HAL
open science

Voltage-stepping schemes for the simulation of spiking neural networks

Gang Zheng, Arnaud Tonnelier, Dominique Martinez

To cite this version:

Gang Zheng, Arnaud Tonnelier, Dominique Martinez. Voltage-stepping schemes for the simulation of spiking neural networks. Journal of Computational Neuroscience, 2009, 26 (3), pp.409-423. 10.1007/s10827-008-0119-1 . inria-00189386v4

HAL Id: inria-00189386

https://inria.hal.science/inria-00189386v4

Submitted on 11 Jun 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Voltage-stepping schemes for the simulation of spiking neural networks

G. Zheng[*1], **A. Tonnelier**[*], **D. Martinez**[**]
[*] *INRIA, Inovallée 655 Avenue de l'Europe Montbonnot 38334 Saint Ismier France, and Université de Grenoble et CNRS, Laboratoire Jean Kuntzmann, BP 53, 38041 Grenoble Cdex 9, France {Gang.Zheng, Arnaud.Tonnelier}@inrialpes.fr*
[**] *LORIA, Campus Scientifique, B.P. 239, 54 506 Vandoeuvre-lès-Nancy, France, Dominique.Martinez@loria.fr*

Abstract

The numerical simulation of spiking neural networks requires particular attention. On the one hand, time-stepping methods are generic but they are prone to numerical errors and need specific treatments to deal with the discontinuities of integrate-and-fire models. On the other hand, event-driven methods are more precise but they are restricted to a limited class of neuron models. We present here a voltage-stepping scheme that combines the advantages of these two approaches and consists of a discretization of the voltage state-space. The numerical simulation is reduced to a *local* event-driven method that induces an implicit activity-dependent time discretization (time-steps automatically increase when the neuron is slowly varying). We show analytically that such a scheme leads to a high-order algorithm so that it accurately approximates the neuronal dynamics. The voltage-stepping method is generic and can be used to simulate any kind of neuron models. We illustrate it on nonlinear integrate-and-fire models and show that it outperforms time-stepping schemes of Runge-Kutta type in terms of simulation time and accuracy.
Keywords: *Voltage-stepping; Event-driven; Time-stepping; Spiking neural networks*

1 Introduction

Neuronal information processing involves action potentials, or spikes. Recent findings in neuroscience emphasize the importance of spike timing precision. Individual spike timing seems to play a key role in the coding of sensory information, as some neurons fire no more than one spike during the entire presentation of the stimulus, e.g. Kenyon cells in the insect olfactory system (Perez-Orive *et al.*, 2002) or cortical cells in the rat auditory system (DeWeese *et al.*, 2003). In biological systems where the processing-speed is required to be high, the timings of spikes are very precise and reliable (Mainen and Sejnowski, 1995; VanRullen *et al.*, 2005). Submillisecond precision of spike timing has been reported (Bair and Koch, 1996; Ariav *et al.*, 2003) and small differences in the precision of synaptic events have a severe impact on the plasticity of synapses. Numerical simulations of neural networks are commonly used to explore the spike coding paradigm. It is thus crucial to have accurate and efficient schemes to simulate spiking neural networks. Different strategies have been developed for the simulation of spiking neural networks: event-driven schemes where the timings of spikes are calculated exactly and time-stepping methods

[1] Corresponding Author, Tel: 0033 4 76 61 53 50, Fax: 0033 4 76 61 55 63

that approximate the membrane voltage of neurons on a discretized time (see Brette *et al.* (2007) for a review of simulation environments). In pure event-driven strategies the spike timings are analytically given and are calculated with an arbitrary precision (up to the machine precision). This scheme allows an exact simulation where no spike is missed. This method has become increasingly popular (Mattia and Giudice, 2000; Makino, 2003; Rochel and Martinez, 2003; Brette, 2006; Brette, 2007; Rudolph and Destexhe, 2006; Tonnelier *et al.*, 2007). However only a limited class of simplified neuron models of integrate-and-fire type is amenable to exact simulations.

Time-stepping schemes are generic since they can be applied to any model. Classical integration schemes of Runge-Kutta type have to be modified to properly handle the discontinuities of integrate-and-fire neuron dynamics generated by the resettings and the synaptic events (Hansel *et al.*, 1998; Shelley and Tao, 2001). However when the membrane potential crosses threshold twice during one time step (the first crossing is from below and the second is in the downward direction), the spike event may be missed. Due to the discontinuous nature of integrate-and-fire network, a failure to detect a spike may cause dramatic changes on the behavior of the system and artificial dynamical states may be created if the time step is badly chosen (Hansel *et al.*, 1998). Moreover a fundamental limitation on the accuracy of these methods is imposed by the smoothness of the postsynaptic potentials (Shelley and Tao, 2001).

In this paper we define a generic scheme for the simulation of neural networks based on a voltage-space discretization that we call voltage-stepping scheme. This scheme retains the advantages of the accuracy and the activity-dependent computational cost of event-driven strategies while allowing a generic simulation of any neural model. The greatest asset is to define an implicit and adaptive time-discretization for each neuron that depends on its own activity. A neuron that evolves slowly allows long time steps and has a low computational cost whereas small time steps are required for fast varying neurons. The proposed strategy has a clear advantage when the inter-event period is greater than the computational time step used in classical time-stepping methods. Here we show that our implicit and variable time-stepping scheme allows high-order integration methods. Since recent efforts have been made on the numerical simulations of integrate-and-fire networks (Brette, 2006; Brette, 2007; Brette *et al.*, 2007; Ros *et al.*, 2006; Morrison *et al.*, 2007; Rangan and Cai, 2007; Rudolph and Destexhe, 2006) we illustrate in the next section our method using a general nonlinear integrate-and-fire model with synaptic currents. Generalization to other models are proposed. In section 3 we present numerical results using the quadratic integrate-and-fire model and compare the performance of the voltage-stepping scheme with standard time-stepping integration methods.

2 Method

2.1 Voltage-stepping scheme

Consider the kth integrate-and-fire neuron in a network described by its membrane potential, v^k, that evolves according to the equation

$$C\frac{dv^k}{dt} = f(v^k) + I_0 + I^k_{syn}(t), \tag{1}$$

where C is the membrane capacitance, $f(v)$ the nonlinear current-voltage characteristic of the membrane, I_0 an external constant input current and I^k_{syn} the total synaptic input received by neuron k. A spike is triggered when v^k reaches the threshold v_{th} upon when it is instantaneously reset to v_r. Specific instances of the nonlinear integrate-and-fire model (1) are the quadratic model (Ermentrout and Kopell, 1986), (Hansel and Mato, 2001) and the exponential model (Fourcaud-Trocmé *et al.*, 2003).

Let us consider a discretization of the voltage state-space $V_i = [v_i, v_{i+1}[$ where $v_i = i\Delta v$ and Δv is a fixed voltage-step. The basic idea of our method is to approximate (1) by the voltage-dependent integrate-and-fire neuron

$$C\frac{dv}{dt} = -g_i(v - E_i) + I_0 + I_{syn}(t), \tag{2}$$

for $v \in V_i$. For clarity, the subscript k has been dropped. Parameter g_i is a voltage-dependent conductance and E_i is a voltage-dependent resting potential. Parameters g_i and E_i are such that the linear function $f_{\Delta v}(v) = -g_i(v - E_i)$ approximates the nonlinear characteristic $f(v)$ on V_i. For instance, using the approximation of f by the linear interpolation function at the boundaries of V_i gives

$$g_i = -\frac{f(v_{i+1}) - f(v_i)}{\Delta v},$$
$$E_i = v_i + \frac{f(v_i)}{g_i}.$$

In this case the voltage-dependence of the approximated IF parameters is piecewise linear.
Note that we keep the same notation for the membrane potential and its approximation (if ambiguity we will note $v_{\Delta v}$ the approximation of v). We note $V_{reset} =]-\infty, v_r]$ and $V_{th} = [v_{th}, +\infty[$ the resetting and threshold intervals, respectively.
Let t_0 the time at which the membrane potential of the neuron reaches V_i ($v(t_0) = i\Delta v$) and assume that the neuron stays in V_i during a non-empty time interval. Integrating (2) between t_0 and t yields

$$\begin{aligned} v(t) &= i\Delta v\ e^{-(t-t_0)/\tau_i} + (E_i + I_0/g_i)(1 - e^{-(t-t_0)/\tau_i}) \\ &\quad + \int_{t_0}^{t} e^{-(t-y)/\tau_i} \frac{I_{syn}(y)}{C} dy \end{aligned} \tag{3}$$

where $\tau_i = C/g_i$ is the voltage-dependent membrane time constant. The synaptic current I_{syn} is given by

$$I_{syn}(t) = w \sum_{t^f_{pre}} \alpha(t - t^f_{pre}) \tag{4}$$

where the t^f_{pre} are the firing times of the presynaptic neurons, w represents the weight of the synapse, and α is a given function that describes the post synaptic-potential. A common choice is $\alpha(t) = 1/\tau_s e^{-t/\tau_s} H(t)$ or $\alpha(t) = 1/(\tau_1 - \tau_2)(e^{-t/\tau_1} - e^{-t/\tau_2})H(t)$ where H is the Heaviside step function with $H(t) = 1$ if $t > 0$ and $H(t) = 0$ otherwise. Since I_{syn} is a combination of exponential functions the integral in (3) can be computed analytically and an event-driven method can be used to calculate the next *exit* time, t_1. Three possibilities occur: (i) the membrane potential goes back to its value at time t_0, i.e. interval V_{i-1} is reached (ii) the membrane potential reaches the interval V_{i+1}, i.e. $v(t_1) = v_{i+1}$ and (iii) the neuron is at rest, i.e. $t_1 = +\infty$.
If the spiking interval V_{th} is reached, then a firing event occurs, $t^f = t_1$ and the neuron is reset. The event-driven method is applied on a voltage-step and therefore our method may be seen as a *local* event-driven method.
Let $(t_k)_{k\in N}$ be the sequence of times at which the successive intervals $(V_i)_{i\in I}$ are reached. This sequence defines the integration points of an implicit variable time-step method. The numerical integration is reduced to the detection of the occurrence of discrete events that is achieved using symbolic computation or a Newton-Raphson algorithm (that is very efficient and only few

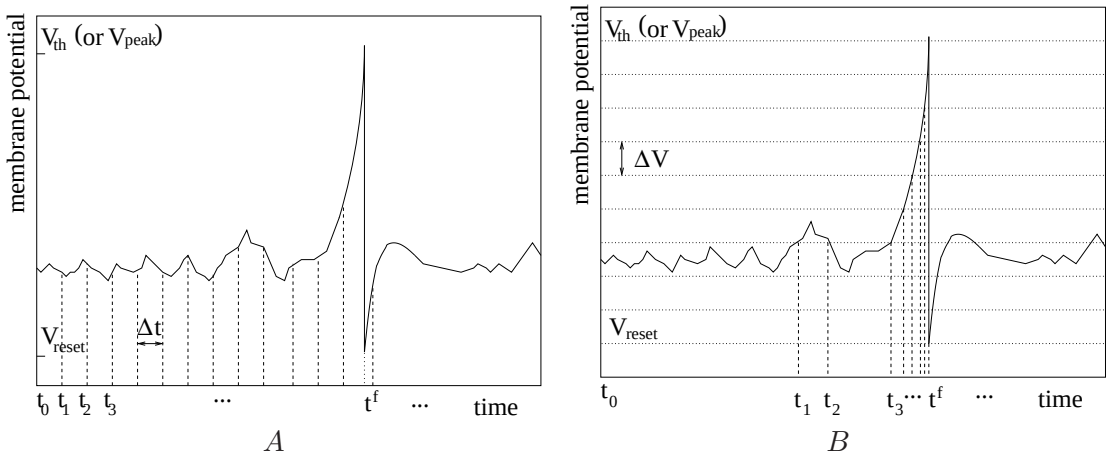

Figure 1: Schematic view of the numerical integration of an integrate-and-fire neuron using (A) a time-stepping scheme and (B) a voltage-stepping scheme. The time stepping method with a fixed time step Δt requires calculations at each step independently of the membrane potential fluctuations. The voltage-stepping approach induces an adaptive time-step leading to a precise approximation of the firing time.

iterations are needed). Symbolic derivation of the events is possible for constant input currents and for special cases of synaptic currents (Dirac synaptic currents). Otherwise Newton-Raphson algorithm has to be used. Note that an alternative method based on polynomial root finding algorithms could be used for exponential currents (Brette, 2007). Within each time interval, a symbolic expression of the membrane potential is given by (3). The result is schematically illustrated in Fig. 1B and compared with a fixed time-step method (Fig. 1A). Advantages are clearly seen. At the neuron level, when the membrane potential is slowly varying the corresponding time-steps are large whereas small time-steps are used when the membrane potential strongly fluctuates. Near the threshold, due to the nonlinear voltage-dependent current, the membrane voltage changes quickly leading to short time steps to accurately follow the trajectory. At the network level, the voltage stepping method presents some interesting properties. Firstly, the computational cost of the simulation is significantly reduced when the activity of the network is localized. Serial activation of areas, like propagating wave or synfire activity, frequently occur in neuronal tissue, notably the cortex, the thalamus and hippocampus (Foldiak and Young, 1995). Neurons that participate to the wave activity are excited while the others are at rest or poorly activated. Time steps are used to update excited neurons whereas no or little computation is done for the others. Secondly, the spike-spike interactions [2] which is usually ignored in modified Runge-Kutta schemes (Rangan and Cai, 2007) is naturally handled in our local event-driven strategy. Moreover unlike standard time-stepping scheme, the implicit time-discretization defined by our technique is different for each neuron in the network and only depends on its own membrane potential fluctuations.

The basic idea of the voltage-stepping approach is to define a local variable-step integrator using an approximation of the nonlinear characteristic f by a function that is amenable to an event-driven scheme. The piecewise linear interpolation of the nonlinear current $f(v)$ leads to an approximation of the original model by a voltage-dependent linear integrate-and-fire (LIF) neuron (2). As v evolves, parameters of the LIF change. This approach is reminiscent to the piecewise linear caricature of neuron models (McKean, 1970; Tonnelier and Gerstner, 2003). In

[2] interactions between spikes that are in the same time interval

general, as we show in the Appendix, the voltage-dependent LIF, defined using an interpolation of the nonlinear currents at the boundaries of V_i, leads to a numerical integration with an accuracy of $O(\Delta v^2)$ (Appendix 5.1). A clever choice of the interpolation points within a voltage-step leads to an error of order $O(\Delta v^4)$. However it should be noted (see Appendix 5.2) that the fourth order accuracy is only reached for the simulation of one-dimensional neuron models. A lower order scheme ($O(\Delta v)$) is obtained using a voltage-dependent non-leaky IF model ($f_{\Delta v}$ is piecewise constant). Similarly, a more accurate scheme is obtained using a piecewise quadratic approximation and the corresponding approximated model is a voltage-dependent QIF model ($f_{\Delta v}$ is piecewise quadratic). In this paper, numerical simulations are done using voltage-dependent LIF neurons to approximate the original neuron models.

2.2 Algorithm

Each neuron maintains a mode (i.e. its location within the discretized voltage space) and an exit time that is the time at which a new voltage interval is reached. The exit time becomes a spike timing whenever the neuron reaches the threshold interval.

- Initialization: Compute the events, i.e. the exit times of each neuron (including spike timings) and insert them in a priority queue.
- Process events. Extract the event to be processed, i.e. the one with the lowest timing. Note that local events do not have any consequence on the overall network dynamics and only require a local updating of the corresponding neuron mode and exit time. For a spike timing, the neuron is reset and the firing event is propagated, i.e. modes and exit times of the target neurons are updated.

Optimization of the code can be obtained using precalculated tables of the exit-time in each voltage-step without input. If spikes are received, these timings can be used as an initial guess for the iterative search algorithm.

2.3 Generalization

The voltage-stepping strategy is not limited to the simulation of one dimensional integrate-and-fire neurons. Below we sketch how to apply this technique for different widely used neural models.

A. **Integrate-and-fire neurons with adaptation.**
An improvement of the nonlinear integrate-and-fire model is achieved by adding a second variable to Eq. 1 (Izhikevich, 2003; Brette and Gerstner, 2005)

$$C\frac{dv}{dt} = f(v) - u + I_0 + I_{syn}(t) \quad (5)$$

$$\frac{du}{dt} = a(bv - u) \quad (6)$$

where u represents an adaptation current. At each firing time, the variable u is increased by an amount c ($u \leftarrow u + c$). As previously, we **can** derive a voltage-dependent LIF using a linear interpolation of f on V_i. The system is now two-dimensional but since the equations are linear on a voltage-step the approximated model can be solved analytically in this interval, then the local event-driven scheme applies similarly. Note that (i) nonlinear currents can be added in the adaptation equation provided that a piecewise linear approximation is used. (ii) Without resetting on v and without spike-triggered-adaptation (5)-(6) has the FitzHugh-Nagumo model as a special instance.

B. **Conductance based synaptic currents.**
More realistic descriptions of synaptic current incorporate conductance changes

$$I_{syn}(t) = g_{syn}(v_e - v) \sum_{t^f_{pre}} \alpha(t - t^f_{pre}) \tag{7}$$

where v_e is a reversal potential. Recently the event-driven simulation of LIF models has been extended to synaptic conductances (Brette, 2006). Thus we can adapt the voltage-stepping method to neural models with synaptic conductances using the method developed in (Brette, 2006) to compute the local events. However to reduce the computational cost of the simulation, it is useful to use the following approximation $v_e - v = v_e - v_i$ on the voltage-step $V_i = [v_i, v_i + \Delta v[$ and go back to the voltage-dependent LIF previously defined. In this case we introduce an error of order Δv but it is possible to restore the accuracy of the scheme rewriting the equation as

$$\begin{aligned} C\frac{dv}{dt} &= f(v) + I_0 + g(v_e - v) \\ \frac{dg}{dt} &= -g/\tau_s \end{aligned}$$

where an incoming spike triggers an instantaneous additive change $g \to g + w$. Nonlinearities appear both in the characteristic, f, and in the conductance, g. To achieve a local event-driven scheme, it is necessary to discretize not only the voltage space but the entire state-space (v, g). The method is similar to conductance-based models and is detailed below.

C. Hodgkin-Huxley type neurons.
Our approach is not limited to integrate-and-fire neuron models and can be used to simulate detailed neuron models including spike-description such as Hodgkin-Huxley type models. For simplicity, we consider the Morris-Lecar model

$$\begin{aligned} C\frac{dv}{dt} &= \bar{g}_{Ca} m_\infty(v)(v_{Ca} - v) + \bar{g}_K u(v_K - v) + \bar{g}_l(v_l - v) + I \\ \frac{du}{dt} &= \frac{u_\infty(v) - u}{\tau(v)} \end{aligned}$$

that describes an instantaneously responding voltage-sensitive Ca2+ conductance for excitation and a delayed voltage-dependent K^+ conductance for recovery (see (Rinzel and Ermentrout, 1998) for a complete definition). Since the nonlinearity involves both v and u, it is necessary to discretize the entire state space (v, u) (the term state-stepping is more appropriate in this case). The state space is partitioned into subdomains where the function is approximated by a linear system $F_i(X) = A_i X + b_i$ where $X = (v, u)$. The shape of subdomains is triangular and a simplicial partition (i.e. triangulation based on a rectangular partition) can be used (Girard, 2002). In each triangle the approximated neuron model has a symbolic expression from which we calculate the switching time, i.e. the time at which a new triangle is reached. In Fig. 2 we show the result of the numerical integration. The method performs like an adaptive time-stepping scheme. The neuron is updated frequently when one state variable has large variation, specially near the threshold and at the peak of the spike where the membrane voltage present abrupt polarization. Note that the integration points are not only determined by the membrane potential but also by the recovery variable. The fastest changing variable imposes the time-steps of the neuron.

The algorithm can be extended to general n-dimensional conductance-based neuron using a simplicial subdivision of the neuron state-space and interpolating the vector field at the vertices of the simplex.

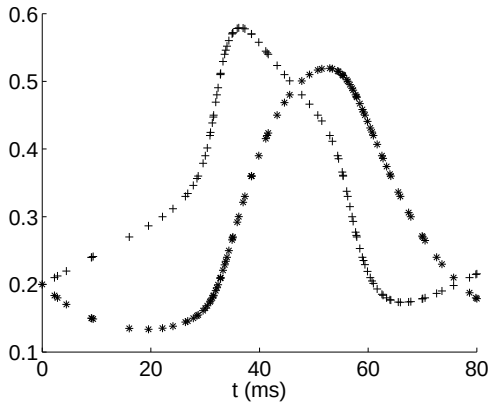

Figure 2: Numerical simulation of the Morris-Lecar model with a state-stepping algorithm. We simulate the model over one period of its oscillatory regime. Crosses are the membrane potential and stars are for the potassium channel. We use a voltage-step of $6mV$ and a step of 0.03 for the recovery variable. For convenience the membrane potential has been rescaled (vertical axis is dimensionless).

2.4 Link with previous works

Traditional event-driven methods are actually spike-driven schemes; the neuron state variables are updated when a spike is received or emitted. The voltage-stepping technique produces new events: the events are not only firing times or spike receptions but also the times of mode switching, i.e. when the neuron reaches a new voltage interval. Consequently the number of events is increased, albeit at a reduced computational cost because local events do not have any repercussion on the overall activity of the network and are only used to update the corresponding neuron parameters. When the cost of managing events become prohibitive, it could be useful to introduce local event queues to reduce the cost of event management (Morrison *et al.*, 2007).
The voltage-stepping scheme combines event-driven and discretization techniques. A method based on a combination of event-driven and time-driven schemes has been recently developed but requires a minimal synaptic propagation delay greater than the computation time step (Morrison *et al.*, 2007). Fast methods have been developed but are limited to the linear IF neuron (Rangan and Cai, 2007). To simulate more realistic neural models, an alternative approach has been proposed by (Ros *et al.*, 2006) in which an event-driven scheme uses lookup tables. The simulation is reduced to a search within a precalculated table of function values. This scheme combines the benefits of using realistic neural models and high-speed simulations but becomes cumbersome to manage when a good accuracy is required for the numerical simulation.
Adaptive time-stepping schemes provide short time-step integration for active neurons and long-time step integration when the neuron is at rest or slightly activated. These methods have a clear advantage when the entire shape of the spike is calculated. When simulating neural networks, classical variable-step integrators fail to be efficient since the fastest changing neuron imposes the time-discretization for the entire network. To avoid this problem, Lytton and Hines (2005) have used an independent variable time-step integrator for each neuron. A critical problem is to coordinate the local integrators of each neuron in the network and to properly handle the events. Indeed, when an event arrives at a neuron at time t_e it is necessary to have all the states of the receiving neurons at time t_e. This is accomplished using additional operations: a fixed-step integration, interpolation and reinitialization. Moreover the integration coordinator must ensure that the individual time-steps are correctly chosen:there is always an overlap between all the integration intervals of neurons. In our approach both local and global events are

nicely handled in an event-driven scheme and there is no requirement on the time-steps. The variable time-steps induced by the voltage-stepping scheme present interesting properties: (i) by construction the threshold event lies on an integration time-step boundary. (ii) Integration points are independent, i.e. in network simulation, the time steps are different from one neuron to another. (iii) Time-steps are imposed by the voltage-trajectory and when the neuron is at rest, no step is computed.
Approximation by piecewise linear systems has become a classical tool for the global qualitative analysis of dynamical systems and has been proposed as a technique for numerical simulations (Girard, 2002). The voltage-stepping method is a variant of the hybrid computation method where the simulation is done by using an approximation of the vector field (Della-Dora *et al.*, 2001). The hybrid computation requires a full discretization of the state space that appears to be prohibitive for large dynamical systems. Taking advantage of point-like interactions between neurons this scheme could be efficiently implemented for the simulation of spiking networks. Following the hybrid-system framework (differential equations with discrete events), we interpret spiking neural networks as an hybrid system where global events are spikes and local events are mode switches. During simulation the neuron switches between modes depending on the value of the voltage, i.e. parameters of the model change when a mode transition is detected.

3 Numerical Results

To illustrate our numerical scheme we consider the quadratic integrate-and-fire (QIF) model. The QIF model includes nonlinear spike generating current and represents the normal form of any type I neuron model near the bifurcation (Ermentrout, 1996; Ermentrout and Kopell, 1986). It is widely used as a realistic neural model (Brunel and Latham, 2003; Fourcaud-Trocmé *et al.*, 2003; Hansel and Mato, 2001). The dynamics of the QIF model is described by

$$\tau \frac{dv}{dt} = v^2 + I_0 + I_{syn}(t) \tag{8}$$

where v, I_0 and I_{syn} are dimensionless membrane potential, input current and synaptic current, respectively. Parameter τ is the membrane time constant. We treat synaptic current of the form

$$\begin{aligned} I_{syn}(t) &= w\ exp(-(t-t^f)/\tau_s), \quad t \geq t^f \\ &= 0, \quad t < t_f \end{aligned} \tag{9}$$

that could be rewritten as

$$\begin{aligned} \frac{dI_{syn}}{dt} &= -I_{syn}/\tau_s, \\ I_{syn} &\leftarrow I_{syn} + w \quad when\ t = t^f \end{aligned}$$

where τ_s is the synaptic time constant and w the synaptic weight. Numerical values of QIF neurons are taken from (Martinez, 2005), the membrane time constant is $\tau = 0.25ms$, the reset potential is $v_r = -0.0749$ and the threshold is $v_{th} = 0.7288$. The synaptic time constant is $\tau_s = 6ms$ and the synaptic strength is $w = 5 \times 10^{-4}$. Note that the membrane potential, in voltage unit, is obtained using the variable change $V \leftarrow C/qv + V_0$ where $C = 0.2nF$, $q = 0.00643mS.V^{-1}$ and $V_0 = -60.68mV$. Therefore a factor of $C/q = 31.1mV$ has to be applied to the voltage step Δv to retrieve the physical unit.
One of the goals of our numerical scheme is to simulate accurately the dynamics of spiking neurons and to investigate temporal coding properties. Therefore we are interested in reproducing the exact timing of spikes and we use the following measure of error:

$$E(t^f) = \frac{1}{N} \sum_f |t^f_{\text{ex}} - t^f_{\text{ap}}| \tag{10}$$

where N is the number of spikes, t^f_{ex} are the exact firing times (i.e. with an arbitrary precision) and t^f_{ap} are the corresponding approximated firing times that depends on Δv for voltage-stepping methods or on Δt for time-stepping methods (see below). Using (10) we implicitly assume, as a minimal requirement, that the number of spikes between the exact and the approximated spike trains does not differ. This is achieved using a sufficiently accurate numerical scheme, i.e. the steps are small enough to capture all the spikes. When the number of spikes may differ, we use the following error

$$E(\nu) \quad = \quad |\nu_{\text{ex}} - \nu_{\text{ap}}| \tag{11}$$

where ν_{ex} and ν_{ap} is the exact and approximated firing rate, respectively.
We have shown previously that the QIF model is amenable to an exact simulation (Tonnelier *et al.*, 2007) that we will use for t^f_{ex} in the error analysis (here the precision is fixed at $10^{-7}ms$ on individual spike times). We focus our numerical study to the voltage-stepping scheme with piecewise linear approximation. Parameters of the approximated voltage-dependent LIF, using an interpolation at the boundaries of V_i (we will call VS2 scheme), are $g_i = -\Delta v(2i+1)$ and $E_i = \Delta v\ i(i+1)/(2i+1)$ where Δv is a fixed voltage-step. We also consider the voltage-stepping scheme using a linear interpolation at gaussian abscissas (hereafter VS4 scheme).
Since fixed time-step integration remains the simulation standard, we compare the performance of our schemes (VS2 and VS4) to the one of the 'corresponding' time-stepping algorithms. The second or fourth order Runge-Kutta scheme with a linear or cubic interpolation of firing times (i.e. modified RK2 or modified RK4) has been shown to be a second or fourth order scheme, respectively, in the simulation of spiking neurons (Hansel *et al.*, 1998; Shelley and Tao, 2001). The RK schemes are monitored by a fixed time-step Δt that controls the error-functions (10), (11) of the methods. It is important to notice that several problems may arise when comparing the different methods: i) the computation time of voltage-stepping schemes is implementation-dependent and many algorithmic tricks could speed up the calculations[3], ii) the accuracy and computation time of voltage-stepping schemes is activity-dependent (in the extreme case where no spike is emitted the computational cost is negligible for our method whereas Runge-Kutta schemes perform all the steps to reach a predetermined simulation time). We overcome issue i) by comparing straight-forward implementations of each method and issue ii) by using different inputs leading to different spiking activities. Voltage-stepping and time-stepping methods are applied to simulate a single neuron with a constant input current and a Poisson input spike train. The accuracy of the voltage-stepping approach is also demonstrated on the simulation of a network of spiking neurons. The algorithms were programmed with Matlab and the numerical simulations are performed on a portable PC running Windows at 1.7GHz.

3.1 Constant input current

We ignore the synaptic current, $I_{syn} = 0$, and consider a QIF neuron with a constant external driving current I_0. For $I_0 < 0$ there is a pair of equilibrium points. One is stable and the other is an unstable fixed point above which a spike is emitted. The neuron is said to be excitable. For $I_0 > 0$ the neuron fires regularly. The neuron is said to be in the oscillating regime. We quantify the accuracy by calculating the error $E(t^f)$ (Eq. 10) and $E(\nu)$ (Eq. 11) in the excitable and oscillating regime, respectively.

[3]In particular, since our approach can be seen as a local event-driven technique, all the recently proposed methods to optimize event-driven schemes could be tested and used.

A. Excitable regime

Let $I_0 < 0$ and consider an initial condition $v(0)$ slightly above the unstable fixed point $\sqrt{-I_0}$. Without further input, the membrane potential is given by

$$v(t) = -\sqrt{-I_0}\coth(\sqrt{-I_0}t/\tau - \operatorname{atanh}(\sqrt{-I_0}/v(0))),$$

and the exact spike timing is given by

$$t^f_{\text{ex}} = \tau/\sqrt{-I_0}\left(\operatorname{atanh}(\sqrt{-I_0}/v(0)) - \operatorname{atanh}(\sqrt{-I_0}/v_{peak})\right).$$

We compute error $E(t^f)$ (Eq. 10) between the exact firing time given above and the approximated one, numerically obtained using the voltage-stepping method VS2 with $\Delta v = 0.005$ and VS4 with $\Delta v = 0.01$ ($\Delta v = 0.15mV, 0.31mV$ in physical unit). To assess the performance of the method, we compare it to the modified RK2 and RK4 respectively with $\Delta t = 0.03ms$ and $\Delta t = 0.2ms$ that could be considered as a very high temporal resolution but are required by the modified Runge-Kutta methods in order to calculate each neuronal trajectory accurately (usually smaller than $0.02ms$ to reach a good accuracy (Shelley and Tao, 2001; Rangan and Cai, 2007)). The precise values of Δt are chosen here in order to obtain a mean cost similar to the one obtained for the voltage-stepping scheme. Errors on the firing time as a function of the initial voltage value $v(0)$ are depicted in Fig.3. The slow decrease of the error for VS methods

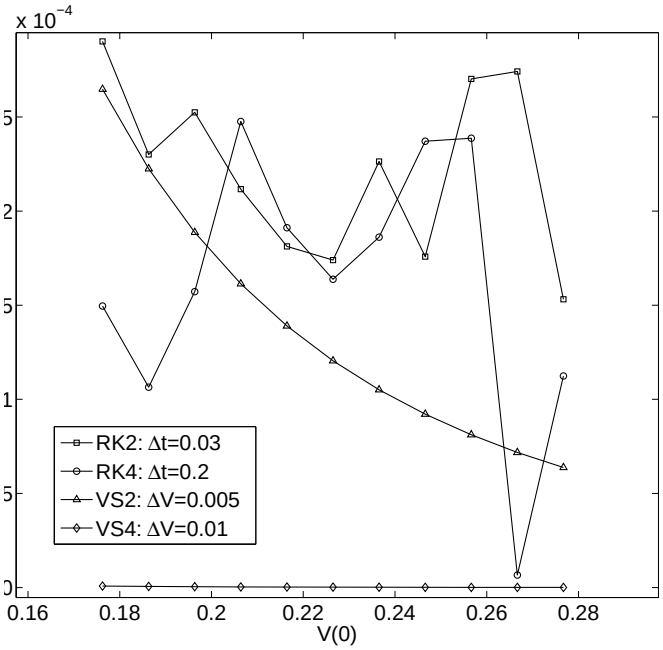

Figure 3: Error $E(t^f)$ (in ms) on the firing time of the QIF neuron as a function of $v(0)$ for different algorithms. Squares and circles are the modified RK2 with $\Delta t = 0.03ms$ and RK4 with $\Delta t = 0.2ms$, triangles and diamonds are VS2 with $\Delta v = 0.005$ and VS4 with $\Delta v = 0.01$ respectively.

with respect to $v(0)$ is a consequence of the activity-dependent accuracy of the method. As $v(0)$ increases, the number of voltage-steps necessary to reach the voltage peak decreases and thus the accumulating error. The RK methods are less robust and we suspect that the oscillating behavior observed in Fig.3 is related to the uniform and non-optimized distribution of

	RK2	RK4	VS2	VS4
Mean error (μs)	0.245	0.177	0.129	3×10^{-4}
Mean time cost (ms)	0.8878	0.7705	0.7681	0.7659

Table 1: Mean error and time cost for the numerical schemes used in Fig. 3.

the time-steps. As $v(0)$ increases the firing time occurs faster and the number of time intervals needed to reach firing time gets smaller. As a consequence, one would expect a reduction of the error. However, for RK methods, the same discretization of time is used independently of $v(0)$. As $v(0)$ increases, the firing time occurs at different positions within a time step leading to different accuracies that mask the expected reduction of the error. Such oscillating phenomenon does not occur using the voltage-stepping approach since the time-steps are implicitly defined and are adjusted to the activity profile of the neuron. The time cost of the different algorithms applied in Fig.3 are given in Table 1. Accuracy and time cost are computed as the mean over the different initial values $v(0)$ taken in Fig.3. Since the mean computation time is approximately the same for all numerical schemes, a direct comparison of the methods is done observing the produced errors. In the following, a further comparative analysis of the efficiency is done.
To have a more practical view of the efficiency of the schemes, we compare in Fig.4 the computation times of each method as a function of the accuracy.

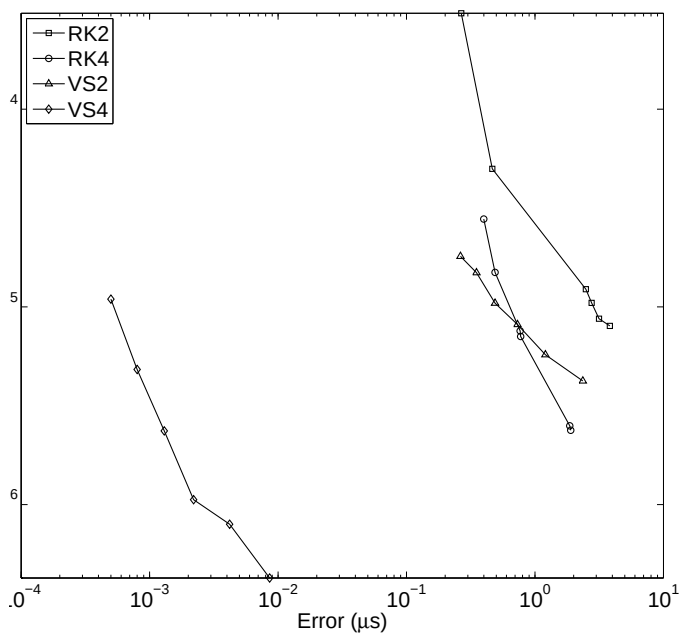

Figure 4: Log-Log plot of the time cost (ms) for both modified Runge-Kutta methods (RK2 and RK4) and voltage-stepping methods (VS2 and VS4) as a function of the error.

Results show that, in terms of efficiency, VS2 is superior to the modified RK2 method and comparable to the modified RK4 method. For finer resolutions, the VS2 algorithm performs faster than RK4 whereas for an error above approximately $1\mu s$ the RK4 method is more efficient. In all cases, the VS4 method is always more efficient. A major reason of the lack of efficiency of the Runge-Kutta scheme is that the uniform distribution of time steps implies the use of unnecessary time steps for subthreshold voltage in order to have short time steps at the peak of

RK2: $\Delta t(ms)$	0.2000	0.1667	0.1333	0.1000	0.0667	0.0333
Error (μs)	3.8257	3.1537	2.7606	2.4888	0.4650	0.2668
Time cost (ms)	0.3093	0.3119	0.3177	0.3228	0.3714	0.4452
RK4: $\Delta t(ms)$	0.5000	0.4500	0.4000	0.3500	0.3000	0.2500
Error (μs)	1.8993	1.8645	0.7734	0.7659	0.4890	0.3992
Time cost (ms)	0.2739	0.2753	0.3055	0.3075	0.3292	0.3503
VS2: Δv	0.0101	0.0072	0.0056	0.0046	0.0039	0.0034
Error (μs)	2.3573	1.2058	0.7302	0.4891	0.3503	0.2631
Time cost (ms)	0.2901	0.2991	0.3098	0.3176	0.3291	0.3354
VS4: Δv	0.0201	0.0182	0.0168	0.0155	0.0144	0.0134
Error (μs)	0.0086	0.0042	0.0022	0.0013	0.0008	0.0005
Time cost (ms)	0.2306	0.2455	0.2526	0.2737	0.2940	0.3191

Table 2: Error and time cost for the different algorithms and parameters used in Fig. 4.

the spike.
We also examine the errors of VS2 and VS4 schemes as a function of the voltage-step (Fig. 5). As expected, the errors of our VS2 scheme decreases quadratically with Δv while the error of the VS4 scheme diminishes as the fourth power of Δv.

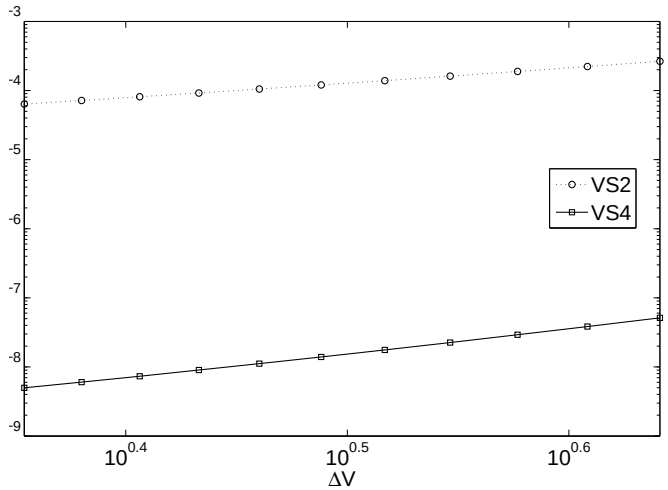

Figure 5: Log-Log plot of the error on the first spike timing as a function of the voltage-step. Circles and squares are voltage-stepping methods with a piecewise linear interpolation at the boundaries of the voltage-interval (VS2) and at the gaussian abscissas (VS4), respectively. The lines (not fits) indicate the order of the methods. Dotted-line is second order and solid-line is fourth order.

B. Oscillating regime

For $I_0 > 0$ the neuron fires regularly and the firing rate is given by

$$\nu_{\text{ex}} = \frac{\sqrt{I_0}}{\tau\left(\arctan\frac{\vartheta}{\sqrt{I_0}} - \arctan\frac{v_r}{\sqrt{I_0}}\right)} \tag{12}$$

In Fig. 6, we compare the error $E(\nu)$ (Eq. (11)) obtained by our voltage-stepping methods (VS2 and VS4) and the modified Runge-Kutta methods (RK2 and RK4) for different step-sizes. Time-steps are those usually used for modified Runge-Kutta schemes (of order $0.01ms$) (Shelley and Tao, 2001; Rangan and Cai, 2007). The voltage steps are chosen in order to reach a similar mean time cost. As previously, the VS method is more robust in the sense that changing the input parameter I_0 does not affect significantly the accuracy. The error and the time cost computed as the mean over the different input currents used in Fig. 6 are computed in Table 3. Results obtained on the error $E(\nu)$ are in line with those obtained with $E(t^f)$. The VS4 method is the most efficient algorithm and reaches the best accuracy with the lowest computational time.

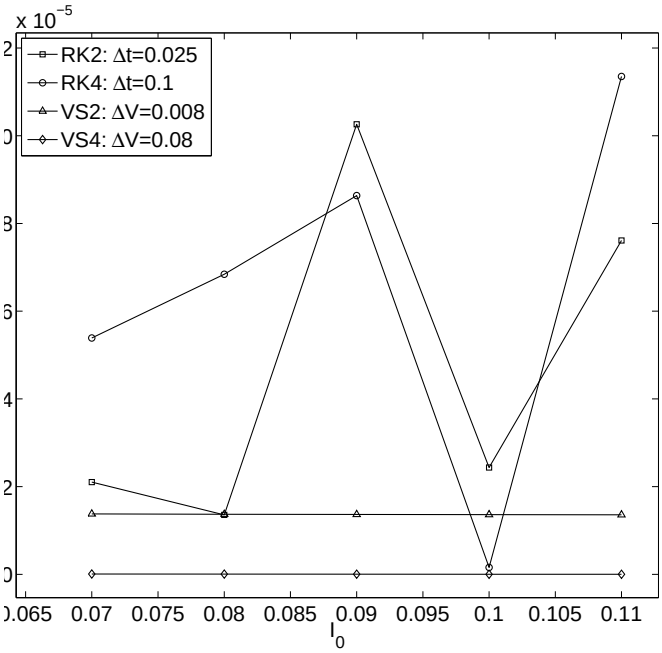

Figure 6: Error $E(\nu)$ (in Hz) on the firing rate of the QIF neuron as a function of the input current. The squares and circles correspond to the modified RK2 with $\Delta t = 0.025$ ms and the modified RK4 with $\Delta t = 0.1$ ms, respectively. Triangles represent VS2 with $\Delta v = 0.008$ and diamonds VS4 with $\Delta v = 0.08$.

	RK2	RK4	VS2	VS4
Mean error (Hz)	0.0475	0.0647	0.0137	3×10^{-5}
Mean time cost (ms)	0.9766	0.7542	0.6034	0.4423

Table 3: Mean error and time cost for the different algorithms with parameters used in Fig. 6.

Again, the order of the voltage-stepping methods VS2 and VS4 is numerically evaluated in Fig. 7 and corroborates the result previously obtained.

3.2 Poisson input spike train

We consider a simple test neuron receiving a synaptic activity modeled by a fluctuating spike train. Note that we use a current injection and a more realistic input scenario would be stochastic conductance changes. However a random conductance scenario can be replaced, to a high

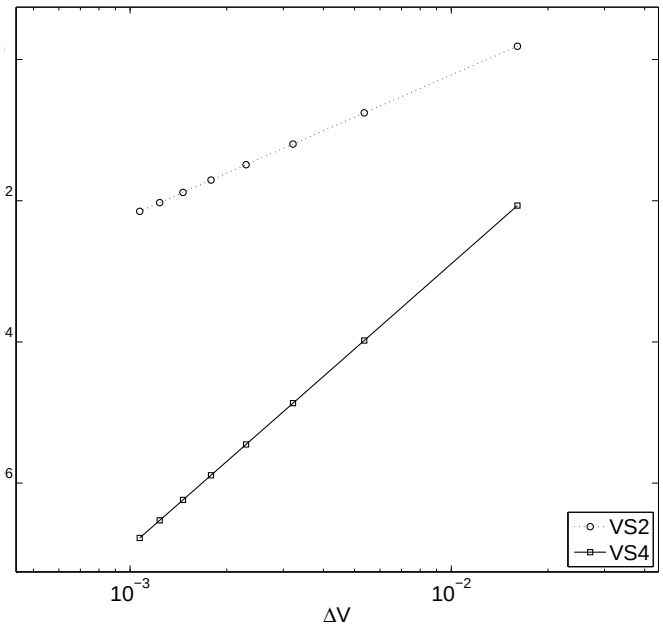

Figure 7: Log-Log plot of the error $E(\nu)$ on the firing rate as a function of the voltage-step for the voltage-stepping methods VS2 (circle) and VS4 (square). The lines (not fits) indicate the order of the methods. Dotted-line is second order and solid-line is fourth order.

degree of accuracy, by a random current injection (Richardson, 2004). For clarity, we keep the current injection paradigm. Firstly, we investigate two scenarios that reproduce two different regimes of neural activity. Secondly, we study the dependence on the input rate.
In the first scenario, we use a fixed excitatory spike train generated by a Poisson process with rate $\nu_E = 10^4 spikes/s$ that models the interaction with 1000 excitatory presynaptic neurons firing at 10 Hz. In this scenario, the neuron operates in a high activity regime with a firing rate $\sim 400 spikes/s$. The second scenario is described by two fluctuating synaptic currents, one excitatory Poisson process ($\nu_E = 10^4 spikes/s$) and one inhibitory Poisson process ($\nu_I = 10^4 spikes/s$). The neuron operates in a fluctuation-driven regime with a moderate firing rate ($\sim 30 spikes/s$). We compute the error on the spike timings of the QIF model using the voltage-stepping methods (VS2 and VS4) and the Runge-Kutta methods (RK2 and RK4). In order to compare the different numerical schemes, the voltage-steps and the time-steps are chosen so that to obtain a corresponding error. Simulation times are then compared. In the first scenario the neuron fires regularly with monotonic subthreshold membrane voltage trajectories. For different values of the time-steps and voltage-steps we compute the error and the time cost of the different schemes. Results are depicted in Fig 8, and the corresponding data are given in Table 4.
In this scenario, we found that the VS2 scheme is more efficient than the modified RK2 but better results are obtained using the RK4 scheme at coarse resolutions (see Table 4). However when a high accuracy is required (error in the order of $10\mu s$), the time cost of the time-stepping scheme significantly increases and VS2 becomes faster. Again, the VS4 method outperforms the other schemes. The possible reason for the discrepancy of the modified RK schemes for high accuracy is due to the non-smooth dynamic of the synaptic-induced changes given by (9). More precisely, for time-stepping schemes, when a spike is emitted by the neuron (at time t^f), the additional postsynaptic changes after the spike times are neglected (until a new step $t_{n+1} > t^f$

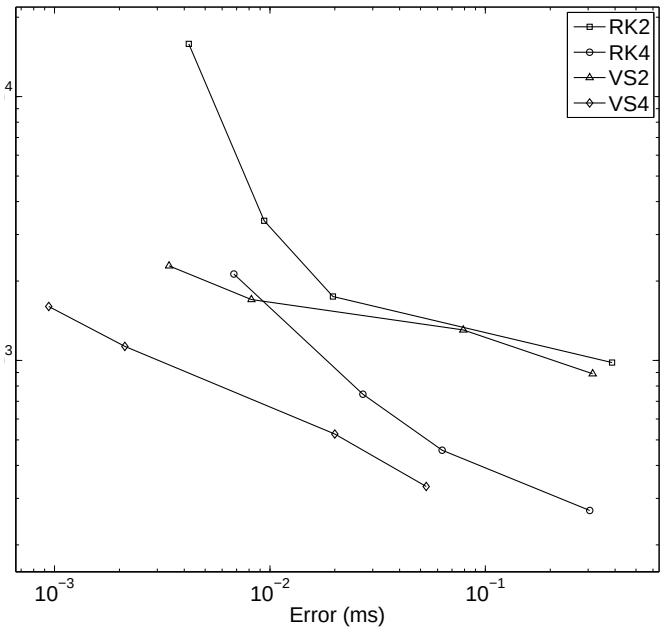

Figure 8: Log-Log plot of the time cost (s) for the modified Runge-Kutta methods (RK2 and RK4) and voltage-stepping methods (VS2 and VS4) as a function of the accuracy in the high activity regime.

RK2: Δt (ms)	0.0200	0.0100	0.0050	0.0010
Error (ms)	0.3874	0.0196	0.0094	0.0042
Time cost (s)	982	1747	3383	15845
RK4: $\Delta t(ms)$	0.1000	0.0500	0.0300	0.0100
Error (ms)	0.3051	0.0633	0.0272	0.0068
Time cost (s)	270	457	745	2126
VS2: Δv	0.0161	0.0080	0.0054	0.0040
Error (ms)	0.3151	0.0794	0.0082	0.0034
Time cost (s)	890	1305	1702	2283
VS4: Δv	0.0179	0.0124	0.0095	0.0077
Error (ms)	0.0532	0.0200	0.0021	0.0009
Time cost (s)	333	526	1131	1602

Table 4: Error and time cost for the different algorithms and parameters used in Fig. 8 (high activity regime).

is reached). For post-synaptic changes with non-smooth initiation (like (9)) the error becomes relevant and we suspect that the modified RK scheme behaves like a first-order scheme and therefore requires small time-steps to accurately computes spike times. This problem does not occur in voltage-stepping scheme since the step is automatically adjusted to the computed firing time.

In the second scenario, the membrane potential is driven by the balance of excitation and inhibition leading to irregular spike-times. For a fixed voltage-step value, the accuracy of the voltage-stepping is affected (see Table 5). The efficiency of Runge-Kutta type schemes dra-

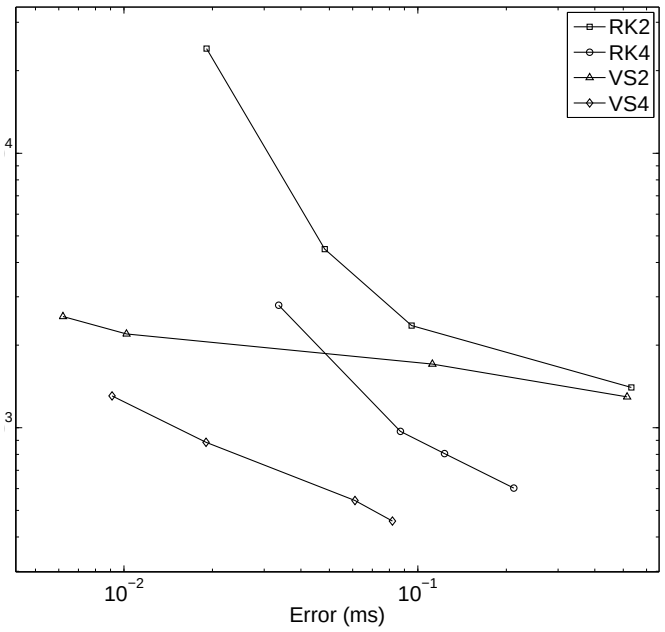

Figure 9: Log-Log plot of the time cost (s) for both the modified Runge-Kutta methods (RK2 and RK4) and voltage-stepping (VS2 and VS4) as a function of the accuracy for the balanced regime.

matically decreases (mainly for the RK2 scheme) and small time-steps are necessary to compute accurately the spike times. The respective efficiency of the schemes is similar to the one obtained in the high activity regimes: VS4 has the better performance and RK2 the worst. The efficiency of VS2 versus RK4 depends on the level of required accuracy.
Finally, we investigate the dependence on the rate of the input spike train. We consider the first scenario using different values of the input firing rate $\nu_E = 10^4$ $spikes/s$, 10^3 $spikes/s$, 500 $spikes/s$ and 333 $spikes/s$. For a given level of accuracy (here approximatively $0.1\mu s$) the computation times are computed for the different numerical schemes (see Fig. 10 and Table 6). For all schemes an increase in time cost is observed as the activity becomes higher. In the high activity regimes, the computation time of Runge-Kutta algorithms dramatically increases. Moreover for all input firing rate the relative efficiency of the methods is the same as the one previously obtained. Note that since high accuracy is required the VS2 scheme is more efficient than the modified RK4 scheme.

3.3 Network activity

We demonstrate the accuracy of our integration scheme by applying it to a network of $N = 100$ inhibitory neurons. Such network shows rapid synchronization through mutual inhibitions and variations of this model have been widely studied (Wang and Buzsaki, 1996; Martinez, 2005; Ambard and Martinez, 2006). We consider all-to-all coupling between inhibitory neurons with a synaptic strength $w = 0.005$. Each inhibitory neuron is driven by a pre-synaptic excitatory spike train (Poisson process, $\nu_E = 10^4 spikes/s$) with a synaptic weight $w = 0.005$. The neurons are modeled as QIF with the same parameters as before. We integrate until $t = 40ms$ and use the VS2 method with a subdivision of the voltage-space into $N_{\Delta v} = 250$ voltage-intervals. For error analysis we simulate the network using an exact event-driven simulation (Tonnelier *et*

RK2: $\Delta t(ms)$	0.0200	0.0100	0.0050	0.0010
Error (ms)	0.5327	0.0952	0.0482	0.0191
Time cost (s)	1402	2356	4476	24055
RK4: $\Delta t(ms)$	0.1000	0.0500	0.0300	0.0100
Error (ms)	0.2123	0.1232	0.0872	0.0336
Time cost (s)	603	805	970	2795
VS2: Δv	0.0161	0.0080	0.0054	0.0040
Error (ms)	0.5152	0.1112	0.0102	0.0062
Time cost (s)	1296	1706	2196	2543
VS4: Δv	0.0179	0.0124	0.0095	0.0077
Error (ms)	0.0819	0.0611	0.0190	0.0091
Time cost (s)	457	543	885	1306

Table 5: Error and time cost for the different algorithms and parameters used in Fig. 9 (balanced regime).

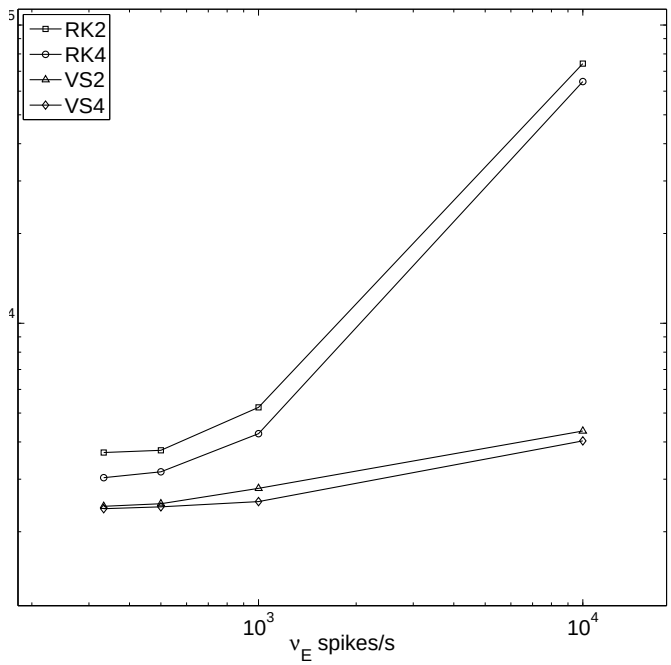

Figure 10: Log-Log plot of the time cost (s) for the modified Runge-Kutta methods (RK2 and RK4) and voltage-stepping methods (VS2 and VS4) as a function of the spikes rate in the high activity regimes.

al., 2007). The exact and approximated spike times are shown in Fig.11. The network produces 344 spikes. Here, an accuracy of $0.22\mu s$ on individual spike-time is reached.

4 Conclusions

Recent efforts have been made to simulate integrate-and-fire neuronal networks. Specific methods like event-driven schemes (Makino, 2003; Brette, 2006; Brette, 2007), fast methods (Rangan

		$\nu_E = 333 spikes/s$		
	RK2	RK4	VS2	VS4
Eerror (μs)	0.1003	0.1044	0.0959	0.0939
Time cost (s)	3687	3035	2434	2389
		$\nu_E = 500 spikes/s$		
	RK2	RK4	VS2	VS4
Error (μs)	0.1069	0.1027	0.1047	0.0990
Time cost (s)	3749	3177	2482	2423
		$\nu_E = 10^3 spikes/s$		
	RK2	RK4	VS2	VS4
Error (μs)	0.0961	0.0959	0.0955	0.1012
Time cost (s)	5224	4264	2795	2523
		$\nu_E = 10^4 spikes/s$		
	RK2	RK4	VS2	VS4
Error (μs)	0.1032	0.1075	0.0987	0.0964
Time cost (s)	74235	64635	4353	4032

Table 6: Mean error and time cost for the different algorithms and spikes rate used in Fig. 10.

and Cai, 2007) or exact time-stepping schemes (Morrison *et al.*, 2007) are limited to linear integrate-and-fire models. Voltage-stepping methods are generic numerical schemes that realize an efficient and accurate numerical integration of spiking neural networks. Important elements in our approach are (i) the variable time-steps that are different for each neuron in the network depending on their activity (ii) the treatment of the possible discontinuities of the dynamics (iii) the event-driven nature of the simulation.

In this paper we have mainly addressed the single neuron case even if we have shown that network simulation could benefit from the voltage-stepping integration scheme. We expect that the superiority of the method over traditional time-stepping schemes observed for one neuron will be more patent in network simulations. In fact, it frequently appears that in large network simulation some area are quiescent and relatively few neurons are activated. Since our method only handles active neurons, a speed up in simulation time is expected. Our approach forms the basis of further studies on numerical methods with an emphasis on computation time. The recent advances made on event-driven techniques could be adapted to the voltage-stepping scheme. Critical points are the management of the event queue and the efficiency of the zero search algorithm. The local event queues employed in (Morrison *et al.*, 2007) could be combined with the voltage-stepping algorithm to reduce the cost of event management. For the second point, improvement could be done in several ways: i) an optimized algorithm devoted to the function considered here, ii) a 'good' initial guess using an a priori prediction of the exit-time. This estimation could be done using precalculated table or using more elaborate approximation techniques. Our voltage-stepping method is not necessarily restricted to a uniform voltage step Δv. There exist efficient algorithms (Breiman, 1993) that can be used to optimize both the non-uniform distribution of intervals V_i and their associated linear approximations. Therefore, a possible extension of our approach is to use a voltage-discretization adapted to the nonlinear voltage-dependent current of the model.

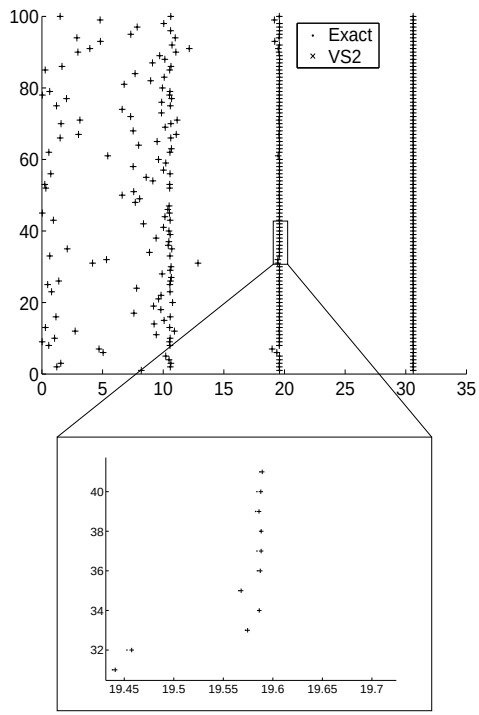

Figure 11: Simulation of a network of 100 inhibitory neurons. Spike times are computed exactly (dots) and with the VS2 method using $N_{\Delta v} = 250$ voltage-intervals ('+'). A high degree of accuracy is obtained and spikes are superimposed most of time (see enlargement).

5 Appendix: Order of voltage-stepping schemes

For simplicity, we consider a general neuron model described as follows (notations are defined in Sections 2 and 3)

$$\frac{dv(t)}{dt} = f\left(v(t)\right) \tag{13}$$

Over V_i, one possible linear interpolation of the function $f(v)$ is achieved by interpolating it at the boundaries v_i and v_{i+1}. Other possible linear interpolations will be discussed in the second part of this appendix.

5.1 Linear interpolation at boundaries (VS2 method)

Let us consider $v_{\Delta v}$ the solution of the following dynamical system:

$$\frac{dv}{dt} = f_{\Delta v}(v), \tag{14}$$

where $f_{\Delta v}$ is the piecewise linear function defined as follows:

$$f_{\Delta v}(v) = \frac{v_{i+1} - v}{\Delta v} f\left(v_i\right) + \frac{v - v_i}{\Delta v} f\left(v_{i+1}\right)$$

It is straightforward to show that the error of approximation is given by

$$\left|f\left(v\right) - f_{\Delta v}\left(v\right)\right| = O\left(\Delta v^2\right) \tag{15}$$

Considering the following theorem:

Theorem 1 *Fundamental Inequality (see for instance (Hubbard and West, 1991)). For a differential equation $\dot{x} = F(x)$ satisfying the Lipschitz condition with $K \neq 0$ and if $u_1(t)$ and $u_2(t)$ are two continuous, piecewise differentiable functions satisfying $|\dot{u}_i(t) - F(u_i(t))| \leq \varepsilon_i$ for all t at which $u_1(t)$ and $u_2(t)$ are differentiable and if $|u_1(0) - u_2(0)| \leq \delta$, then*

$$|u_1(t) - u_2(t)| \leq \delta e^{K|t|} + \frac{\varepsilon_1 + \varepsilon_2}{K}\left(e^{K|t|} - 1\right).$$

Applying Theorem 1 to (13)-(14) and using (15), it can be proved that $|v - v_{\Delta v}| = O\left(\Delta v^2\right)$. It follows $\left|t_{ex}^f - t_{ap}^f\right| = O\left(\Delta v^2\right)$ that means that the estimate error on the exact spike time is of order $O\left(\Delta v^2\right)$.

Remarks:

- At the neural network level, the incoming spikes generated by presynaptic neurons introduced a second order error (since $\left|t_{ex}^f - t_{ap}^f\right| = O(\Delta v^2)$). Noting the fact that $f_{\Delta v}$ also introduced a second-order error, the proposed voltage-stepping scheme (VS2) guarantees the same accuracy at the network level as the neuron level, even after considering the effect of propagation of error on spike times.
- For the p-dimensional case, the only difference is to approximate $f(v)$ over $V_i \subseteq R^p$ by a linear system: $f_{\Delta v}(v) = A_i v + b_i$, which is uniquely determined as the linear interpolation vector field of $f(v)$. The same result can be proved using a norm on $\mathcal{R}^p$, $\|\cdot\|$, instead of the absolute value $|\cdot|$ (see (Girard, 2002) for more details).

5.2 Linear interpolation at gaussian abscissas (VS4 method)

Let us consider (13) and (14) over a voltage interval V_i. Without loss of generality, assume that the voltage interval V_{i+1} is reached. We have

$$\Delta t_i = t_{ex}^i - t_{ap}^i = \int_{v_i}^{v_{i+1}} \left(\frac{1}{f(v)} - \frac{1}{f_{\Delta v}(v)}\right) dv \tag{16}$$

where t_{ex}^i represents the exact exit time of V_i, and t_{ap}^i represents its approximation. The best choice for $f_{\Delta v}$ is those that minimize (16). We have $1/f_{\Delta v}(v) \in C^\infty$ (almost everywhere). For $1/f(v) \in C^k$, $k \geq 4$, and according to Gaussian quadrature rule, the linear interpolation at gaussian abscissas $f_{\Delta v}$ satisfies

$$\int_{v_i}^{v_{i+1}} \left(\frac{1}{f(v)} - \frac{1}{f_{\Delta v}(v)}\right) dv = O\left(\Delta v^5\right) \tag{17}$$

The only point is to calculate the gaussian abscissas over V_i. Over $V_i = [v_i, v_{i+1}[$, the gaussian abscissas are :

$$\begin{aligned} v_{i,1} &= \frac{\sqrt{3}-1}{2\sqrt{3}} v_{i+1} + \frac{\sqrt{3}+1}{2\sqrt{3}} v_i \\ v_{i,2} &= \frac{\sqrt{3}+1}{2\sqrt{3}} v_{i+1} + \frac{\sqrt{3}-1}{2\sqrt{3}} v_i \end{aligned}$$

based on which, the linear interpolation of $f(v)$ can be described as follows:

$$f_{\Delta v}(v) = \frac{v_{i,2} - v}{v_{i,2} - v_{i,1}} f(v_{i,1}) + \frac{v - v_{i,1}}{v_{i,2} - v_{i,1}} f(v_{i,2})$$

Over each V_i, the local error (approximation of exit time) is of order $O\left(\Delta v^5\right)$. The estimate error on the exact spike time is obtained considering the exit times over the entire voltage-space that gives a global error of $O\left(\Delta v^4\right)$. This error estimate agrees exactly with the results of the numerical simulations.

It should be noted that this method requires one-dimensional neural models. The major reason is that (17) cannot be always fulfilled in high dimensional case. Moreover the methods also failed for neural network simulation. Assume that we can estimate the incoming spikes generated by presynaptic neurons to an accuracy of $O\left(\Delta v^4\right)$. Therefore a fourth order error is introduced in (17) and the error can be no better than $O\left(\Delta v^4\right)$ which makes impossible to calculate the exit time with a fifth-order accuracy.

Acknowledgments

Research supported by the INRIA cooperative research initiative RDNR.

References

Ambard, M. and D. Martinez (2006). Inhibitory control of spike timing precision. *NeuroComputing* **70**, 200–205.

Ariav, G., A. Polsky and J. J. Schiller (2003). Submillisecond precision of the input-output transformation function mediated by fast sodium dendritic spikes in basal dendrites of ca1 pyramidal neurons. *J. Neurosci.* **23**, 7750–7758.

Bair, W. and C. Koch (1996). Temporal precision of spike trains in extrastriate cortex of the behaving macaque monkey. *Neural Comp.* **8**, 1185–1202.

Breiman, L. (1993). Hinging hyperplanes for regression, classification, and function approximation. *IEEE Trans. Inf. Theory* **39**, 999–1013.

Brette, R. (2006). Exact simulation of integrate-and-fire models with synaptic conductances. *Neural Comp.* **18**, 2004–2027.

Brette, R. (2007). Exact simulation of integrate-and-fire models with exponential currents. *Neural Comp.* **19**, 2604–2609.

Brette, R. and W. Gerstner (2005). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *J. Neurophysiol.* **94**, 3637–3642.

Brette, R., M. Rudolph, T. Carnevale, M. Hines, D. Beeman, J.M. Bower, M. Diesmann, A. Morrison, P.H. Goodman, F.C. Harris, M. Zirpe, T. Natschlager, D. Pecevski, B. Ermentrout, M. Djurfeldt, A. Lansner, O. Rochel, T. Vieville, E. Muller, A.P. Davison, S. El Boustani and A. Destexhe (2007). Simulation of networks of spiking neurons:a review of tools and strategies. *J. Comput. Neurosci.* **23**, 349–398.

Brunel, N. and P. Latham (2003). Firing rate of the noisy quadratic integrate-and-fire neuron. *Neural Comp.* **15**, 2281–2306.

Della-Dora, J, A Maignan, M Mirica-Ruse and S Yovine (2001). Hybrid computation. *ISSAC'01.*

DeWeese, M., M. Wehr and A. Zador (2003). Binary spiking in auditory cortex. *J. Neurosci.* **23**, 7940–7949.

Ermentrout, G.B. (1996). Type i membranes, phase resetting curves, and synchrony. *Neural Comp.* **6**, 979–1001.

Ermentrout, G.B. and N. Kopell (1986). Parabolic bursting in an excitable system coupled with a slow oscillation. *SIAM J. Appl. Math.* **46**, 233–253.

Foldiak, P. and M. Young (1995). Sparse coding in the primate cortex. *The handbook of brain theory and neural networks (M. Arbib, ed.), Mit Press* pp. 895–898.

Fourcaud-Trocmé, N., D. Hansel, C. van Vreeswijk and N. Brunel (2003). How spike generation mechanisms determine the neuronal response to fluctuating inputs. *J. Neurosci.* **23**, 11628–11640.

Girard, A. (2002). Approximate solutions of odes using piecewise linear vector fields. *5th International Workshop on Computer Algebra in Scientific Computing.*

Hansel, D. and G. Mato (2001). Existence and stability of persistent states in large neuronal networks. *Phys. Rev. Lett.* **10**, 4175–4178.

Hansel, D., G. Mato, C. Meunier and L. Neltner (1998). On the numerical simulations of integrate-and-fire networks. *Neural Comp.* **10**, 467.

Hubbard, J. and B. West (1991). Differential equations: A dynamical systems approach. *Texts in Applied Mathematics, Vol. 5, Springer-Verlag, New York.*

Izhikevich, E. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks* **14**, 1569–1572.

Mainen, Z. and T. Sejnowski (1995). Reliability of spike timing in neocortical neurons. *Science* **1503**, 268.

Makino, T. (2003). A discrete-event neural network simulator for general neuron models. *Neural. Comput. and Applic.* **11**, 210–223.

Martinez, D. (2005). Oscillatory synchronization requires precise and balanced feedback inhibition in a model of the insect antennal lobe. *Neural Comp.* **17**, 2548–2570.

Mattia, M. and P. Del Giudice (2000). Efficient event-driven simulation of large networks of spiking neurons and dynamical synapses. *Neural Comp.* **12**, 2305.

McKean, H. P. (1970). Nagumo's equation. *Advances in Mathematics* **4**, 209–223.

Morrison, A., S. Straube, H. E. Plesser and M. Diesmann (2007). Exact subthreshold integration with continuous spike times in discrete time neural network simulations. *Neural Comp.* **19**, 44–79.

Perez-Orive, J., O. Mazor, G. C. Turner, S. Cassenaer, R. I. Wilson and G. Laurent (2002). Oscillations and sparsening of odor representations in the mushroom body. *Science* **297**, 359–365.

Rangan, V. A. and D. Cai (2007). Fast numerical methods for simulating large-scale integrate-and-fire neuronal networks. *J. Comput. Neurosci.* **22**, 81–100.

Richardson, M.J.E. (2004). Effects of synaptic conductance on the voltage distribution and firing rate of spiking neurons. *Phys. Rev. E.*

Rinzel, J. and B. Ermentrout (1998). Analysis of sneural excitability. *In:C Koch and I Segev, Eds. Methods in Neuronal Modeling: From ions to networks. MIT Press* pp. 251–291.

Rochel, O. and D. Martinez (2003). An event-driven framework for the simulation of networks of spiking neurons. *Proc. 11th European Symposium on Artificial Neural Networks.*

Ros, E., R. Carrillo, E. M. Ortigosa, B. Barbour and R. Agis (2006). Event-driven simulation scheme of spiking neural networks using lookup tables to characterize neuronal dynamics. *Neural Comp.* **18**, 2959–2993.

Rudolph, M. and A. Destexhe (2006). Analytical integrate-and-fire neuron models with conductance-based dynamics for event-driven simulation strategies. *Neural Comp.* **18**, 2305.

Shelley, M. J. and L. Tao (2001). Efficient and accurate time-stepping schemes for integrate-and-fire neuronal networks. *J. Comp. Neurosci.* **11**, 111–119.

Tonnelier, A. and W. Gerstner (2003). Piecewise linear differential equations and integrate-and-fire neurons: insights from two-dimensional membrane models. *Phys. Rev. E.*

Tonnelier, A., H. Belmabrouk and D. Martinez (2007). Event driven simulation of nonlinear integrate-and-fire neurons. *Neural Comp.* **19**, 3226–3238.

VanRullen, R., R. Guyonneau and S. J. Thorpe (2005). Spike times make sense. *Trends in Neurosciences* **28**, 1–4.

Wang, X. and G. Buzsaki (1996). Gamma oscillation by synaptic inhibition in a hippocampal interneuronal network model. *J. Neurosci.* **16**, 6402–6413.