in the present article will be made available with one of next major releases of the open-source simulation software NEST. The conceptual and algorithmic work described here is a module in the long-term collaborative project to provide the technology for neural systems simulations (Gewaltig and Diesmann, 2007).

2. IDEA: MOVING A SURFACE BACKWARDS INSTEAD OF A POINT FORWARD

Let us summarize the problem mentioned in the previous section. We assume that a neuron's state evolves according to three different dynamical laws or motions in state-space: (a) an integrable subthreshold dynamical law as long as the neuron's membrane potential is below threshold and there are no changes in input currents; (b) discrete jumps in the subthreshold motion at predetermined times, corresponding to incoming spikes or to sudden changes in external currents; these can be formally incorporated into the subthreshold dynamical law via delta functions; (c) a "spike," i.e., an instantaneous jump of the membrane potential from threshold to a reset value, as soon as the potential reaches the threshold value. The jump may be followed by a refractory period in which the membrane potential remains constant at the reset value. Then the integrable motion (a) takes place again.

The advantage of an integrable subthreshold dynamical law is that the state of the neuron at a time $t_0 + h$ can be analytically determined by that at time t_0 ; here h can be negative or positive, and for simplicity we set $t_0 = 0$ in what follows. The evolution can thus be calculated in discrete time steps, in particular in between times at which jumps (b) occur. The spiking component

of the motion, however, forces us to check whether the membrane potential V reached a threshold value θ within the timestep interval $[0, h]$. We call this event *threshold crossing*; by "crossing" we also mean tangency. A sufficient condition for threshold crossing is that the membrane potential be above threshold at the end h of the time step: by continuity, it must have assumed the threshold value at some time in the interval $[0, h]$. But, as mentioned in the introduction, this condition is not necessary: during the time step the potential may touch or surpass the threshold value and then go below it again, as in **Figure 1**, an even number of times. Its value is then below threshold at both ends of the time step, $t = 0$ and $t = h$. A test that only relies on the sufficient condition $V(h) \geq \theta$, like the standard test, can therefore miss some spikes, leading to an incorrect motion in state-space. We need a test based on a necessary and sufficient condition.

The threshold is a hyperplane in state space with equation $V = \theta$, and "threshold crossing" means that the trajectory of the state during $[0, h]$, a curve in state space, intersects this hyperplane. This is equivalent to looking for the roots of the equation $V(t) - \theta = 0$, checking whether this set of roots is empty or complex (no threshold crossing) or non-empty and real (threshold crossing). This idea is illustrated in **Figure 2** for a two-dimensional state space. Such equation is usually transcendental and its roots have to be found numerically (Press et al., 2007, ch. 9). As one of the first steps in this root search, "you should always bracket a root, that is, know that the function changes sign in an identified interval, before trying to converge to the root's value" (Press et al., 2007, § 9.0); and this is what the necessary but insufficient condition $V(h) \geq \theta$ does. When this condition is not met, a necessary and sufficient condition is that the maximum of the voltage $V(t)$ be above threshold for $t \in [0, h]$. This would lead us to maximization algorithms (Press et al., 2007, ch. 10).

We want to approach this problem from a point of view that mathematically is equivalent to checking whether the maximum of the voltage $V(t)$ in the time interval $t \in [0, h]$ is above threshold, but geometrically is very different. Up to now we have imagined to move the initial state of the neuron at time $t = 0$, which is a point in state space, and to check whether this moving point touches the threshold hyperplane before time $t = h$. All the while the threshold hyperplane has been at rest, so to speak. This is like filming a runner rushing toward the finish line, by means of a video camera stationed beside the latter. But the same problem can be looked at from a different frame of reference: we can keep the state of the neuron at rest, and move the whole threshold (i.e., the states that lie on the threshold) instead, for the same amount of time, and check whether this moving hypersurface touches the point. This is like having a video camera fastened on the runner, filming the approaching finish line. As long as the *relative* motion of the two objects is the same, the question of whether they meet has the same answer whether we see the one moving while the other is at rest, or vice versa. The two frames of reference are illustrated in **Figure 2** for a two-dimensional state space.

In the second reference frame, the propagation of the points of the threshold has a dynamical law where time appears with a negative sign (see section 3.2), as if these points were moving backwards in time. For this reason, as a visual aid, we picturesquely call the first point of view "propagation of the state

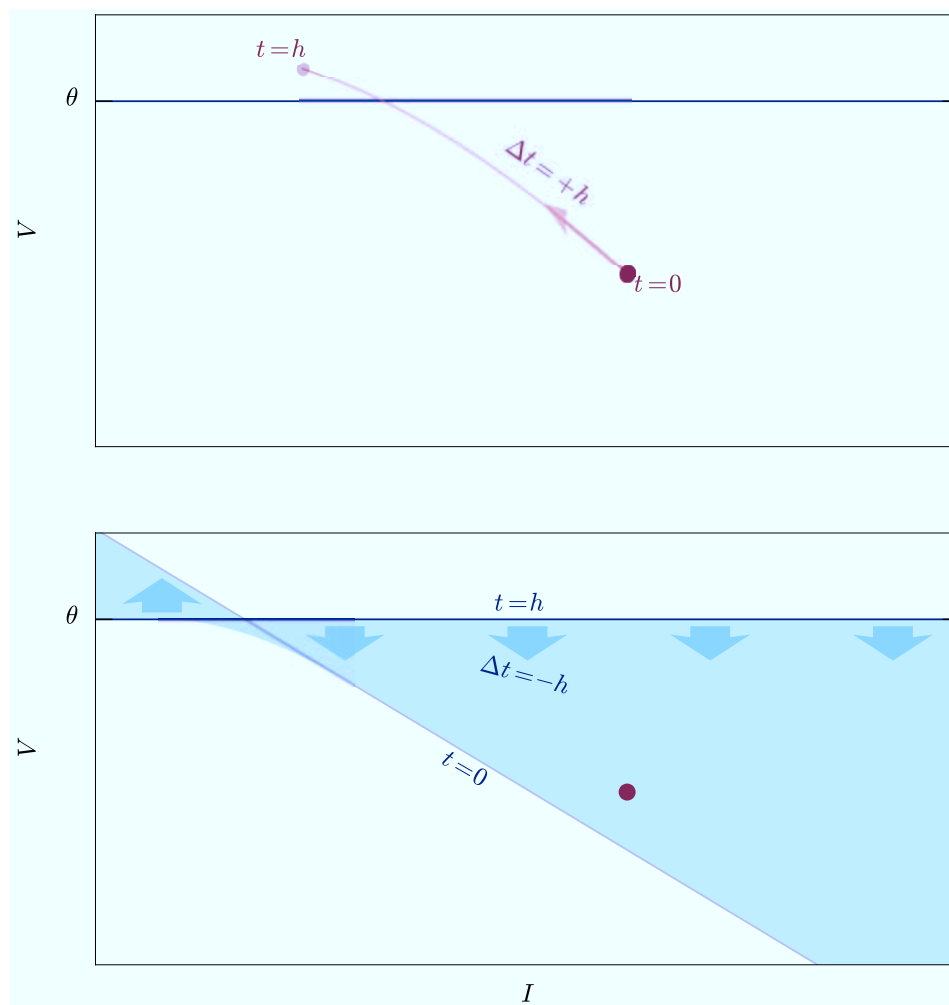


FIGURE 2 | Illustration of two exact methods to check whether an initial state (red dot) crosses the threshold (blue horizontal line) during the evolution from $t = 0$ to $t = h$. **Upper:** The idea of a root-finding method is to evolve the state forwards in time by a step $\Delta t = +h$, and to check whether its trajectory (light-red curve) intersects the threshold. **Lower:** The idea of the lossless method is to evolve the whole threshold “backwards in time” from $t = h$ to $t = 0$ by a step $\Delta t = -h$ and to check whether its trajectory, which is a volume in state-space (light-blue region), contains the initial state, which is kept fixed. The threshold shifts and rotates as it evolves, and the trajectories of its individual points are unknown: this method does not inform us of when and where on the crossing it occurs, and is therefore computationally faster.

forwards in time,” and the second “propagation of the threshold backwards in time,” or just “backpropagation.”

In the first frame of reference we are checking whether a curve (the trajectory of the state) intersects a hypersurface (the threshold). In the second frame of reference we are checking whether a hypervolume (the trajectory of the threshold) intersects a point (the state); in other words we are checking whether a point belongs to a particular state-space region. The test for the intersection of a 1-dimensional curve with an $(N - 1)$ -dimensional surface is replaced by the test for the membership of a point in an N -dimensional volume.

Geometrically the latter test translates into a system of *inequalities* that the initial state at $t = 0$ must satisfy if it does not cross threshold within the time interval $[0, h]$. This means that the maximum of the voltage $V(t)$ in that same interval is below threshold.

The equations corresponding to these inequalities represent the boundary between the set of states that will cross the threshold within the timestep h —which we call *spike region*—and the set of those that won’t—which we call *no-spike region*. Finding these inequalities in explicit form is the most important point of this method, and can be achieved with this heuristic procedure:

- I. Find, in parametric form, the hypervolume swept by the moving threshold. This is done by representing the propagation of the threshold in state-space as a map between two manifolds: the product manifold threshold $\times$ time, and the state-space manifold.
- II. Find the boundary of this hypervolume, in parametric form. This is done by determining the placement of the images of the boundaries of the product manifold and, most important,

of the images of the critical points of the map. The latter are the points at which the map becomes singular, thus mapping N -dimensional regions to $(N-1)$ -dimensional ones (Choquet-Bruhat et al. 1996, § II.A.1; Spivak 1999, ch. 2).

III. Transform the equations of the boundary above from parametric to implicit form. If this is not possible, the boundary can still be approximated by a simplicial mesh, as finely as we please.

At this point we have a set of inequalities that can separate the states that will spike from those that won't.

IV. Finally, optimize the resulting set of inequalities, by e.g., adding to them a preliminary set of linear inequalities that delimit the larger parts of the spike or no-spike regions.

In the following analysis we mathematically develop the first two steps in a general way for any affine neuron dynamics. They can be generalized to other kinds of dynamics; we get back to this point in section 5. The procedure for the last two steps depends on the particular neuron model, so we can only give general guidelines. section 4 provides a concrete example of all steps. An advantage of our algorithm is that its construction needs to be carried out only once for any given neuron model.

Our algorithm tells whether a neuron state will cross the threshold within a time increment h . By construction it does not tell at what time t_θ the crossing, if any, occurs. To determine such time we must call a root-finding algorithm (Press et al., 2007, ch. 9); but our algorithm has already bracketed the root to be found.

We believe that our geometric, state-space point of view can more readily suggest possible optimizations for the threshold-crossing test than a point of view focused on state trajectories and voltage maxima, as we will show in section 4.

3. A TIME-REVERSED STATE-SPACE ANALYSIS

3.1. Mathematical Preliminaries

The final equations to be obtained, Equation (25), can be derived by concepts from vector analysis, Cartesian geometry, and functional analysis; but the derivation is lengthy. To shorten it we use concepts and terminology from affine spaces (Artin, 1955; Coxeter, 1969; Rockafellar, 1972; Nomizu and Sasaki, 1994; Porta Mana, 2011), and differential manifolds (de Rham, 1984; Burke, 1987; Schouten, 1989; Bossavit, 1991, 2002; Dodson and Poston, 1991; Simon et al., 1992; Nomizu and Sasaki, 1994; Burke, 1995; Choquet-Bruhat et al., 1996; Ramanan, 2005; Marsden and Ratiu, 2007).

The state space S of a neuron has a natural vector-space structure (an affine-space structure would also suffice), inherited from the physical quantities that define it: the membrane potential V and other $N-1$ physical quantities I whose exact number and definition depend on the specific neuron model (e.g., I could represent currents or additional voltages, for example of different compartments).

Every hyperplane in the state-space is defined by an affine equation $\mathbf{k}^\top \mathbf{s} = \kappa$, the covector $\mathbf{k}^\top$ being the normal to the

hyperplane, where κ is the affine term. The inequality $\mathbf{k}^\top \mathbf{s} > -\kappa$ defines one of the two half-spaces delimited by the hyperplane. The threshold hyperplane is especially important: it is the set of states $\mathbf{s}$ whose membrane potential has the threshold value: N -dimensional differential manifold: its points $\{\mathbf{s}\}$ are the neuron states, and the quantities (I, V) are coordinates $V: S \rightarrow \mathbf{R}$ and $I: S \rightarrow \mathbf{R}^{N-1}$ e.g., the membrane potential of a state $\mathbf{s}$ is $V(\mathbf{s})$. These coordinates respect the vector structure of the state-space, i.e., $V(\mathbf{s}_1 + \mathbf{s}_2) = V(\mathbf{s}_1) + V(\mathbf{s}_2)$ and likewise for I .

$$V(\mathbf{s}) = \theta. \quad (3)$$

Its equation $\mathbf{k}^\top \mathbf{s} = \kappa$ in coordinates (I, V) has coefficients

$$\mathbf{k}^\top = (\mathbf{0}^\top, 1), \quad \kappa = \theta, \quad (4)$$

$$\mathbf{k}^\top \mathbf{s} = \kappa \quad \text{on threshold}, \quad \mathbf{k}^\top \mathbf{s} < \kappa \quad \text{below threshold}.$$

An affine transformation of the state-space onto itself,

$$\mathbf{s} \mapsto \mathbf{M}\mathbf{s} + \mathbf{m}, \quad (5)$$

where $\mathbf{M}$ is a linear transformation and $\mathbf{m}$ a state, maps each hypersurface and half-space $\mathbf{k}^\top \mathbf{s} \geq \kappa$ to a hypersurface and half-space $\mathbf{k}'^\top \mathbf{s} \geq \kappa'$ with

$$\mathbf{k}'^\top = \mathbf{k}^\top \mathbf{M}^{-1}, \quad \kappa' = \kappa + \mathbf{k}^\top \mathbf{M}^{-1} \mathbf{m} \quad (6)$$

(the transformation of the normal $\mathbf{k}^\top$ shows why it is a covector rather than a vector).

We now show that the integrable dynamical law within a finite time step h generates affine transformations. Consider the evolution equation

$$\dot{\mathbf{s}}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{q}. \quad (7)$$

In a time interval h , an initial state $\mathbf{s}_0$ propagates under this evolution into the final state at time t_0

$$\mathbf{s}(t_0 + h) = e^{h\mathbf{A}}\mathbf{s}(t_0) + (e^{h\mathbf{A}} - 1)\mathbf{A}^{-1}\mathbf{q}, \quad \mathbf{s}(t_0) = \mathbf{s}_0. \quad (8)$$

This, for each h , is an affine transformation of the form (5). In coordinates (I, V) the linear operator $\mathbf{A}$ and vector $\mathbf{q}$ have the block form

$$\mathbf{A} = \begin{pmatrix} \mathbf{B} & d \\ \mathbf{c}^\top & \alpha \end{pmatrix}, \quad \mathbf{q} = \begin{pmatrix} r \\ \beta \end{pmatrix}, \quad (9)$$

where $\mathbf{B}$ is an $(N-1, N-1)$ matrix, α and β numbers, and the dimensionalities of the rectangular matrices $\mathbf{c}^\top$, d follow accordingly.

3.2. Derivation of the Threshold-Crossing Condition

Let us mathematically summarize the first threshold-crossing test discussed in section 2. We said that the state evolution (Equation 8) can be efficiently used in a time-step scheme in numerical simulations, but we need to test whether a threshold crossing occurred at some time $t \in]t_0, t_0 + h]$. A necessary and

sufficient condition would be the existence of solutions of the transcendental equation in t

$$V(e^{t\mathbf{A}}\mathbf{s}(t_0) + (e^{t\mathbf{A}} - 1)\mathbf{A}^{-1}\mathbf{q}) = \theta, \quad t \in [0, h], \quad (10)$$

which corresponds to the intersection of the trajectory (Equation 8) and the threshold hyperplane (Equation 3); but it is a costly condition to test.

We now develop the second kind of threshold-crossing test discussed in section 2, according to the steps I–IV. Steps I and II are performed for any affine dynamics. Steps III and IV have to be solved on a case-by-case basis, so their analysis below is only a guideline; a concrete example on how to perform them is given in section 4.2.3 and 4.2.4.

3.2.1. Hypervolume in Parametric Form

Consider the threshold hyperplane $V(\mathbf{s}) = \theta$ as an $(N - 1)$ -dimensional manifold with coordinates $\mathbf{x}$. It is embedded in state-space via the map

$$\mathbf{x} \mapsto (\mathbf{I}, V) = (\mathbf{x}, \theta). \quad (11)$$

Each state $(\mathbf{x}, \theta)$ on the threshold, when propagated backwards in time for an interval h , traces a curve in state-space (the yellow lines of Figure 3).

The union of these curves is an N -dimensional product manifold, called the *extrusion* (cf. Chap. 5. Bossavit, 2003) of the threshold hyperplane. We can use coordinates $(\mathbf{x}, t) \in (\mathbf{R}^{N-1} \times [0, h])$ on this manifold. Its mapping into state-space is given, with the help of Equation (8), by

$$E: (\mathbf{x}, t) \mapsto e^{-t\mathbf{A}} \begin{pmatrix} \mathbf{x} \\ \theta \end{pmatrix} + (e^{-t\mathbf{A}} - 1)\mathbf{A}^{-1} \begin{pmatrix} \mathbf{r} \\ \beta \end{pmatrix}, \quad (12)$$

where $t > 0$ is the direction of the past. This map is analytic, but generally not an *embedding* because it can have self-intersections. We will see the significance of this in section 3.3. It is not an *immersion* either because it can have singular points; these will be especially important for us because they constitute part of the boundary between spike and no-spike regions. See Figure 3 for a two-dimensional example, and Figure 4 for a three-dimensional one.

For fixed t , the map $\mathbf{x} \mapsto E(\mathbf{x}, t)$ is affine, and its image is a hyperplane representing the states on the set of states at threshold propagated backwards in time for an interval t . Combining it with Equations (4–6) we find that the set of states of backpropagated threshold state has t -dependent normal and affine terms

$$\begin{aligned} \mathbf{k}_t^\top &= (\mathbf{0}^\top, 1) e^{t\mathbf{A}} \\ \kappa_t &= \theta + (\mathbf{0}^\top, 1) (1 - e^{t\mathbf{A}})\mathbf{A}^{-1} \begin{pmatrix} \mathbf{r} \\ \beta \end{pmatrix}. \end{aligned} \quad (13)$$

The inequality $\mathbf{k}_t^\top \mathbf{s} < \kappa_t$ determines the backpropagated half-space, which is below threshold.

3.2.2. Hypervolume Boundary in Parametric Form

We must now find the boundary of the image of the map E . The latter is a closed set, being the image of a closed set under

a continuous map; its boundary must therefore be the image of some points of the domain. Such points must either lie on the boundaries of the domain, $\mathbf{R}^{N-1} \times \{0\}$ and $\mathbf{R}^{N-1} \times \{h\}$, or be critical points of E , or both, because E is differentiable. See the example of Figure 3.

The images of the boundary are easily found from (Equation 12): one (image of $t = 0$) is the threshold hyperplane, the other ($t = h$) is the hyperplane $\mathbf{k}_h^\top \mathbf{s} = \kappa_h$, with coefficients given by Equation(33). Explicitly, in coordinates $(\mathbf{I}, V)$,

$$V = \theta, \quad (14)$$

$$(\mathbf{0}^\top, 1) e^{h\mathbf{A}} \begin{pmatrix} \mathbf{I} \\ V \end{pmatrix} = \theta + (\mathbf{0}^\top, 1) (1 - e^{h\mathbf{A}})\mathbf{A}^{-1} \begin{pmatrix} \mathbf{r} \\ \beta \end{pmatrix}. \quad (15)$$

Let us find the image of the critical points of E . The derivative of E at a point $(\mathbf{x}, t)$ is

$$E'(\mathbf{x}, t) = -e^{-t\mathbf{A}} \begin{pmatrix} \mathbf{I} & \mathbf{B}\mathbf{x} + d\theta + \mathbf{r} \\ \mathbf{0}^\top & \mathbf{c}^\top \mathbf{x} + \alpha\theta + \beta \end{pmatrix}. \quad (16)$$

This is also called the tangent map of E , and denoted E_* , DE , or TE in differential-geometry texts.

Its determinant, the Jacobian determinant of the map, represents the inverse ratio between a volume element at that point and its image in state-space (the sign determines their relative orientation). Hence this ratio vanishes at points where volume elements are mapped onto area elements (in other words, N linearly independent vectors in the domain are mapped onto N linearly dependent vectors), which is a feature of the boundary. See the two-dimensional example of Figure 3.

Let us look for points where $\det E'(\mathbf{x}, t) = 0$. In Equation (16), the determinant of the exponential never vanishes, so we only have to consider the determinant of the matrix on the right. This is easily calculated by Laplace expansion along the last row, whose elements all vanish except the last. The cofactor of the last element is $\det \mathbf{I}$ (modulo a sign). Hence

$$\det E'(\mathbf{x}, t) = 0 \iff \mathbf{c}^\top \mathbf{x} + \alpha\theta + \beta = 0 \quad (17)$$

and the coordinates $(\mathbf{x}, t)$ of critical points satisfy

$$\mathbf{c}^\top \mathbf{x} + \alpha\theta + \beta = 0, \quad 0 \leq t \leq h. \quad (18)$$

This equation says that one of the coordinates $\mathbf{x}$ has an affine dependence on the remaining ones; let us call these $\mathbf{y}$. For example, if $c_{N-1} \neq 0$, the equation above has the parametric solution

$$\begin{aligned} x_1 &= y_1, & \dots, & & x_{N-2} &= y_{N-2}, \\ x_{N-1} &= -\frac{c_1 y_1 + \dots + c_{N-2} y_{N-2} + \beta + \alpha\theta}{c_{N-1}}. \end{aligned} \quad (19)$$

Denote this affine dependence by $\mathbf{x}(\mathbf{y})$. By taking the derivative of Equation (18) with respect to $\mathbf{y}$ we have

$$\mathbf{c}^\top \partial_{\mathbf{y}} \mathbf{x} = \mathbf{0}^\top, \quad (20)$$

a property we will use later.

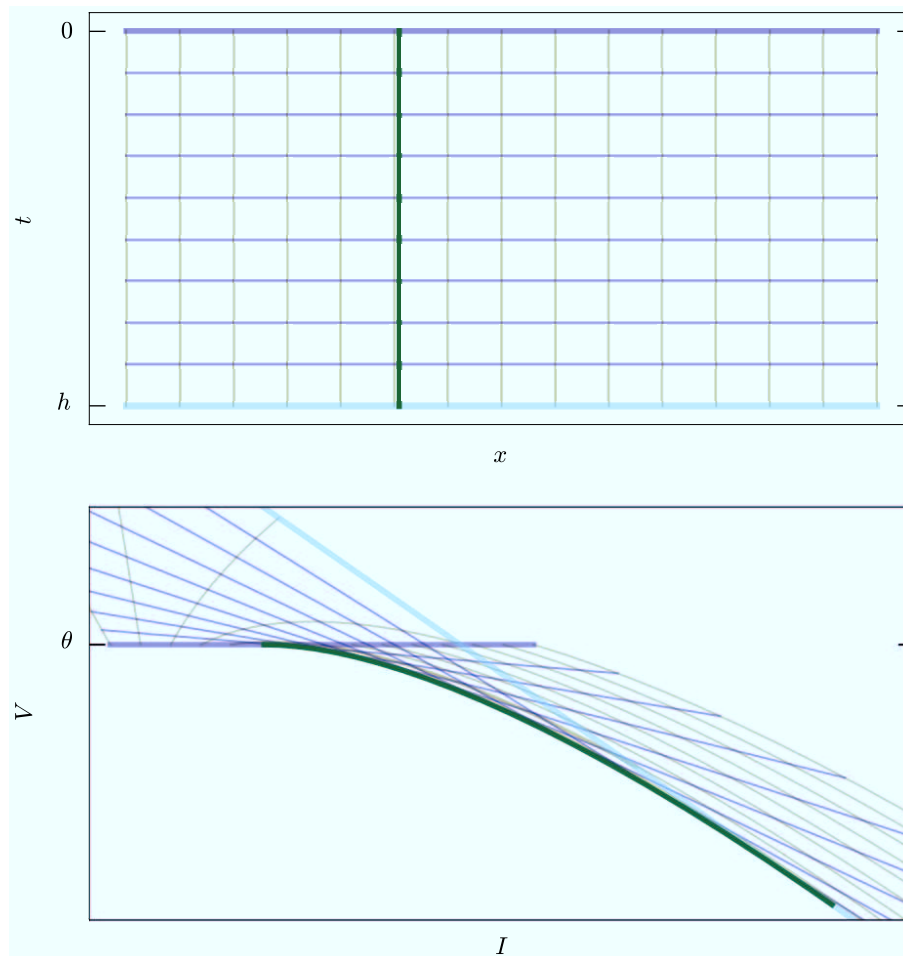


FIGURE 3 | Example of map $E: (x, t) \mapsto (I, V)$, Equation (12), in two dimensions. **Upper:** The abstract manifold with coordinates (x, t) corresponding to the values of current and time to threshold crossing. **Lower:** The image of the map in state space. Thick green curve corresponds to the set of singular points where $\det(TE) = 0$, Equation (10). Yellow lines, constant x , are trajectories of states ending on the threshold. Violet lines, constant t , are snapshots of the threshold moving “backwards in time.” The thicker violet and blue lines correspond to the boundaries $t = 0$ and $t = h$. For $t = 0$ we have $I = x$, $V = \theta$.

The locus of critical points in state-space is then given parametrically by a map Γ , found by substitution of Equations (19) in Equation (12):

$$\Gamma: (y, t) \mapsto e^{-tA} \begin{pmatrix} x(y) \\ \theta \end{pmatrix} + (e^{-tA} - 1)A^{-1} \begin{pmatrix} r \\ \beta \end{pmatrix}, \quad 0 < t < h. \quad (21)$$

This locus has four important features:

1. The points on the locus corresponding to fixed t belong to a hyperplane with coefficients (Equation 33), as is easily checked by substitution. In other words, the locus of critical points is the envelope of the backpropagated threshold hyperplanes $k_t^T s = \kappa_t$, Equation (33), at different times t , and its tangent hyperplanes have normals k_t^T given by Equation (33). In particular, this locus is tangent to the threshold hyperplane.
2. Comparing the dynamical law (Equation 7), the map for the threshold hyperplane (Equation 11), and the critical-point condition (Equation 18), we notice that the latter is also the condition for the trajectory (Equation 8) of a state $s_0 = (x, \theta)$ on the threshold hyperplane to have an extremum in the membrane potential V . Hence, *the locus of critical points is the trajectory of the intersection between the threshold and the V -nullcline.*
3. From the form of Equations (18, 21), the locus of critical points is flat along $N - 2$ dimensions—corresponding to a fixed value of the coordinate t —and is curved normally to the direction t : *it is an $(N - 2)$ -ruled surface.*
4. The dynamical law (Equation 7) preserves affine combinations of solutions—and therefore convex combinations as well. If s_1, s_2 are two arbitrary initial states in the no-spike region, then their propagated states also satisfy $V[s_1(t)] < \theta$ and $V[s_2(t)] < \theta$ when $0 \leq t \leq h$; and also the propagation of their convex combination $\lambda s_1 + (1 - \lambda)s_2$, $0 < \lambda < 1$ satisfies

$$V(\lambda s_1(t) + (1 - \lambda)s_2(t)) < \theta, \quad 0 < \lambda < 1, \quad (22)$$

i.e., it lies in the no-spike region. This proves that the no-spike region is *convex*. This also means that the locus of critical points is a *concave*, i.e., $\cap$ -shaped, surface: the segment between any two of its points lies completely within the no-spike region.

In view of the first property above, let us call the locus of critical points *envelope*, for brevity.

3.2.3. Hypervolume Boundary in Implicit Form or Its Approximation

The next step is the elimination of the parameters (y, t) to express the envelope Γ as one or more implicit equations $\chi(\mathbf{s}) = 0$ defined on domains D_χ for the state-space coordinates $\mathbf{s}$. This step can involve a transcendental equation, and a closed-form solution for it may not exist. We must therefore proceed in two possible ways depending on its existence:

- If an implicit, closed form for the envelope Γ exists, we can find a function χ of the state-space variables such that $\chi(\mathbf{s}) = 0$ is the envelope:

$$\{\mathbf{s} | \chi(\mathbf{s}) = 0, \mathbf{s} \in D_\chi\} = \{\Gamma(y, t) | y \in \mathbf{R}^{N-2}, 0 \leq t \leq h\}. \quad (23)$$

Since χ is determined apart from a multiplicative constant, we can choose this constant in such a way that the inequality $\chi(\mathbf{s}) < 0$, $\mathbf{s} \in D_\chi$, determines the region below the hypersurface contained in the no-spike region.

It is sometimes possible to find an implicit closed-form for the envelope Γ even when a closed-form solution for the intersection of a propagated state and the threshold does not exist. The concrete model discussed in section 4 is such an example.

- If an implicit, closed form for the envelope Γ cannot be found, it is still possible to approximate the latter by a mesh of $(N-1)$ -dimensional simplices (i.e., triangles, tetrahedra, and so on), to arbitrary precision. The simplicial mesh should extend from the $(N-2)$ -dimensional curve where the envelope is tangent to the threshold, Equation (14), to the one where it is tangent to the backpropagated threshold at $t = h$, Equation (15). In this case instead of an inequality $\chi(\mathbf{s}) < 0$ we have a system of linear inequalities.

Exact spike detection can still be achieved in this case. Owing to the concavity of the envelope discussed in the previous section, all the simplices of the mesh lie in the no-spike region. Therefore, if the system of linear inequalities is satisfied the state surely lies in the no-spike region. If the system is not satisfied, a spike can be assumed to occur, and the threshold-crossing time can be sought by a root-finding algorithm. In this case there still is a small chance that the state lies in one of the small no-spike pockets between the envelope and the mesh, but the absence of a spike would still be detected by noting that the root-finding algorithm has no solution in this case.

The coordinates of the vertices of the mesh are obtained via Equation (21) by selecting a set of points $\{(y_i, t_i)\}$, $0 < t_i < h$. Since the envelope is a ruled surface, as shown in section 3.2.2,

the mesh can be conveniently chosen in such a way that one face of each simplex fully lies on the envelope. The coefficients of the linear inequalities related to the mesh therefore do not depend on the state of the neuron, but depend on the parameters of the model and, in exponential form, on the timestep h . Their functional form needs to be determined *only once* for a given model.

Although less efficient than the case in which an implicit closed-form for the envelope exists, the approximation with a simplicial mesh can still be quite effective, because not all the linear inequalities of the linear system will have to be checked in general: as soon as one of them is not satisfied the test will stop. The most efficient coarseness of the mesh has to be evaluated case-by-case for each model.

Whether an implicit closed form for the envelope or only an approximating mesh is available, the spike test can in any case be optimized by first testing a set of linear inequalities that delimit the larger parts of the spike and no-spike regions. Such set would be a very coarse mesh approximating the envelope. A two-dimensional example is given in **Figure 6**, and we discuss this with a concrete example in section 4.4.

In the following we denote by $\chi(\mathbf{s}) < 0$ either the closed form of the inequality, if it exists, or the system of linear inequalities that approximate it, if the former does not exist in closed form.

3.2.4. Boundary Intersections and Final System

If a state does not cross the threshold hyperplane at any time $t \in [0, h]$, then it must, by definition, belong for every t to the image of the half-space that lies below threshold, when this half-space is backpropagated by the time t . Mathematically, this is simply the statement of the equivalence

$$\mathbf{k}^T(\mathbf{M}\mathbf{s} + \mathbf{m}) < \kappa \Leftrightarrow (\mathbf{k}^T\mathbf{M})\mathbf{s} < (\kappa - \mathbf{k}^T\mathbf{m}) \quad (24)$$

established in section 3.1, where $\mathbf{M}$ and $\mathbf{m}$ are the coefficients of the affine evolution by time t .

By construction in the previous subsection, the inequality $\chi(\mathbf{I}, V) < 0$ defines the region of intersection of all such backpropagated half-spaces, bounded by the envelope $\chi(\mathbf{I}, V) = 0$. We just need to join to it the condition for the boundaries corresponding to the times $t = 0$ and $t = h$, $V < \theta$, $\mathbf{k}_h^T\mathbf{s} < \kappa_h$, discussed in section 3.2.1.

The states that do not cross the threshold during the time interval $[0, h]$ belong therefore to the no-spike region defined, in coordinates, by

$$\mathbf{s}_0 = (\mathbf{I}, V) \in \text{no-spike region} \Leftrightarrow \begin{aligned} &V < \theta \\ &\text{and} \end{aligned} \quad (25a)$$

$$(\mathbf{0}^T, 1) e^{h\mathbf{A}} \begin{pmatrix} \mathbf{I} \\ V \end{pmatrix} < \theta + (\mathbf{0}^T, 1) (1 - e^{h\mathbf{A}}) \mathbf{A}^{-1} \begin{pmatrix} \mathbf{r} \\ \beta \end{pmatrix} \quad (25b)$$

and

$$\chi(I, V) < 0, \quad (I, V) \in D_\chi. \quad (25c)$$

Some inequalities in this system may turn out to be redundant, i.e., automatically satisfied if the remaining ones are, and can thus be dropped. Section 4.2.3 illustrates such a redundancy.

3.3. The Region of Missed Spikes

In the previous section we mentioned that the N -dimensional product manifold formed by the $(N - 1)$ -dimensional threshold hyperplane and the time interval $[0, h]$ presents self-intersections when mapped to the state-space. Two-dimensional examples of such a region are shown in **Figure 3**, lower panel, and in **Figure 6**, the region called S_2 .

A state $\mathbf{s}$ in the self-intersection region corresponds to two or more different coordinates $(\mathbf{x}, t)$:

$$\mathbf{s} = E(\mathbf{x}, t) = E(\mathbf{x}', t'), \quad t \neq t'. \quad (26)$$

Note that $\mathbf{x} \neq \mathbf{x}', t = t'$ is impossible for an affine transformation, since the first column of Equation (16) can never vanish. The condition above simply means that $\mathbf{s}$ crosses the threshold at $\mathbf{x}$ after an interval t and at $\mathbf{x}'$ after an interval t' . By continuity of the trajectory of Equation (7), one of the two must be a crossing from above, and one from below: this is exactly the scenario of double threshold crossing illustrated in **Figure 1**.

Suppose we have a probability distribution for the initial states, for example one that is invariant under the dynamical law (Equation 7). The probability of the self-intersection region $P(X)$ is the probability that the initial state will lead to a double-crossing of the threshold, and therefore be missed. This fact will be used in section 4.4.2 to estimate the number of spikes missed.

4. IMPLEMENTATION EXAMPLE: LEAKY INTEGRATE-AND-FIRE NEURON WITH EXPONENTIALLY DECAYING POST-SYNAPTIC CURRENTS

4.1. The Example Model: Terms in Block Form

In the previous section we mathematically developed the idea of propagating the threshold backward in time in order to check whether a threshold-crossing occurs in a time-stepped dynamical process. The derivation, valid for an affine subthreshold dynamical law, is general and therefore also quite abstract; moreover, it involves a couple of mathematical steps (III and IV in section 2) for which no general formula can be given.

To explain the idea in more concreteness and to give an example of how to face all its steps, we now apply the scheme to a simple but relevant model with a 2-dimensional state-space: the leaky integrate-and-fire neuron with exponentially decaying post-synaptic currents. This model satisfies a homogeneous linear dynamical law on a 3-dimensional state-space (Rotter and

Diesmann, 1999), where the third coordinate is the input current. If this current is constant, the dynamical law can be rewritten as a 2-dimensional affine one.

Leaky integrate-and-fire models, despite their simplicity, approximate the behavior of real neurons with high accuracy (Rauch et al., 2003). The model with exponential synaptic currents captures important properties of real neurons: The post-synaptic potential has a finite rise and decay time and the membrane potential is a continuous function of time. Continuity avoids artificial synchronization, present in simpler models. Moreover, the model is to some extent analytically tractable. For short synaptic time constants, the mean firing rate (Fourcaud and Brunel, 2002) as well as the linear response to small inputs (Schucker et al., 2015) can be obtained analytically.

The example model has a 2-dimensional state-space for a single neuron, defined by the post-synaptic current I and the membrane potential V , which are also our coordinates. Its subthreshold interspike dynamical law (a) in section 3 is affine:

$$\begin{aligned} \dot{I}(t) &= -\frac{1}{\tau_s} I(t), \\ \dot{V}(t) &= \frac{1}{C} I(t) - \frac{1}{\tau} V(t) + \frac{1}{C} I_e, \end{aligned} \quad (27a)$$

or in matrix form

$$\frac{d}{dt} \begin{pmatrix} I \\ V \end{pmatrix} = \begin{pmatrix} -\frac{1}{\tau_s} & 0 \\ \frac{1}{C} & -\frac{1}{\tau} \end{pmatrix} \begin{pmatrix} I \\ V \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{C} I_e \end{pmatrix}, \quad (27b)$$

where C is the membrane capacitance, τ is the membrane time constant, I is the synaptic input current, and I_e is the external input current. The membrane potential V is subject to dissipation with time constant τ and integrates the post-synaptic current I . The latter decays exponentially with time constant τ_s . Typical values of the parameters are $\tau = 10$ ms, $C = 250$ pF, $\tau_s = 2$ ms, and the threshold $\theta = 20$ mV.

Incoming spikes are incorporated in the equation for the current as Dirac deltas, and the external current I_e has jump discontinuities in time. In the time-stepped evolution, such discontinuous events are implemented as instantaneous changes in the initial state $\mathbf{s}_0$ at each timestep. Hence we do not need to consider them explicitly in the equations above (Rotter and Diesmann, 1999).

In terms of the block form of section 3.1 we have

$$\begin{aligned} \mathbf{B} &= \begin{pmatrix} -\frac{1}{\tau_s} \end{pmatrix}, \quad \mathbf{d} = (0), \quad \mathbf{r} = (0), \\ \mathbf{c}^\top &= \begin{pmatrix} \frac{1}{C} \end{pmatrix}, \quad \alpha = -1/\tau, \quad \beta = \frac{1}{C} I_e. \end{aligned} \quad (28)$$

The exponential of $\mathbf{A}$ is

$$\exp(-t\mathbf{A}) = \begin{pmatrix} e^{-\frac{t}{\tau_s}} & 0 \\ \frac{(e^{-\frac{t}{\tau}} - e^{-\frac{t}{\tau_s}})\tau\tau_s}{C(\tau - \tau_s)} & e^{-\frac{t}{\tau}} \end{pmatrix}, \quad (29)$$

which determines the evolution of the neuron state $\mathbf{s}_0$ by an affine map as in Equation (8).

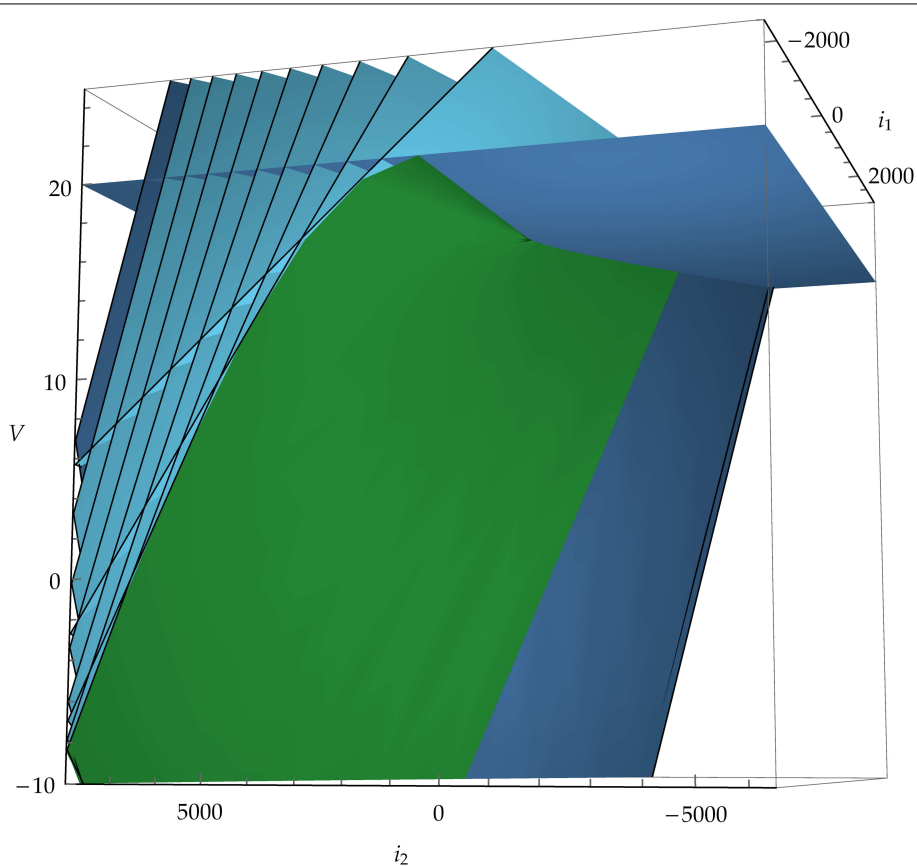


FIGURE 4 | Example of the image of the map $E: (x, t) \mapsto (I, V)$, Equation (12), in a three-dimensional state space. Blue planes are snapshots of the threshold plane (horizontal, darker blue plane) moving “backwards in time.” The green, curved surface corresponds to the set of singular points where $\det E' = 0$, Equation (10). The neuron model in this example is leaky integrate-and-fire with α -shaped post-synaptic currents i_1, i_2 and membrane potential V .

Since the state space is 2-dimensional, the “hyperplanes” and “hypersurfaces” of section 3 are straight lines and curves. In particular, the threshold hyperplane is a line; when propagated by a time t it maps onto a line with covector and affine term given by Equation (33), explicitly

$$\begin{aligned} k_t^T &= \left(\frac{(e^{-\frac{t}{\tau}} - e^{-\frac{t}{\tau_s}}) \tau \tau_s}{C(\tau - \tau_s)} e^{-\frac{t}{\tau}}, \right. \\ \kappa_t &= \theta - \frac{I_e \tau (1 - e^{-\frac{t}{\tau}})}{C}. \end{aligned} \quad (30)$$

For this model, testing for threshold-crossing by checking the intersection of a propagated state and the threshold line means finding the roots of this equation in t , given the initial state $s_0 = (I, V)$:

$$e^{-\frac{t}{\tau}} V + \frac{I_e \tau}{C} (1 - e^{-\frac{t}{\tau}}) + I \frac{\tau \tau_s}{C} \frac{e^{-\frac{t}{\tau}} - e^{-\frac{t}{\tau_s}}}{\tau - \tau_s} = \theta, \quad 0 \leq t \leq h, \quad (31)$$

for which we cannot find a solution in analytic form.

4.2. The Threshold-Crossing Condition

4.2.1. Hypervolume in Parametric Form

The product manifold “threshold line $\times$ time interval” is in this case 2-dimensional, with coordinates (x, t) . Here x is the value of the current at threshold crossing and t the corresponding time point. Its mapping E , Equation (12), to the state-space is

$$\begin{aligned} E: (x, t) \mapsto (I, V) &= \left(e^{-\frac{t}{\tau_s}} x, \quad e^{-\frac{t}{\tau}} \theta + \frac{I_e \tau}{C} (1 - e^{-\frac{t}{\tau}}) \right. \\ &\quad \left. + x \frac{\tau \tau_s}{C} \frac{e^{-\frac{t}{\tau}} - e^{-\frac{t}{\tau_s}}}{\tau - \tau_s} \right). \end{aligned} \quad (32)$$

This map is shown in **Figure 3**: the x isocurves are yellow and in the lower panel they represent the trajectories of states terminating on the threshold; the t isolines are blue and represent the threshold line propagated at different times. Each area element $dx \wedge dt$ in the domain—the small rectangles in the upper plane—is mapped into an area element $dI \wedge dV$ in the image. Note how these area elements are rotated and sheared. The thicker green curve is the set of singular points where the Jacobian determinant, i.e., the determinant of the derivative, vanishes: $\det(E') = 0$. Such points are singular because around them the

images of the area elements get flattened by one dimension. In section 4.3 we discuss the region of self-intersection bounded by the thick light cyan line, the thick dark blue line, and the thick green curved line.

For fixed t , the map $\mathbf{x} \mapsto E(\mathbf{x}, t)$ is affine, and its image is a hyperplane representing the states on the set of states at threshold propagated backwards in time for an interval t . Combining it with Equations (4–6) we find that the set of states of backpropagated threshold state has t -dependent normal and affine terms

$$\begin{aligned} \mathbf{k}_t^T &= (\mathbf{0}^T, 1) e^{t\mathbf{A}} \\ \kappa_t &= \theta + (\mathbf{0}^T, 1) (1 - e^{t\mathbf{A}}) \mathbf{A}^{-1} \begin{pmatrix} \mathbf{r} \\ \beta \end{pmatrix}. \end{aligned} \quad (33)$$

The inequality $\mathbf{k}_t^T \mathbf{s} < \kappa_t$ determines the backpropagated half-space, which is below threshold. An example of the map E for a model with a three-dimensional state space is shown in **Figure 4**.

4.2.2. Hypervolume Boundary in Parametric Form

The boundaries of the image of the map E must be, as explained in section 3.2.3, a subset of the images of the boundaries of the domain, $\mathbf{R} \times \{0\}$ and $\mathbf{R} \times \{h\}$, and the envelope.

The images of the boundaries, with general Equations (14, 15), in terms of (I, V) follow

$$V = \theta, \quad (34)$$

$$V = e^{\frac{h}{\tau}} \theta + \frac{I_e \tau}{C} (1 - e^{\frac{h}{\tau}}) + \frac{I \tau}{C} \tau_s e^{-\frac{h}{\tau_s}} \frac{e^{\frac{h}{\tau}} - e^{\frac{h}{\tau_s}}}{\tau - \tau_s}. \quad (35)$$

The set of critical points of the map E is in this case a 1-dimensional curve, given in parametric form by

$$\mathbf{r} : t \mapsto \left(e^{\frac{t}{\tau_s}} \left(\frac{\theta C}{\tau} - I_e \right), \frac{\tau e^{\frac{t}{\tau}} - \tau_s e^{\frac{t}{\tau_s}}}{\tau - \tau_s} \frac{\tau}{C} \left(\frac{\theta C}{\tau} - I_e \right) + \frac{\tau}{C} I_e \right); \quad (36)$$

the coordinate $\mathbf{y}$ of the general form (Equation 21) do not exist in this case, because the threshold is 1-dimensional. **Figure 3** visualizes the set by the green curve.

4.2.3. Hypervolume Boundary in Implicit Form

The next step in our procedure is to convert the parametric Equation (36) of the envelope into an explicit or implicit equation for the coordinates (I, V) . As section 3.2.3 does not provide a general algorithm, below we illustrate the process using our example model.

By equating the first component of Equation (36) to the coordinate I and solving for t , we find

$$t = -\tau_s \ln \left[\left(\frac{\theta C}{\tau} - I_e \right) / I \right], \quad (37a)$$

subject to the condition for I

$$e^{-\frac{h}{\tau_s}} \leq \frac{1}{I} \left(\frac{\theta C}{\tau} - I_e \right) \leq 1, \quad (37b)$$

required for $0 \leq t \leq h$ and a real logarithm.

Substituting Equation (37a) into the V coordinate of Equation (36) we find

$$V = \frac{\tau I_e}{C} + \frac{\tau I}{C} \frac{\tau \left[\left(\frac{\theta C}{\tau} - I_e \right) / I \right]^{1 - \frac{\tau_s}{\tau}} - \tau_s}{\tau - \tau_s} \quad (38)$$

subject to the condition (Equation 37b). Let us analyse this equation, in view of its extension to an inequality of the form $\chi(I, V) < 0$ as required in section 3.2.3. First, we observe that

$$\frac{\tau e^{\frac{t}{\tau}} - \tau_s e^{\frac{t}{\tau_s}}}{\tau - \tau_s} \leq 1 \quad \text{for } 0 \leq t \leq h. \quad (39)$$

The inequality can be proven by studying the derivative of the fraction with respect to t . The derivative is always negative in the range above and the only maximum of the fraction is the value unity assumed at $t = 0$.

Inspection of Equation (38) and of its parametric form (Equation 36) shows that we must consider three cases: $I_e \leq \theta C / \tau$, i.e., whether the external current is smaller or larger than the *rheobase current*

$$I_\theta = \theta C / \tau; \quad (40)$$

this is the current necessary to reach threshold in an infinite time starting from any state with $I \leq 0$.

- If $I_e < I_\theta$, then I is restricted to

$$0 < I_\theta - I_e \leq I \leq e^{\frac{h}{\tau_s}} (I_\theta - I_e). \quad (41)$$

In this case, using the inequality (Equation 39), the V component of the envelope (Equation 36) is always smaller than the threshold:

$$V \equiv \frac{\tau I_e}{C} + \frac{\tau I}{C} \frac{\tau \left[(I_\theta - I_e) / I \right]^{1 - \frac{\tau_s}{\tau}} - \tau_s}{\tau - \tau_s} \leq \theta \quad \text{when } I_e < I_\theta. \quad (42)$$

- If $I_e > I_\theta$, then I must be negative and restricted to

$$e^{\frac{h}{\tau_s}} (I_\theta - I_e) \leq I \leq I_\theta - I_e < 0. \quad (43)$$

In this case, using the inequality (Equation 39), the V component of the envelope (Equation 36) is always larger than the threshold:

$$V \equiv \frac{\tau I_e}{C} + \frac{\tau I}{C} \frac{\tau \left[(I_\theta - I_e) / I \right]^{1 - \frac{\tau_s}{\tau}} - \tau_s}{\tau - \tau_s} \geq \theta \quad \text{when } I_e > I_\theta. \quad (44)$$

- If $I_e = I_\theta$, the envelope degenerates to a point: $\mathbf{r}(t) = (0, \theta)$, which is the limit point reached in infinite time from any initial point in the no-spike region; this is the geometric interpretation of the equality of external and rheobase currents.

The function representing the envelope, Equation (23) of section 3.2.3, has in this case the explicit form

$$\begin{aligned} \chi(I, V) &:= V - \frac{\tau I_e}{C} - \frac{\tau I}{C} \frac{\tau \left[(I_\theta - I_e) / I \right]^{1 - \frac{\tau_s}{\tau}} - \tau_s}{\tau - \tau_s}, \\ &\text{with } (I, V) \in D_\chi := \left[I_\theta - I_e, e^{\frac{h}{\tau_s}} (I_\theta - I_e) \right] \times \mathbf{R}. \end{aligned} \quad (45)$$

If we calculate its differential, as discussed in section 3.2.3, we find that the latter is a positive multiple of the differential of the backpropagated threshold for each t . This is true in each of the three cases above. Consequently, the inequality $\chi(I, V) < 0$, $(I, V) \in D_\chi$ always includes the no-spike region.

In summary, we arrive at an analytic, implicit equation for the curved boundary (Equation 45). The expression is a transcendental function in I , owing to the generally irrational exponent $1 - \tau_s/\tau$, but it is used in an inequality, hence we do not need to find its roots.

4.2.4. Boundary Intersections and Final System

We can now assemble the system of inequalities defining the no-spike region consisting of the boundaries (Equations 34, 35), and the envelope (Equation 45).

With some rearrangements, simplifications, and the introduction of two new functions f , b , the condition reads:

$$\mathbf{s}_0 = (I, V) \in \text{no-spike region} \Leftrightarrow$$

$$V < \theta \quad (46a)$$

and

$$V < f_{I_e}(I) := e^{\frac{h}{\tau}}\theta + \frac{\tau I_e}{C} \left(1 - e^{\frac{h}{\tau}}\right) + \tau_s e^{-\frac{h}{\tau_s}} \frac{\tau I e^{\frac{h}{\tau}} - e^{\frac{h}{\tau_s}}}{\tau - \tau_s} \quad (46b)$$

and

$$V < b_{I_e}(I) := \frac{\tau I_e}{C} + \frac{\tau I \tau [(I_\theta - I_e)/I]^{1-\frac{\tau_s}{\tau}} - \tau_s}{\tau - \tau_s} \quad \text{if } I \in [I_\theta - I_e, e^{\frac{h}{\tau_s}}(I_\theta - I_e)] \quad (46c)$$

Figure 5 illustrates the system for the two cases $I_e < I_\theta$ and $I_e > I_\theta$. In the former case all three inequalities are necessary; in the latter, as well as for $I_e = I_\theta$, the last inequality is automatically enforced by the first because its right-hand side is larger than the threshold θ [see Equation (44)].

4.3. The Region of Missed Spikes

In Figure 3, lower panel, the trajectories of several states during a timestep h and ending on the threshold line are represented by yellow curves. In that figure we can identify a region where such trajectories self-intersect: it is bounded by a segment of the thick light blue line, a segment of the thick dark blue line, and a portion of the thick green curved line. Trajectories with initial states in this region must therefore cross the threshold twice during the interval $[t_0, t_0 + h]$. As explained in section 3.3, their threshold-crossing is not detected by the sole condition $V[s(t+h)] \geq \theta$. All

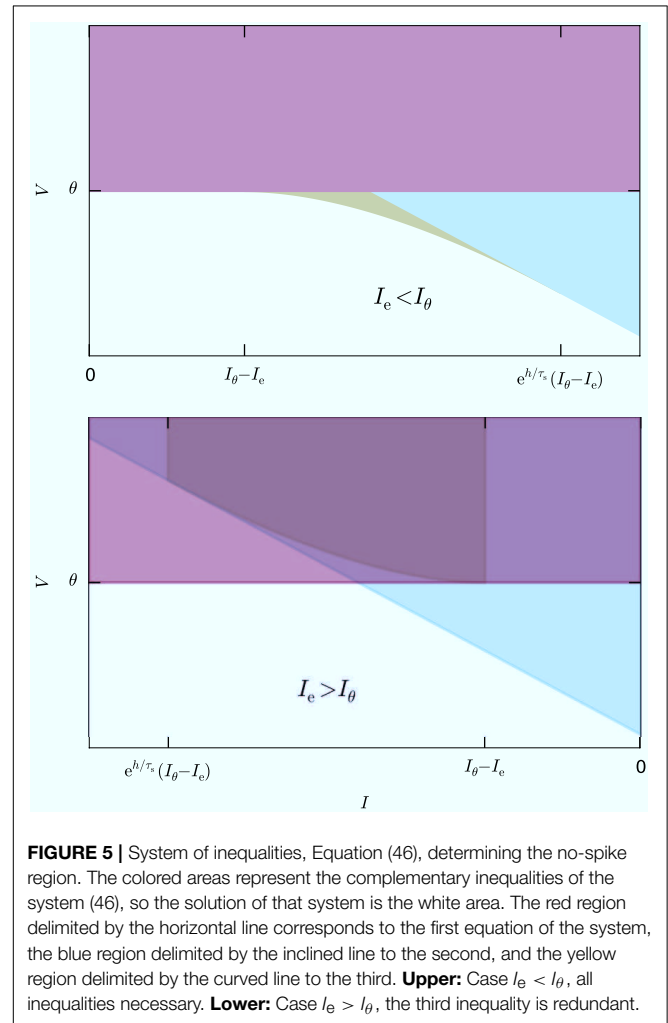


FIGURE 5 | System of inequalities, Equation (46), determining the no-spike region. The colored areas represent the complementary inequalities of the system (46), so the solution of that system is the white area. The red region delimited by the horizontal line corresponds to the first equation of the system, the blue region delimited by the inclined line to the second, and the yellow region delimited by the curved line to the third. **Upper:** Case $I_e < I_\theta$, all inequalities necessary. **Lower:** Case $I_e > I_\theta$, the third inequality is redundant.

states in this region thus generate spikes that are missed by the standard test (Equation 1).

This region is crucial for the comparison of the performances of schemes implementing the present lossless method and schemes relying on the standard test (Equation 1). This comparison is quantitatively made in section 4.4.3.

4.4. Optimization, Time Performance, and Accuracy

4.4.1. Numerical Implementation and Optimization

In the last section we arrived at the system of three inequalities (Equation 46) that determines whether or not the current state (V, I) will cross the threshold within the timestep h . The state will cross the threshold if the system is not satisfied, and it will not cross the threshold if the system is satisfied. This system of inequalities constitutes the lossless method in the present model. The system requires the current timestep h , state (V, I) , and external electric current I_e as inputs (cf. section 3). The timestep h is the minimum between the global timestep and the time interval up to the next input coming from other neurons, hence it can

differ every time the test is called. The external electric current I_e may also vary, stepwise, during the simulation, hence it may also be distinct at each call of the test.

If the system of inequalities is satisfied, thus predicting the absence of a spike within a timestep h , then the state of the neuron is evolved by applying the propagator (Equation 8) with Equations (28, 29), leading to a new state, and the procedure starts again. If the system is not satisfied, thus predicting the occurrence of a spike within h , it is then necessary to compute the time t_θ at which the threshold is crossed. This calculation, explained in Appendix section A.1, is made by interpolation between the current state and time, and the state and time $t_{\max}$ at which the membrane potential would reach its maximum if allowed to increase above threshold. Once t_θ is calculated, a spike is emitted, communicated to the post-synaptic neurons, and the membrane potential is reset to V_{reset} for a refractory period. When this refractory period is over the procedure starts again.

In the evolution loop just described, the membrane potential is reset to a value below threshold as soon as it crosses the latter. Thus no initial state can have $V \geq \theta$. This means that the first inequality (Equation 46a) in the system is always satisfied and can be dropped. Only inequalities (Equations 46b,c) have to be assessed, leading to the reduced system

$$\begin{cases} V < f_{h,I_e}(I), \\ V < b_{I_e}(I). \end{cases} \quad (47)$$

We first discuss the case $I_e < I_\theta \equiv \theta C/\tau$.

The geometric meaning of the inequalities above is illustrated in **Figure 6**. The figure shows four regions: NS_1 , NS_2 , S_1 , S_2 . The no-spike region is the union of NS_1 and NS_2 , the spike region the union of S_1 and S_2 . In the figure they are separated by a thick line, partly curved and black, partly straight and blue. Subregion S_2 is particularly important: it is the region of missed spikes discussed in section 4.3, corresponding to the self-intersection region of **Figure 3**, lower panel. It contains those states that lead to spikes missed by schemes that rely on the standard test (Equation 1).

Region S_1 is separated from S_2 by a dashed blue line, and from NS_1 by a continuous blue line, the continuation of the dashed one. This partly dashed, partly continuous blue line corresponds to the equation $V = f_{h,I_e}(I)$. Hence if the inequality $V < f_{h,I_e}(I)$ is not satisfied then the initial state is in region S_2 or on its blue boundary, and there will be a spike. If the inequality is satisfied the state could be in S_2 – spike – or $NS_1 \cup NS_2$ – no spike; an undetermined case. This inequality requires modest computational costs because it is linear in I and I_e and involves exponentials of h . Regions S_2 and NS_2 are separated by a black curve: this is the envelope, corresponding to the equation $V = b_{I_e}(I)$ for $I \in [I_\theta - I_e, e^{\frac{h}{\tau C}}(I_\theta - I_e)]$. Hence if the inequality $V < b_{I_e}(I)$ is not satisfied the initial state is in S_2 or on its boundary, and there will be a spike. If the inequality is satisfied the initial state is either in NS_2 , or in NS_1 with $I_\theta - I_e < I < e^{\frac{h}{\tau C}}(I_\theta - I_e)$, and no spike will occur.

The computationally most expensive inequality is $V < b_{I_e}(I)$ because it involves irrational powers of I and I_e . It is advisable to

avoid its direct computation as often as possible by pre-testing a linear inequality. In section 3.2.3 we discussed how such a pre-test is indeed possible thanks to the convexity of the no-spike region. There, we argued that the curved envelope can be approximated by a simplicial mesh, which simply reduce to one straight segment in the present two-dimensional case: this is the dot-dashed red line in **Figure 6**, separating NS_1 and NS_2 . This line has equation $V = g_{h,I_e}(I)$ with

$$g_{h,I_e}(I) := \theta + \frac{\tau_s e^{\frac{h}{\tau C}}}{\tau - \tau_s} \frac{\tau}{C} I + e^{\frac{h}{\tau C}} \frac{\tau}{C} (I_\theta - I_e), \quad (48)$$

and the corresponding inequality

$$V < g_{h,I_e}(I) \quad (49)$$

has the same computational costs as $V < f_{h,I_e}(I)$.

If the auxiliary inequality $V < g_{h,I_e}(I)$ is satisfied, the initial state is in NS_1 and $V < b_{I_e}(I)$ is also satisfied. It is therefore convenient to test the auxiliary inequality before the computationally costly one, which can be discarded if the test is positive. **Figure 6** suggests that this test might be positive for the majority of initial states because region NS_1 is much wider than NS_2 . This possibility would be very advantageous, but we now argue that it should be verified by a dynamical analysis.

The system (47) can be translated into a computational algorithm in several different ways, depending on the order of evaluation of its two inequalities and of the auxiliary inequality (49). In simplified terms, such an algorithm consists in a sequence of tests—variously implemented as *if*, *and*, or *or* constructs—for finding the initial state in space-time regions R_1 , R_2 , and so on. The order of these tests is important. The average time cost of the algorithm in a long simulation is given by $\sum_i p_i c_i$, where p_i is the frequency with which states are found in region R_i , which we call “occupation frequency,” and c_i is the cumulative time cost of the test for region R_i . This time cost c_i is cumulative in the sense that all tests up to the $(i - 1)$ th must have been performed, with *false* outcomes, to arrive at the test for R_i . The efficiency of an algorithm therefore depends on the mathematical form of the inequalities defining a region and on the occupation frequencies of the regions, determined by the dynamics. These two factors can be extrapolated by a theoretical analysis, or more practically measured by running long test simulations with typical network setups corresponding to the cases one is interested in.

We now try to determine the most efficient algorithm for the present case. Region S_1 is the least costly, because bounded by one line and therefore involving one inequality linear in I ; then region NS_1 , bounded by two lines involving two linear inequalities; and finally regions NS_2 and S_2 , bounded by the curve that involves rational exponentiation. For this example model, we tried different orderings but show here only two possible extreme cases to illustrate that there is no significant difference in the computational cost.

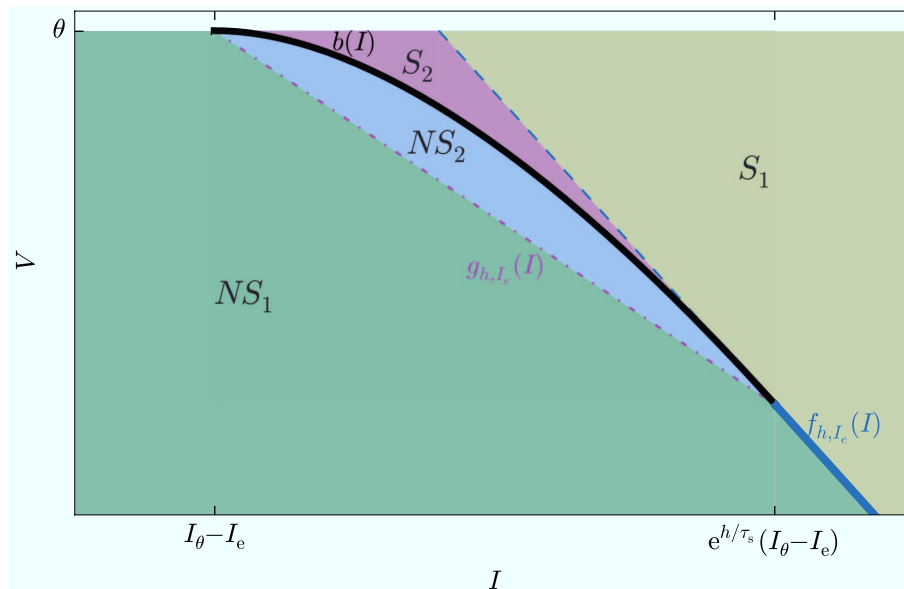


FIGURE 6 | State-space subregions formed by the intersections of the reduced inequalities (Equation 47): $V < f_{h,I_e}(I)$ (straight blue line, partly dashed partly continuous) and $V < b_h(I)$ (black curve), and by the auxiliary inequality (Equation 49): $V < g_{h,I_e}(I)$ (dot-dashed straight purple line). The spike region is $S_1 \cup S_2$, the no-spike region is $NS_1 \cup NS_2$. The subregion S_2 contains all states that emit spikes undetected by the standard test (Equation 1); they are detected by the lossless method.

Algorithm 1

```

bool is_spike( $h$ ):
  pre-compute  $g_{h,I_e}(I)$ 
  if  $V \leq f_{h,I_e}(I)$  and  $V < g_{h,I_e}(I)$  then
    return false
  else if  $V \geq g_{h,I_e}(I)$  then
    return true
  else if  $V \geq b_{I_e}(I)$  then
    return true
  else
    return false

```

Algorithm 1 is based on the assumption that the occupation frequency of a subregion is proportional to that subregion's relative size. If we check the two largest first, in the order NS_1 , S_1 , S_2 , NS_2 , we are therefore more likely to exit the test in its first if branches. The value of g_{h,I_e} is used in two if branches, so it is computed just once and saved before the if sequence in order to save some computations.

Algorithm 2 uses a composite or-and condition rather than several ifs. Assuming left-to-right evaluation, the algorithm corresponds to testing first S_1 (left side of or), then S_2 (right side of or), either case leading to a spike. If neither is true, no further tests are necessary because the state must necessarily be in the no-spike region. The test for S_2 is made less costly on average by using the auxiliary inequality (Equation 49). This algorithm uses one test less overall than the previous one, but it may require one more test on average, if NS_1 is the region with highest occupation frequency. The left-to-right evaluation assumption does not always hold in modern processors, which build their own statistics to optimize the test order of logical constructs.

Algorithm 2

```

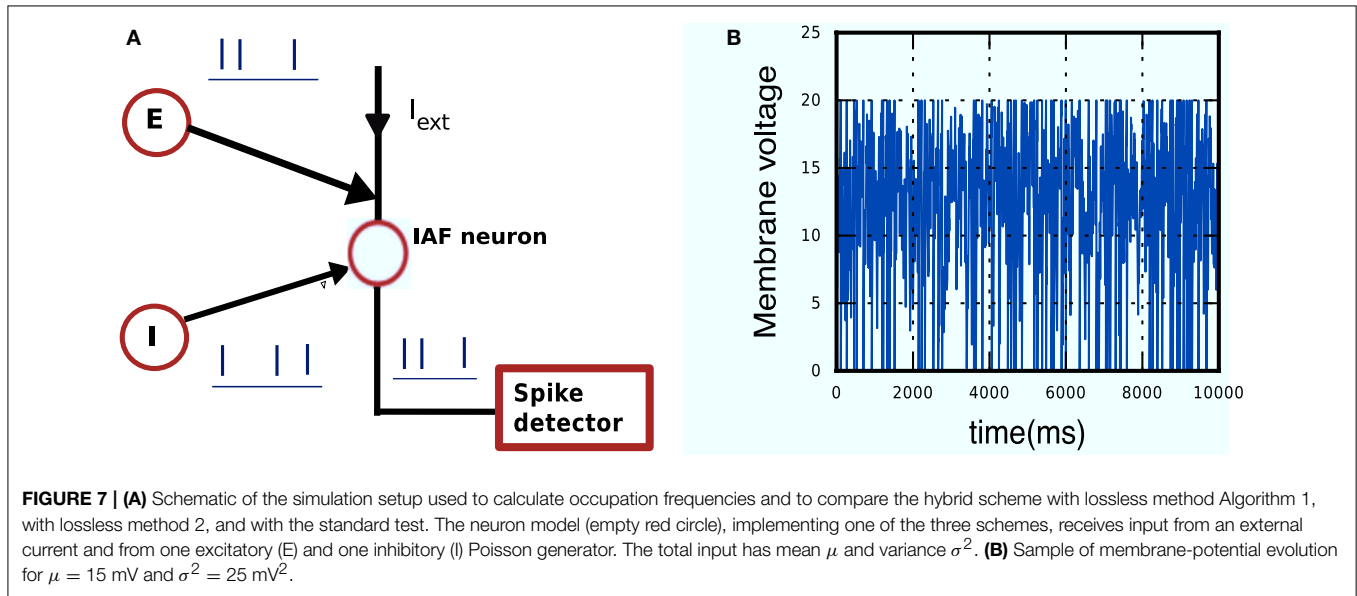
bool is_spike( $h$ ):
  if  $V \geq g_{h,I_e}(I)$  or
    [ $V \geq f_{h,I_e}(I)$  and  $V \geq b_{I_e}(I)$ ] then
    return true
  else
    return false

```

The analysis assumed $I_e < I_\theta \equiv \theta C/\tau$. The reduced system (47) and the Algorithms 1 and 2 are, however, also valid in the case that $I_e \geq I_\theta$, corresponding to the lower panel of Figure 5. In this case subregions NS_2 and S_2 do not exist below the threshold, and the inequalities $V < b_{I_e}(I)$ and $V < g_{h,I_e}(I)$ are always satisfied when $V < \theta$; they always evaluate to true in both algorithms. Both algorithms therefore correctly distinguish spiking from non-spiking states in this case, although they become inefficient owing to the additional superfluous evaluations of $b_{I_e}(I)$ and $g_{h,I_e}(I)$ for spiking states. We decide not to modify them in the present work because the case $I_e \geq I_\theta$ is unusual in real applications. More efficient algorithms for this case can be designed by interested readers following the guidelines just given in this section.

4.4.2. Occupation Frequencies

We want to measure the occupation frequencies in the typical case of a neuron embedded in a recurrent network, receiving fluctuating synaptic input. The setup for this simulation is illustrated in Figure 7A and its formula explained in Appendix section A.2. One neuron is coupled with strengths J and $-J$ to an excitatory and an inhibitory Poisson generator and



also receives a constant external current. The Poisson generators each mimic an excitatory and an inhibitory population. The neuron thus receives a fluctuating input current having average μ and variance σ^2 . In this instance the code of the simulation includes a subroutine that informs us of the current state every time the lossless method test is called, without altering the test or the dynamics.

Figure 8A gives a visual idea of the occupation frequencies for $\mu = 15$ mV, $\sigma^2 = 25$ mV², and $J = 0.1$ mV (all expressed in volts through multiplication by a resistance of $\tau/C = 40$ M Ω), which correspond to the case $I_e < I_\theta$. These values correspond to a composite average input of 250,000 spikes/s, and a total average input current of 400 pA. This presynaptic input makes the neuron fire at an average rate of seven spikes/s. A sample of its membrane evolution is shown in **Figure 7B**. Subregion NS_1 has the overwhelmingly largest occupation frequency. The other three subregions have actually very small areas, as clear from the axis ranges of **Figure 8B**, owing to the very small value of the average timestep, $h = 4 \times 10^{-3}$ ms, given by the inverse of the input rate in this hybrid scheme.

A more precise comparison of the occupation frequencies of the four regions NS_1 , NS_2 , S_1 , S_2 is shown in **Figure 9** for several combinations of three network parameters, producing different dynamical regimes. The parameters are the average μ , the variance σ^2 of the input current, and the presynaptic coupling strength J (all expressed in volts through multiplication by a resistance of $\tau/C = 40$ M Ω). The values of the parameters (μ, σ^2, J) include typical realistic cases as well as some extreme cases, like unusually high coupling strengths. Each panel of **Figure 9** shows the occupation frequencies for a set of dynamical regimes with constant J, β , and several σ^2 . The panels in the last row correspond to the case $I_e \equiv \mu C/\tau \geq I_\theta \equiv \theta C/\tau$, or $\mu \geq \theta$, in which subregions NS_2 and S_2 do not exist below threshold.

It is important to remember that the boundary and size of the regions of **Figure 6** vary with the timestep h , which is

a parameter of the simulation scheme, not of the dynamics *per se*. In an event-driven or hybrid scheme, this step varies inversely with the event input rate, which for Poisson input generators is proportional to σ^2/J^2 . As a consequence, the frequencies displayed in **Figure 9** are not determined by the Liouville distribution of the dynamical trajectory (Equation 27) alone, but also by the details of the numerical-implementation scheme. The dependence of the boundaries on h is illustrated in **Figure 8B**. As h decreases, the line $V = f_{h,I_e}(I)$ and the auxiliary line $V = g_{h,I_e}(I)$ gets closer to the threshold, and the subregions S_1, S_2, NS_2 disappear. This is plausible since in the limit of $h = 0$ we are not evolving the initial state at all. As h increases the point of tangency between the envelope $V = b_{I_e}(I)$ and the line $V = f_{h,I_e}(I)$ moves to increasingly lower voltages and higher currents; subregion S_2 takes over S_1 and subregion NS_1 becomes wider. For typical timestep values of several milliseconds, though, subregions S_1, S_2, NS_2 are still very small.

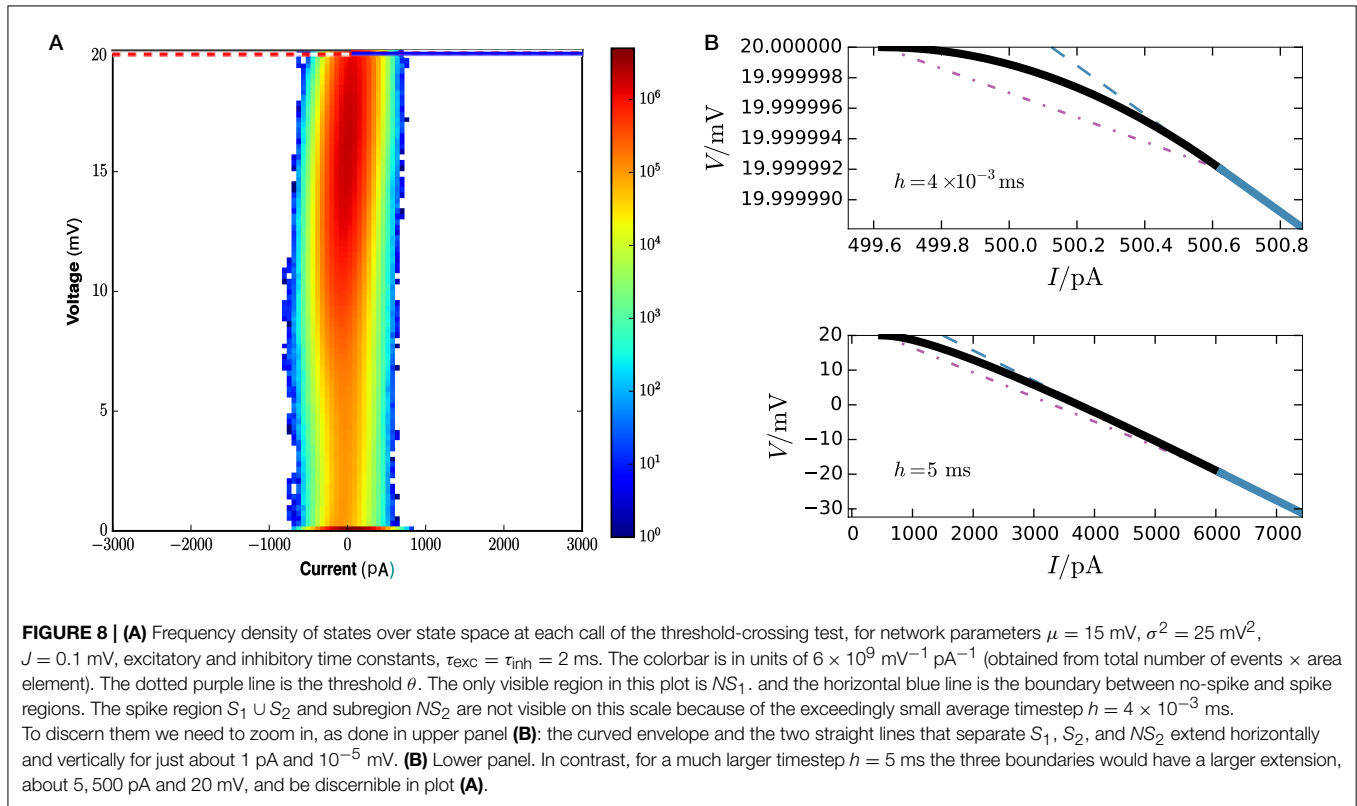
A rough estimate of the dependence of the areas of the bounded regions S_2 and NS_2 on the parameters (μ, σ^2, J) , for $\mu < \theta$, can be obtained by looking at **Figure 9** and considering that these areas together form a triangle with vertices

$$(I_\theta - I_e, \theta), \quad (e^{h/\tau_s}(I_\theta - I_e), b_{I_e}[e^{h/\tau_s}(I_\theta - I_e)]), \quad (I_f, \theta),$$

with I_f such that $f_{h,I_e}(I_f) = \theta$. (50)

This triangle has base $|I_f - (I_\theta - I_e)|$ and height $|\theta - b_{I_e}[e^{h/\tau_s}(I_\theta - I_e)]|$. Expressing h and I_e in terms of μ and σ^2 using Equation (A3), where h is inversely proportional to the input rate $r_I + r_E$, we find

$$\text{areas of } S_2 \text{ and } NS_2 \propto \frac{C}{\tau} (\theta - \mu)^2 \left[1 - \frac{\tau e^{h/\tau} - \tau_s e^{h/\tau_s}}{\tau - \tau_s} \right] \left[\frac{e^{h/\tau_s} (\tau - \tau_s) (1 - e^{h/\tau})}{\tau_s (e^{h/\tau} - e^{h/\tau_s})} - 1 \right] \text{ with } h = \frac{\tau J^2}{\sigma^2}. \quad (51)$$



When $\tau_s \lesssim \tau$ and $J^2 \lesssim \sigma^2$ a Taylor expansion in J^2/σ^2 to fourth order gives a good approximation, with a relative error below 10%:

$$\begin{aligned} \text{areas of } S_2 \text{ and } NS_2 \propto & \frac{C}{\tau} (\theta - \mu)^2 \left[\frac{\tau^2}{4\tau_s^3} \left(\frac{J^2}{\sigma^2} \right)^3 \right. \\ & \left. + \frac{\tau^2 (\tau + \tau_s)}{8\tau_s^3} \left(\frac{J^2}{\sigma^2} \right)^4 + O\left(\frac{J^2}{\sigma^2} \right)^5 \right]. \end{aligned} \quad (52)$$

This approximate formula shows that subregions NS_2 and S_2 grow with the square of the input mean μ and with the third or fourth power of the ratio J^2/σ^2 . Recall that these regions do not exist for $\mu \geq \theta$. The occupation frequencies do not depend on the areas alone, however, but also on the dynamics, as explained in the previous section. We can identify several other dynamical mechanisms for their dependence on the parameters (μ, σ^2, J) :

- an increase in mean input μ leads to more frequent threshold crossings, thus frequently bringing the voltage to its reset value, underneath subregions NS_2 and S_2 . The occupation frequencies of these subregions may therefore decrease with μ even though their areas grow with μ ;
- for low mean input μ , an increase in variance σ^2 means a higher chance of high- V regions, and thus an increase in the

occupation frequencies of NS_2 and S_2 , even though their areas shrink with σ^2 ;

- for mean input μ close to the threshold, an increase in the variance σ^2 leads to more frequent threshold crossings, and may thus increase occupation frequency of S_2 with σ^2 , even though its area shrinks with σ^2 .

The occupation frequency of subregion NS_1 (green circles) dominates all others, varying from 90% to 100% depending on the network parameters. Subregion S_1 (yellow crosses) follows in order of frequency and is the most frequently visited between the two spike subregions. Subregions NS_2 (blue squares) and S_2 (red stars) are scarcely visited for lower synaptic amplitudes, with frequencies from 0 to 10^{-6} ; and slightly more often at higher synaptic amplitudes (frequencies from 0 to 10^{-2}).

4.4.3. Time Performance and Accuracy of a Hybrid Scheme Based on the Lossless Method

The average time costs of Algorithms 1 and 2 within a hybrid scheme can be assessed by real-time simulation measurements. The average time cost of a hybrid scheme based on the standard test (1) can also be assessed in the same way for comparison. We therefore compare the two algorithms of the lossless method and the standard test in this section.

The basic setup is the same as for the frequency analysis of section 4.4.1, explained in Appendix section A.2, with network parameters (μ, σ^2, J) . The only difference is that in the present case the code does not include the subroutine

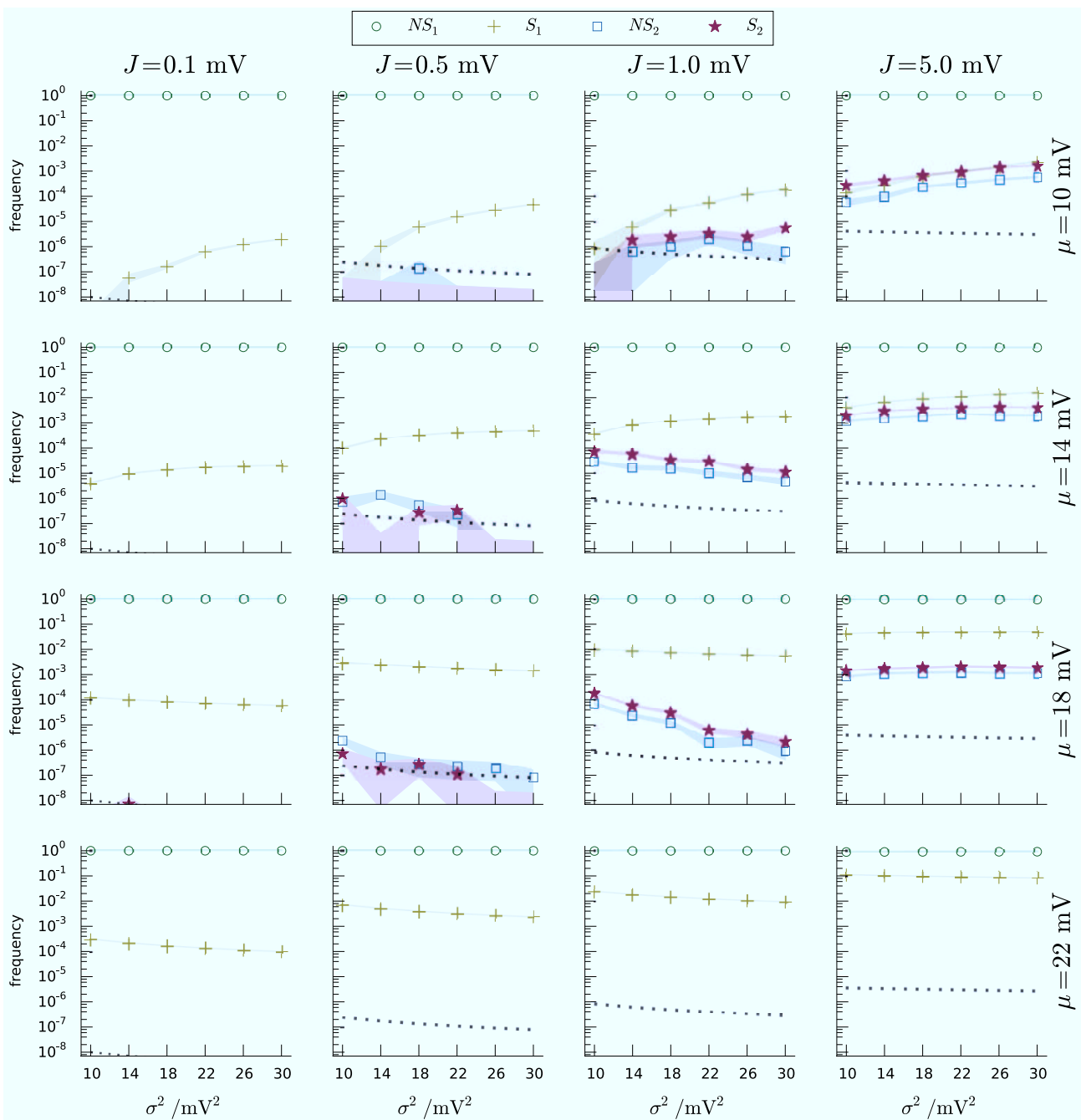


FIGURE 9 | Occupation frequencies of the four subregions NS_1 (green circles), S_1 (yellow crosses), NS_2 (blue squares), S_2 (red stars) of **Figure 6**, for various sets (μ, σ^2, J) of input-current mean and variance, and synaptic strength. Each panel shows the frequencies vs. current variance for fixed current mean and synaptic strength. The columns have the same J , ranging from 0.1 mV (leftmost) to 5 mV (rightmost). The rows have the same μ , ranging from 10 mV (top) to 22 mV (bottom). The frequencies were measured from N samples, depending on (μ, σ^2, J). Excitatory and inhibitory time constants, $\tau_{\text{exc}} = \tau_{\text{inh}} = 2$ ms. The dotted lines in each plot show the inverse number of samples $1/N$ for that network regime. The thickness of the segments connecting the data points equals one standard deviation. The shaded regions show where the limiting frequencies ($N \rightarrow \infty$) are expected to lie with 87% probability, using a Johnson-Dirichlet model with parameter $k = 0.05$ determined by posterior maximization (Johnson, 1932; Good, 1966; Zabell, 1982, 2005; Bernardo and Smith, 2000, § 3.2.5). The frequencies of region S_2 (red stars) are particularly important: they are the frequencies of spike-misses of the standard test (Equation 1).

that informs us of the frequencies, which would otherwise increase and bias the real-time durations of the simulations. Three instances of the basic setup are prepared: in the first the

neuron is modeled by a hybrid scheme with lossless method Algorithm 1, in the second the neuron is modeled by a hybrid scheme with lossless method Algorithm 2, and in the third

the neuron is modeled by a hybrid scheme with the standard threshold-crossing test (Equation 1). The various random-number-generator seeds of the three instances are exactly the same, so that the three neurons receive exactly the same input, spike-for-spike; this is essential for a fair comparison between the three schemes. The three instances are run for a long time (2×10^6 ms) and repeated (in parallel) for several times (Equation 10), enough to collect reliable statistics. The statistics are collected for the same sets of (μ, σ^2, J) values as in the frequency analysis. The total real-time length of a simulation depends, in all three schemes, on how often the checkpoints and threshold-crossing tests occurs, and this in turn depends on the presynaptic input frequency, as already discussed.

Rather than showing the results for all sets of parameters (μ, σ^2, J) , which in this case are not very informative, we show in **Figure 10** those with the (J, μ) values that yield the slowest and fastest performances. The average computation costs of Algorithms 1 and 2, which embody the lossless method, turn out to be very similar—within each other's standard deviations—and basically identical in comparison with the cost of the scheme based on the standard test (Equation 1). Both are slower than the hybrid scheme with the standard test: from around 30% slower in the case of high-activity regime with frequent incoming spikes ($\mu = 18$ mV, $J = 0.1$ mV), to around 20% slower in the case of low activity regime and infrequent incoming spikes ($\mu = 10$ mV, $J = 5$ mV). These are the extremes shown in **Figure 10**. In most other sets of network parameters the hybrid scheme with lossless method was around 20–25% slower than the standard hybrid scheme with threshold-crossing test (Equation 1).

As mentioned in section 4.3, subregion S_2 contains all states for which the standard threshold-crossing test misses a spike. The occupation frequencies of this subregion are therefore a direct measure of the number of spike missed, per neuron, by the standard hybrid scheme. They are shown as red stars in **Figure 9** for the various sets of network parameters (μ, σ^2, J) . The frequency of missed spikes does not have a simple monotonic dependence on the three parameters, owing to the interaction of several mechanisms, discussed in section 4.4.2. An increase in synaptic coupling J or input current μ generally leads to more frequently missed spikes because the neuron spikes more often overall. For very high—suprathreshold—input currents, however, the frequency decreases again until no spikes are missed anymore; this change in trend happens because the checkpoints become more frequent. Regimes of low J are diffusion-like processes, where frequent arrival of synaptic events does not lead to missed spikes. Regimes of high J are shot-noise processes, where sudden and infrequent arrival of synaptic events leads to suprathreshold excursions and missed spikes. Separate simulations show that the hybrid scheme based on the lossless method, with either algorithm, reproduce the analytic solution of the neuron model within floating point precision.

For biologically realistic synaptic couplings, $J < 1$ mV, the standard hybrid scheme is 30% faster than the scheme with the

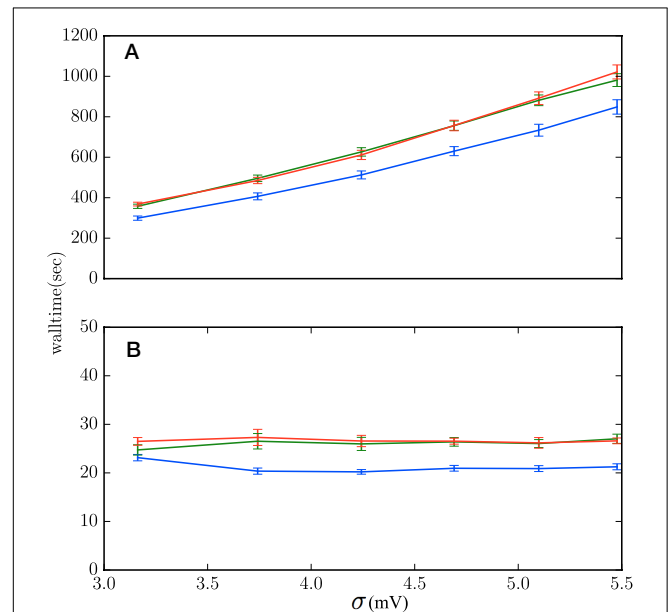


FIGURE 10 | Computational costs of the hybrid scheme with lossless method, Algorithm 1 (red), lossless method, Algorithm 2 (green), and standard threshold-crossing test (1) (blue). **(A)** $\mu = 18$ mV, $J = 0.1$ mV describes regimes where the lossless method has the highest increase in computational cost, around 30%. These are regimes of high activity and frequently incoming spikes. **(B)** $\mu = 10$ mV, $J = 5$ mV describe regimes where the lossless method has the lowest increase in computational cost, around 20%. These are regimes of low activity and infrequently incoming spikes. The data comes from 10 simulations of 2×10^6 ms simulation-time each.

lossless method, and misses <1 spike every 10^6 test calls, per neuron; for very low couplings $J < 1$ mV this figure even becomes <1 spike every 10^8 test calls. For synaptic couplings $J > 1$ mV the standard hybrid scheme starts to miss more spikes, reaching even one missed spike every 500 test calls for subthreshold average input currents; and it is only 8% faster than the hybrid scheme with the lossless method. From these figures a user can decide to use the hybrid scheme with the standard or the lossless test, depending on the desired balance of accuracy and speed.

5. SUMMARY AND DISCUSSION

We have presented a general method to detect threshold crossing for an integrable, affine or linear neuronal dynamical law. This method is based on the geometric idea of propagating the threshold plane backwards in time and to determine whether the swept volume contains the initial state, rather than checking whether the membrane potential of the forward-evolved state has a maximum above threshold. These two procedures are mathematically equivalent but geometrically different. The forward-propagation procedure geometrically checks for the intersection of a curve (the state trajectory) with a hypersurface (the threshold). Our backward-propagation

procedure geometrically checks for the “intersection” of a point (the state) with a hypervolume (the threshold trajectory), i.e., the inclusion of the former in the latter. This geometric view translates into a set of inequalities that an initial state has to satisfy in order not to cross the threshold during its evolution; and vice versa, if the system is not satisfied, a threshold crossing is certain. If in the latter case its time of occurrence is calculated by standard root-finding algorithms (Press et al., 2007, ch. 9).

We have calculated the system of inequalities expressing the threshold-crossing condition for a generic affine or linear neuronal dynamical law in any dimension. The result is the conjunction of inequalities (Equation 25). It consists of two linear inequalities in the state-space variables, voltage and currents, and a non-linear one. The numerical implementation of this system of inequalities can be further optimized, on a case-by-case basis. In order to give a concrete implementation example of the generic inequalities (Equation 25), to show their geometrical meaning, and to give an example of optimization, we have applied our procedure step-by-step to the two-dimensional case of a leaky integrate-and-fire neuron with exponentially decaying post-synaptic currents (Rotter and Diesmann, 1999). The generic inequalities (Equation 25) take in this case the concrete form (Equation 46).

The quantitative data in the present work are obtained by integrating these inequalities into a combined event-and-time-driven simulation framework (Morrison et al., 2007) for large-scale spiking neuronal network models as released by Bos et al. (2015). Implementation and comparison to earlier work show that:

- the system of inequalities, the non-linear one in particular, can be expressed analytically in terms of the state-space variables even when the original threshold-crossing condition involves a transcendental equation; compare Equation (46) with Equation (31);
- the computationally expensive non-linear function in the system can be conveniently approximated by a simplicial mesh, speeding up the algorithm even further by testing a linear inequality first, ruling out the majority of initial states;
- our method reproduces the analytic solution of the neuron model within floating point precision. It detects all threshold crossings, in particular those that the approximate test (Equation 1) of Hanuschkin et al. (2010) misses in some ranges of mean activity, fluctuation, and synaptic-coupling strength (Figure 9);
- at the default spike accuracy of 0.1 ms of the reference simulator, our method is 20–30% slower than the fastest available solver with spike loss (Hanuschkin et al., 2010). It therefore compares with embedded event driven methods in speed (see Hanuschkin et al., 2010, Figure 5, inset).

In practice the method of Hanuschkin et al. (2010) rarely misses spikes for biologically realistic synaptic amplitudes. At low frequencies of afferent synaptic events, however, threshold crossings can be missed, because fewer incoming events mean

fewer checkpoints on the grid, and more time available for the voltage of a neuron to surpass threshold and go below it again. Our method offers an alternative for users who need exact spike detection and are willing to pay a price in terms of a slightly longer computation time.

In many simulations the time constants of excitatory and inhibitory currents are different, because these currents are mediated by various receptors, each with its characteristic time scale. The state of such a model is a vector in a three-dimensional state space, its variables being the excitatory and inhibitory currents and the membrane voltage. If the dynamical law of such model is still affine or linear, the general method we have here presented can still be applied to find the threshold crossings without losses. The threshold is in this case a two-dimensional flat surface, and its back-propagated image is also a flat surface within the three-dimensional space, as shown in Figure 4.

In state spaces of higher dimensions it can be more difficult to derive the non-linear inequality of the curved envelope in the system (Equation 25) in closed form, but it is still possible to construct a nested sequence of linear inequalities, corresponding to a flat simplicial mesh over the curved envelope, that approximate the non-linear inequality to any desired precision, as explained in section 3.2.3 and exemplified in section 4.4.1. Such construction needs to be done only once for any given model, prior to the simulation, and it cannot miss spikes owing to the concavity of the envelope. Being linear, such nested inequalities are likely computationally not too expensive, but further work must investigate the scaling of the computational cost of our method in such cases. The model considered here presents the simplest non-trivial case in which our algorithm can be applied.

At the same time, the step from a model without synaptic filtering to one with synaptic filtering is of qualitative nature: the response properties of the two models are fundamentally different, for example in terms of the weak signal transmission in the limit of high frequencies (Brunel and Wang, 2001). As such the described model is relevant, because it remedies some artifacts of its simpler counterparts.

An important extension of the method presented here is the inclusion of non-linear sub-threshold dynamics, such as in Hodgkin-Huxley dynamics, the quadratic integrate and fire model, or the exponential integrate-and-fire model (Gerstner and Kistler, 2002). Having a non-linear sub-threshold dynamics does not invalidate the general idea we have used, but makes it more complex to derive a system of inequalities separating the states that lead to spikes from those who do not. In principle the idea of back-propagation of the threshold plane can still be applied to a non-linear dynamical law, although the states on the threshold, when backpropagated, will not lie on a hyperplane. The hypersurface separating the spike and no-spike regions is still a set of critical points given by the vanishing of the Jacobian determinant of the backpropagation map, as in section 3.2.1. The uniqueness of the solution of differential equations guarantees the validity of this approach. In practice the resulting mathematical expressions will be analytically tractable only in very special cases. Yet, the resulting envelope could still be found with numerical methods—as we said this only needs to

be done once per simulation—and it could be approximated by a mesh of simplices or splines.

Most models with non-linear sub-threshold evolution equations, however, do not require the approach described here in the first place. The reason is that the non-linearity in these models typically causes the generation of an explicit action potential, the rapid increase (or divergence in some models) of the membrane potential. This is certainly true for the examples mentioned above. The very threshold-crossing problem that we have solved here for the simplified leaky integrate-and-fire dynamics is therefore absent in these more biophysically realistic models at the expense of having to solve a non-linear set of differential equations at each time step.

The problem of detecting some sort of threshold crossing in a system of coupled first-order linear differential equations appears in many other applications and phenomena like switching, friction, and saturation (Hiebert and Shampine, 1980). For example, in an air-conditioning unit a thermostat controls the on-off state based on a certain threshold value of the room temperature (Shampine, 1994). The dynamics of this system is similar to that described in section 4.1. Another example is the problem of ejecting a pilot such that collision with the aircraft stabilizer is avoided.

The present work has used concepts from differential geometry, in particular extrusions (Bossavit, 2003) and critical points of maps between manifolds, and shows that these concepts have a readily understandable geometrical and visual meaning. The notion of extrusion has recently found applications in numerical and discretization techniques for partial and integral differential equations (Desbrun et al., 2005). The notion of critical points of a manifold mapping is ubiquitous in science: from the caustics of propagating seismic fronts, at which the seismic wave changes its phase (Romanowicz and Dziewonski, 2007) § 1.04, to the singularities between two coordinate charts in general relativity (Misner et al., 2003), which affect the accuracy of global navigation satellite systems (Coll et al., 2012; Sáez and Puchades, 2013).

Indeed, the neuron model analyzed in section 4 exhibits a similarity with the dynamics of a point mass near a black hole. If the simulation timestep h is very large the curved surface separating the states that lead to a spike from those that do not acts like an event horizon in general relativity: a state evolved from the spike region can enter the no-spike region, but once

there it cannot escape and will always remain a “no-spike” state. This is only true for the dynamical law (Equation 7), though, with constant affine term and no resets at threshold. Inputs from other neurons lead to discontinuous changes in the affine term of the dynamical law, causing a “transport” of initial states out of the event horizon, from the no-spike to the spike region. Nevertheless, such similarities are maybe more than mere coincidences. For example, the trajectory of the threshold surface of a leaky integrate-and-fire model with α -shaped post-synaptic currents (Bernard et al., 1994) can be implicitly expressed in terms of the Lambert- W function (Corless et al., 1996), as an analysis along the lines of section 3.1 shows; and this function also appears in the implicit expression of point-mass trajectories in $(1+1)$ -dimensional general relativity (Mann and Ohta, 1997). It is surely worthwhile to bring the younger field of neuronal dynamics closer to ideas and techniques of the elder fields of differential geometry and general relativity.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct and intellectual contribution to the work, and approved it for publication.

FUNDING

This work was supported by Deutsche Forschungsgemeinschaft (DFG) through grant GSC111; Helmholtz association: VH-NG-1028 and SMHB; European Union Seventh Framework Programme under grant agreement no. 604102 (Human Brain Project, HBP) and the European Union’s Horizon 2020 research and innovation programme under grant agreement no. 720270 (HBP SGA1); Juelich Aachen Research Alliance (JARA).

ACKNOWLEDGMENTS

PP thanks the Forschungszentrum librarians for their always prompt and kind help, Mari & Miri for continuous encouragement and affection, Buster Keaton and Saitama for filling life with awe and inspiration, and the developers and maintainers of $\text{\LaTeX}$, Emacs, AUCTeX, MiKTeX, Open Science Framework, biorXiv, PhilSci, Python, Inkscape, Sci-Hub for making a free and unfiltered scientific exchange possible.

REFERENCES

- Abramowitz, M., and Stegun, I. A. (eds.) (1972). *Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables*, Vol. 55, National Bureau of Standards Applied Mathematics Series, 10th Edn. Washington, DC: U.S. Department of Commerce. Originally Published in 1964.
- Artin, E. (1955). *Geometric Algebra*, Vol. 3, Interscience Tracts in Pure and Applied Mathematics. New York, NY: Interscience.
- Bernard, C., Ge, Y. C., Stockley, E., Willis, J. B., and Wheal, H. V. (1994). Synaptic integration of NMDA and non-NMDA receptors in large neuronal network models solved by means of differential equations. *Biol. Cybern.* 70, 267–273. doi: 10.1007/BF00197607
- Bernardo, J.-M., and Smith, A. F. (2000). *Bayesian Theory*. Wiley Series in Probability and Mathematical Statistics. New York, NY: Wiley. Originally Published in 1994.
- Bos, H., Morrison, A., Peyser, A., Hahne, J., Helias, M., Kunkel, S., et al. (2015). *Nest 2.10.0*. doi: 10.5281/zenodo.44222. Available online at: <http://www.nest-simulator.org/>
- Bossavit, A. (1991). *Differential Geometry: For the Student of Numerical Methods in Electromagnetism*. Available online at: https://www.researchgate.net/publication/200018385_Differential_Geometry_for_the_student_of_numerical_methods_in_Electromagnetism

- Bossavit, A. (2002). *Applied Differential Geometry (A Compendium)*. Originally Published in 1994. Available online at: <http://wwwold.icm.edu.pl/edukacja/mat/Compendium.php>
- Bossavit, A. (2003). Extrusion, contraction: their discretization via Whitney forms. *COMPEL* 22, 470–480. doi: 10.1108/03321640310474877
- Brette, R. (2007). Exact simulation of integrate-and-fire models with exponential currents. *Neural Comput.* 19, 2604–2609. doi: 10.1162/neco.2007.19.10.2604
- Brunel, N., and Wang, X.-J. (2001). Effects of neuromodulation in a cortical network model of objectworking memory dominated by recurrent inhibition. *J. Comput. Neurosci.* 11, 63–85. doi: 10.1023/A:1011204814320
- Burke, W. L. (1987). *Applied Differential Geometry*. Cambridge: Cambridge University Press. Originally Published in 1985.
- Burke, W. L. (1995). *Div, Grad, Curl Are Dead*. Available online at: https://people.ucsc.edu/~rmont/papers/Burke_DivGradCurl.pdf; “preliminary draft II”. See also <http://www.icolick.org/~burke/>
- Choquet-Bruhat, Y., DeWitt-Morette, C., and Dillard-Bleick, M. (1996). *Analysis, Manifolds and Physics. Part I: Basics*. Amsterdam: Elsevier. Originally Published in 1977.
- Coll, B., Ferrando, J. J., and Morales-Lladosa, J. A. (2012). Positioning systems in Minkowski space-time: bifurcation problem and observational data. *Phys. Rev. D* 86:084036. doi: 10.1103/PhysRevD.86.084036
- Corless, R. M., Gonnet, G. H., Hare, D. E. G., Jeffrey, D. J., and Knuth, D. E. (1996). On the Lambert W function. *Adv. Comput. Math.* 5, 329–359. doi: 10.1007/BF02124750
- Coxeter, H. S. M. (1969). *Introduction to Geometry, 2nd Edn.* New York, NY: Wiley. Originally Published in 1961.
- de Rham, G. (1984). *Differentiable Manifolds: Forms, Currents, Harmonic Forms, Vol. 266, Grundlehren der Mathematischen Wissenschaften*. Berlin: Springer. Transl. by F. R. Smith. Originally Published in French in 1955.
- Dehaene, S., and Changeux, J.-P. (2005). Ongoing spontaneous activity controls access to consciousness: a neuronal model for inattentive blindness. *PLOS Biol.* 3:e141. doi: 10.1371/journal.pbio.0030141
- Desbrun, M., Hirani, A. N., Leok, M., and Marsden, J. E. (2005). Discrete exterior calculus. *arXiv:0508341*.
- D’Haene, M., and Schrauwen, B. (2010). Fast and exact simulation methods applied on a broad range of neuron models. *Neural Comput.* 22, 1468–1472. doi: 10.1162/neco.2010.07-09-1070
- D’Haene, M., Schrauwen, B., Van Campenhout, J., and Stroobandt, D. (2009). Accelerating event-driven simulation of spiking neurons with multiple synaptic time constants. *Neural Comput.* 21, 1068–1099. doi: 10.1162/neco.2008.02-08-707
- Dodson, C. T. J., and Poston, T. (1991). *Tensor Geometry: The Geometric Viewpoint and its Uses, Vol. 130 of Graduate texts in Mathematics, 2nd Edn.* Springer. Originally Published in 1977.
- Ferscha, A. (1996). “Parallel and distributed simulation of discrete event systems,” in *Parallel and Distributed Computing Handbook*, ed A. Y. Zomaya (Maui, HI: McGraw-Hill), 1003–1041.
- Fourcaud, N., and Brunel, N. (2002). Dynamics of the firing probability of noisy integrate-and-fire neurons. *Neural Comput.* 14, 2057–2110. doi: 10.1162/089976602320264015
- Fujimoto, R. M. (2000). *Parallel and Distributed Simulation Systems*. New York, NY: Wiley.
- Gerstner, W., and Kistler, W. (2002). *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press.
- Gewaltig, M.-O., and Diesmann, M. (2007). NEST (NEural Simulation Tool). *Scholarpedia* 2:1430. doi: 10.4249/scholarpedia.1430
- Good, I. J. (1966). How to estimate probabilities. *J. Inst. Maths. Appl.* 2, 364–383. doi: 10.1093/imamat/2.4.364
- Hansel, D., Mato, G., Meunier, C., and Neltner, L. (1998). On numerical simulations of integrate-and-fire neural networks. *Neural Comput.* 10, 467–483. doi: 10.1162/089976698300017845
- Hanuschkin, A., Kunkel, S., Helias, M., Morrison, A., and Diesmann, M. (2010). A general and efficient method for incorporating precise spike times in globally time-driven simulations. *Front. Neuroinform.* 4:113. doi: 10.3389/fninf.2010.00113
- Hiebert, K. L., and Shampine, L. F. (1980). *Implicitly Defined Output Points for Solutions of ODEs*. Albuquerque, NM: Sandia Laboratories.
- Johnson, W. E. (1932). Probability: the deductive and inductive problems. *Mind* 41, 409–423. doi: 10.1093/mind/XLI.164.409
- Krishnan, J., Porta Mana, P., Helias, M., Diesmann, M., and Di Napoli, E. (2016). “Perfect detection of spikes via time-reversal,” in *Proceedings of the Bernstein Conference* (Berlin).
- Kunkel, S., Helias, M., Diesmann, M., and Morrison, A. (2011). Fail-safe detection of threshold crossings of linear integrate-and-fire neuron models in time-driven simulations. *BMC Neurosci.* 12:P229. doi: 10.1186/1471-2202-12-S1-P229
- Mann, R. B., and Ohta, T. (1997). Exact solution for the metric and the motion of two bodies in $(1 + 1)$ -dimensional gravity. *Phys. Rev. D* 55, 4723–4747. doi: 10.1103/PhysRevD.55.4723
- Marsden, J. E., and Ratiu, T. (2007). *Manifolds, Tensor Analysis, and Applications, 3rd Edn.* New York, NY: Springer. Written with the collaboration of R. Abraham. Originally Published in 1983. Available online at: <http://www.math.cornell.edu/~web6520/Abraham-Marsden.pdf>
- Mattia, M., and Del Giudice, P. (2000). Efficient event-driven simulation of large networks of spiking neurons and dynamical synapses. *Neural Comput.* 12, 2305–2329. doi: 10.1162/089976600300014953
- Misner, C. W., Thorne, K. S., and Wheeler, J. A. (2003). *Gravitation, 25th Edn.* New York, NY: W. H. Freeman and Company. Originally Published in 1970.
- Morrison, A., and Diesmann, M. (2008). “Maintaining causality in discrete time neuronal network simulations,” in *Lectures in Supercomputational Neuroscience: Dynamics in Complex Brain Networks, Understanding Complex Systems*, eds P. Graben, C. Zhou, M. Thiel, and J. Kurths (Berlin: Springer), 267–278.
- Morrison, A., Straube, S., Plesser, H. E., and Diesmann, M. (2007). Exact subthreshold integration with continuous spike times in discrete time neural network simulations. *Neural Comput.* 19, 47–79. doi: 10.1162/neco.2007.19.1.47
- Nomizu, K., and Sasaki, T. (1994). *Affine Differential Geometry: Geometry of Affine Immersions, Vol. 111, Cambridge Tracts in Mathematics*. Cambridge: Cambridge University Press.
- Porta Mana, P. G. L. (2011). Notes on affine and convex spaces. arXiv:1104.0032
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). *Numerical Recipes: The Art of Scientific Computing, 3rd Edn.* Cambridge: Cambridge University Press. Originally Published in 1988.
- Ramanan, S. (2005). *Global Calculus, Vol. 65, Graduate Studies in Mathematics*. Providence, RI: American Mathematical Society.
- Rauch, A., La Camera, G., Luscher, H. R., Senn, W., and Fusi, S. (2003). Neocortical pyramidal cells respond as integrate-and-fire neurons to *in vivo* like input currents. *J. Neurophysiol.* 90, 1598–1612. doi: 10.1152/jn.00293.2003
- Rockafellar, R. T. (1972). *Convex Analysis, Vol. 28, Princeton Mathematical Series*. Princeton, NJ: Princeton University Press. Originally Published in 1970.
- Romanowicz, B., and Dziewonski, A. (eds.) (2007). *Seismology and the Structure of the Earth, Vol. 1, Treatise on Geophysics, 1st Edn.* Amsterdam: Elsevier.
- Rotter, S., and Diesmann, M. (1999). Exact digital simulation of time-invariant linear systems with applications to neuronal modeling. *Biol. Cybern.* 81, 381–402. doi: 10.1007/s004220050570
- Sáez, D., and Puchades, N. (2013). Relativistic positioning systems: numerical simulations. *Acta Futura* 7, 103–110. arXiv:1404.1000. doi: 10.2420/AF07.2013.103
- Schouten, J. A. (1989). *Tensor Analysis for Physicists. Dover Books on Physics and Chemistry, 2nd Edn.* New York, NY: Dover. Originally Published in 1951.
- Schucker, J., Diesmann, M., and Helias, M. (2015). Modulated escape from a metastable state driven by colored noise. *Phys. Rev. E* 92:052119. arXiv:1411.0432. doi: 10.1103/PhysRevE.92.052119
- Shampine, L. F. (1994). *Numerical Solution of Ordinary Differential Equations*. New York, NY: Chapman & Hall/CRC.
- Shelley, M. J., and Tao, L. (2001). Efficient and accurate time-stepping schemes for integrate-and-fire neuronal networks. *J. Comput. Neurosci.* 11, 111–119. doi: 10.1023/A:1012885314187
- Simon, U., Schwenk-Schellschmidt, A., and Viesel, H. (1992). *Introduction to the Affine Differential Geometry of Hypersurfaces. Lecture Notes of the Science University of Tokyo*. Tokyo: Science University of Tokyo.
- Spivak, M. (1999). *A Comprehensive Introduction to Differential Geometry, Vol. 1, 3rd Edn.* Houston, TX: Publish or Perish. Originally Published in 1970.

- van Elburg, R. A., and van Ooyen, A. (2009). Generalization of the event-based carnevale-hines integration scheme for integrate-and-fire models. *Neural Comput.* 21, 1913–1930. doi: 10.1162/neco.2009.07-08-815
- Zabell, S. L. (1982). W. E. Johnson's "sufficientness" postulate. *Ann. Stat.* 10, 1090–1099. doi: 10.1214/aos/1176345975
- Zabell, S. L. (2005). *Symmetry and Its Discontents: Essays on the Q16 History of Inductive Probability*. Cambridge Studies in Probability, Induction, and Decision Theory. Cambridge: Cambridge University Press.
- Zeigler, B. P., Praehofer, H., and Kim, T. G. (2000). *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems, 2nd Edn.* Amsterdam: Academic Press.
- Zheng, G., Tonnelier, A., and Martinez, D. (2009). Voltage-stepping schemes for the simulation of spiking neural networks. *J. Comput. Neurosci.* 26, 409–423. doi: 10.1007/s10827-008-0119-1

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The reviewer MD declared a shared affiliation, though no other collaboration, with the authors to the handling editor, who ensured that the process nevertheless met the standards of a fair and objective review.

Copyright © 2018 Krishnan, Porta Mana, Helias, Diesmann and Di Napoli. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

A. APPENDIX

A.1. Interpolation for Threshold-Crossing Time

If we know that the neuron voltage V is below threshold at times t and $t+h$ but above threshold somewhere in the interval $]t, t+h]$, then by continuity it must reach a maximum above threshold at a time $t_{\max} \in]t, t+h]$. This time can be obtained solving the equation $\dot{V}(t_{\max}) = 0$, with $V(t)$ given by Equation (27). The solution (Hanuschkin et al., 2010) is

$$t_{\max} = -\frac{\tau \tau_s}{\tau - \tau_s} \ln \left[\frac{\tau}{\tau_s} - \frac{\tau - \tau_s}{\tau} \left(\frac{I_e}{I} - \frac{CV}{\tau I} \right) \right], \quad (\text{A1})$$

from which the potential $V(t_{\max}) > \theta$ can also be easily calculated.

The time $t_\theta \in]t, t_{\max}]$ at which the first threshold crossing occurs, $V(t_\theta) = \theta$, must lie between t and $t_{\max}$, with $V(t) < V(t_\theta) < V(t_{\max})$, and can thus be interpolated using a root-finding algorithm (Press et al., 2007).

A.2. Network Setup

The dynamics of the membrane potential V and synaptic current I can be described, if the input is treated stochastically and for weak synaptic couplings, by a diffusion process with equations (Fourcaud and Brunel, 2002)

$$\begin{aligned} \tau \frac{dV}{dt} &= -V(t) + (RI)(t), \\ \tau_s \frac{d(RI)}{dt} &= -(RI)(t) + \mu + \sigma \sqrt{\tau} \xi(t). \end{aligned} \quad (\text{A2})$$

with ξ a zero-mean Gaussian process. The parameters μ and σ^2 characterize the stochastic input and are the mean and variance of the total incoming synaptic current. They are related to the synaptic couplings $\{J_i\}$ and firing rates $\{r_i\}$ of the input neurons via $\mu = \tau \sum_i J_i r_i$ and $\sigma^2 = \tau \sum_i J_i^2 r_i$. If we have one excitatory and one inhibitory population of input neurons, mimicked by two Poisson generators with rates r_E and r_I coupled to the neuron with strengths J_E and J_I (where $J = \frac{\tau}{C} w$ and w is the synaptic weight of the current), and by an input current I_e , then

$$\begin{aligned} \mu &= \frac{I_e \tau}{C} + \tau (J_I r_I + J_E r_E) \\ \sigma^2 &= \tau (J_I^2 r_I + J_E^2 r_E), \\ \frac{1}{r} &= \tau_r + \tau \sqrt{\pi} \int_{\frac{V_r - \mu}{\sigma} + \frac{|\zeta(1/2)|}{\sqrt{2}} \sqrt{\frac{\tau_s}{\tau}}}^{\frac{\theta - \mu}{\sigma} + \frac{|\zeta(1/2)|}{\sqrt{2}} \sqrt{\frac{\tau_s}{\tau}}} e^{y^2} [1 + \text{erf}(y)] dy, \end{aligned} \quad (\text{A3})$$

where r is the output firing rate of the neuron, approximated to linear order in $\sqrt{\tau_s/\tau}$, τ_r is the refractory time, V_r the reset voltage, and ζ the zeta function (Abramowitz and Stegun, 1972).