

Minimum energy desynchronizing control for coupled neurons

Ali Nabi · Mohammad Mirzadeh ·
Frederic Gibou · Jeff Moehlis

Received: 24 February 2012 / Revised: 8 July 2012 / Accepted: 25 July 2012 / Published online: 18 August 2012
© Springer Science+Business Media, LLC 2012

Abstract We employ optimal control theory to design an event-based, minimum energy, desynchronizing control stimulus for a network of pathologically synchronized, heterogeneously coupled neurons. This works by optimally driving the neurons to their phaseless sets, switching the control off, and letting the phases of the neurons randomize under intrinsic background noise. An event-based minimum energy input may be clinically desirable for deep brain stimulation treatment of neurological diseases, like Parkinson's disease. The event-based nature of the input results in its administration only when it is necessary, which, in general, amounts to fewer applications, and hence, less charge transfer to and from the tissue. The minimum energy nature of the input may also help prolong battery life for implanted stimulus generators. For the example considered, it is shown that the proposed control causes a considerable amount of randomization in the timing of each neuron's next spike, leading to desynchronization for the network.

Keywords Optimal control · Hodgkin-Huxley neuron · Hamilton-Jacobi-Bellman PDE · Desynchronization

1 Introduction

Pathological synchronization among the spiking neurons of the basal ganglia and the thalamus regions of the brain is thought to be one cause for the involuntary tremors that patients with Parkinson's disease experience (Volkmann et al. 1996). Deep Brain Stimulation (DBS), an FDA-approved surgical treatment procedure, has shown success in alleviating these tremors by administration of high frequency pulsatile stimuli through an electrode implanted deep into the patient's brain which, hypothetically, desynchronizes the neurons (Pare et al. 1990; Nini et al. 1995; Wilson et al. 2011). This has motivated researchers to adopt control theory and investigate alternative desynchronizing stimuli with less possible side-effects such as tissue damage or adaptation, and with less energy consumption. Various control methods have been investigated and applied to different models in the past. Among these control methods, feedback control and optimal control are more prominent. These methods are attractive from a clinical perspective in that the control stimulus is designed to be applied only when needed (characterized by the feedback signal) and in an optimal way (characterized by the optimality criteria). For example, in Tass (1999) a system of noisy coupled phase neurons is studied and a demand-controlled deep-brain double-pulse stimulation has been suggested, where a double-pulse stimulus is administered when a feedback signal indicates occurrence of synchronization. In Popovych et al. (2006) and Kiss et al. (2007) nonlinear delay feedback control has been considered that can achieve desynchronization for systems of globally coupled limit-cycle oscillators. In Feng et al. (2007a, b) the authors used a genetic algorithm to optimally identify

Action Editor: David Terman

A. Nabi (✉) · M. Mirzadeh · F. Gibou · J. Moehlis
Department of Mechanical Engineering,
University of California at Santa Barbara,
Santa Barbara, 93106-5070, CA
e-mail: nabi@engineering.ucsb.edu

a brain region and design a model-independent optimal input specific to each patient's brain. In Schiff and Sauer (2008) and Schiff (2010), unscented Kalman filtering has been shown to achieve success in optimally estimating the unobservable states of a neuron through the feedback information from the observable state, which may be important for designing DBS control. In Danzl et al. (2009), a Hamilton-Jacobi-Bellman approach has been taken to design a minimum time desynchronizing control law for a globally coupled network. In recent work (Nabi and Moehlis 2011a), we have considered the problem of desynchronization for a deterministic system of coupled Hodgkin-Huxley phase neurons driven by a single constrained input. Dynamic programming was used to find a minimum energy desynchronizing control. In addition to those mentioned, a number of other studies have also shown potential in desynchronizing a population of pathologically synchronized neurons (Schöll et al. 2009; Nabi and Moehlis 2010). On a single neuron level, various event-based optimal control ideas have been considered as well (Schiff et al. 1994; Moehlis et al. 2006; Danzl et al. 2010; Nabi and Moehlis 2009; Dasanayake and Li 2011; Stigen et al. 2011). In recent work (Nabi and Moehlis 2011b) we have considered the problem of controlling the inter-spike-interval of different phase models of neurons in minimum (and maximum) possible time given a bounded input.

In this article, we present a non-trivial extension of work done in Danzl et al. (2009) and design an event-based, minimum energy, desynchronizing control stimulus for a network of pathologically synchronized, heterogeneously coupled neurons. We note that applying event-based minimum energy stimuli in a DBS setting for treatment of Parkinson's disease is clinically desirable in that it could reduce the number of stimulus applications and the amount of energy needed per stimulation. The proposed design works by optimally driving the neurons to their *phaseless sets*, switching the control off, and letting the phases of the neurons randomize under the intrinsic background noise (cf., Winfree 2001). We find that with the method of the present study, the amount of energy that is needed to effectively achieve desynchronization can be substantially reduced when compared to the method of Danzl et al. (2009).

For any stable oscillatory system, such as a neuron in a periodic spiking regime, a phase variable can be defined to characterize the evolution of the system states on its periodic orbit. This way, a one-to-one relationship is established between the states on the periodic orbit and the value of the phase variable. The concept of phase can be extended from the periodic

orbit to its entire basin of attraction with the idea of asymptotic phase (Guckenheimer 1975; Brown et al. 2004), defined so that all the points in the basin of attraction of a stable periodic orbit that converge to the same point on the periodic orbit as $t \rightarrow +\infty$ have the same value of phase. This produces level sets of phase within the basin of attraction that are called isochrons (Winfree 2001). Each isochron is the locus of all initial conditions that result in trajectories that asymptotically converge to the same point on the periodic orbit. The points at which all isochrons converge are where a phase value can not be defined, and hence the collection of these points make the phaseless set for the oscillator. One can randomize the phase of a noisy oscillator by steering its state to its phaseless set, and letting it randomly fall on an isochron due to its intrinsic noise. In neuronal systems, this yields randomization of the next spike time for each neuron, and hence, desynchronization in a network structure. As a proof of concept, Fig. 1 shows the voltage traces and spike time histogram for a network of 100 all-to-all coupled and synchronized neurons, more formally introduced in the following section, that are all initialized at their phaseless set after the control is switched off. It is seen that despite the synchronizing force of the coupling and the fact that the network is initialized coherently, the next spike time for the neurons has been effectively randomized due to an intrinsic zero-mean, variance 2, Gaussian white background noise.

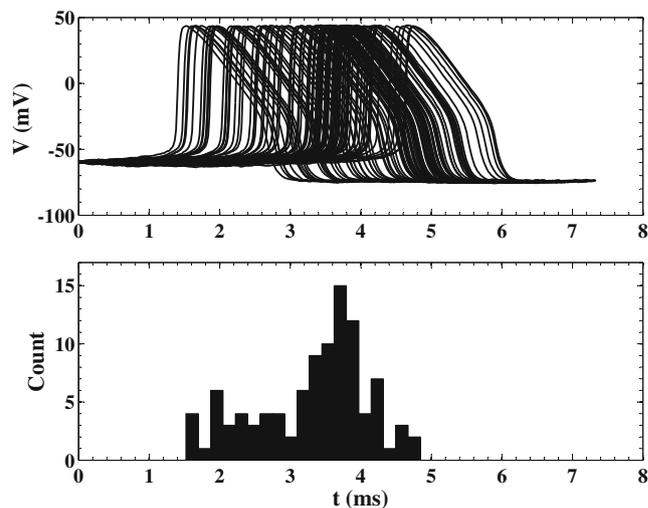


Fig. 1 Voltage traces and spike time histogram for a network of 100 coupled and synchronized neurons (described by (1) in the absence of any control u) that are initialized at their phaseless points, subject to i.i.d. Gaussian white background noise with zero-mean and variance 2

The organization of the paper is as follows. In Section 2 we describe the model used for the neuron. Then we lay out the control method and derive the necessary equations in Section 3. In Section 4, we describe the essentials of the numerical method that was used to solve the equations. Results and discussion are presented in Section 5 for both a single neuron and a network of neurons. In Section 6, we investigate the robustness of the results to heterogeneity in the neural population. Finally, we present concluding remarks in Section 7. This paper extends the preliminary results from Nabi et al. (2012).

2 Model

The model considered for the neurons in the population is as follows:

$$\dot{V}_i = f_V(V_i, n_i) + \eta_i(t) + \frac{1}{N} \sum_{j=1}^N \alpha_{ij}(V_j - V_i) + u, \tag{1}$$

$$\dot{n}_i = f_n(V_i, n_i).$$

Here, $i, j = 1, \dots, N$, where N is the total number of neurons in the network, V_i and n_i are the membrane voltage and the gating variable for neuron i , α_{ij} is the coupling strength between neurons i and j , which are assumed to be electrotonically coupled (Johnston and Wu 1995) with $\alpha_{ij} = \alpha_{ji}$ and $\alpha_{ii} = 0$ for all i, j , $\eta_i(t) = \sqrt{2D}\mathcal{N}(0, 1)$ is the intrinsic noise for each neuron taken as zero-mean Gaussian white noise with variance $2D$, $u = I(t)/c$ is the common control input where $I(t)$, in $\mu\text{A}/\text{cm}^2$, gives the DBS input current, and $c = 1 \mu\text{F}/\text{cm}^2$ is the constant membrane capacitance. Also,

$$f_V = (I_b - \bar{g}_{Na}[m_\infty(V)]^3(0.8 - n)(V - V_{Na}) - \bar{g}_K n^4(V - V_K) - \bar{g}_L(V - V_L))/c,$$

$$f_n = a_n(V)(1 - n) - b_n(V)n$$

are the state dynamics for each neuron in the absence of noise, coupling, and control. This is a two-dimensional reduction of the celebrated four-dimensional Hodgkin-Huxley (HH) model (Hodgkin and Huxley 1952) that captures the essentials of a neuron’s dynamical behavior (cf., Keener and Sneyd 1998; Moehlis 2006). The full HH model was originally developed for the Loligo squid’s giant axon through a series of experiments. The

other functions and parameters in this reduced model are

$$m_\infty(V) = \frac{a_m(V)}{a_m(V) + b_m(V)},$$

$$a_m(V) = 0.1(V + 40)/(1 - \exp(-(V + 40)/10)),$$

$$b_m(V) = 4 \exp(-(V + 65)/18),$$

$$a_n(V) = 0.01(V + 55)/(1 - \exp(-(V + 55)/10)),$$

$$b_n(V) = 0.125 \exp(-(V + 65)/80),$$

$$V_{Na} = 50 \text{ mV}, \quad V_K = -77 \text{ mV}, \quad V_L = -54.4 \text{ mV},$$

$$\bar{g}_{Na} = 120 \text{ mS}/\text{cm}^2, \quad \bar{g}_K = 36 \text{ mS}/\text{cm}^2,$$

$$\bar{g}_L = 0.3 \text{ mS}/\text{cm}^2, \quad c = 1 \mu\text{F}/\text{cm}^2.$$

Also, I_b , in $\mu\text{A}/\text{cm}^2$, is the neuron’s baseline current which represents the effect of other parts of the brain on the neuron under consideration and can be viewed as a bifurcation parameter in the model that controls whether the neuron is in an excitable or an oscillatory regime. We consider $I_b = 10 \mu\text{A}/\text{cm}^2$ to ensure oscillatory (periodic spiking) behavior for the neuron. With this, the period of spiking is $T_s = 11.85 \text{ ms}$. \bar{g}_{Na} , \bar{g}_K , and \bar{g}_L are the conductances of the sodium, potassium, and leakage channels, respectively. Also, V_{Na} , V_K , and V_L represent their respective reversal potentials.

In the absence of noise, coupling, and control, the oscillatory behavior of Eq. (1) is seen as a periodic orbit in the $V - n$ phase plane of the system, shown in Fig. 2 as the thick solid black line. The isochrons for this system are shown as gray lines. The isochrons converge at the unstable fixed point where the V - and n -nullclines intersect. This unstable fixed point is the phaseless set for this system (Osinga and Moehlis 2010). In the present article, we first find the optimal control stimulus that, when applied to a single neuron, drives the system to its phaseless set. As mentioned before, the idea here is that once the state of the system is at the phaseless set, the intrinsic background noise could cause the system to fall on a random isochron, thereby randomizing the phase of the neuron and its next spiking time. We then apply this optimal control to the population of synchronized, coupled, and noisy neurons and evaluate its performance in desynchronizing the population. Although we consider this specific model in this paper, we expect that a similar approach to that described below can be used to find optimal control inputs for other neuron models.

In order to better stabilize the numerical simulation, we scale down the V dimension in Eq. (1) by a

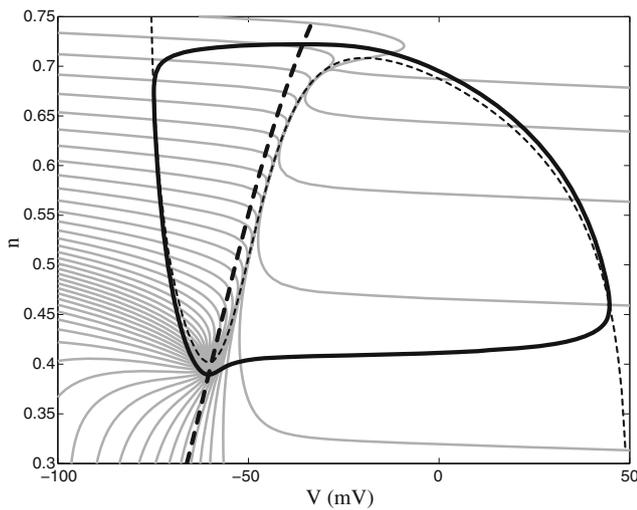


Fig. 2 Periodic orbit (*thick solid*), V -nullcline (*thin dashed*), and n -nullcline (*thick dashed*), and fifty isochrons equally spaced in phase for the two-dimensional reduced Hodgkin-Huxley model in the absence of noise, coupling, and control. The location of the unstable fixed point (*phaseless point*) for this system is at the intersection of the nullclines

factor of $K = 100$ so that the two states are of same order of magnitude. Consider the change of variables $z \equiv (x, y) = (\frac{1}{K}V, n)$. In view of Eq. (1), for a single deterministic neuron under control, we get

$$\dot{z} = F(z) + Bu, \tag{2}$$

where $B = [\frac{1}{K}, 0]^T$ and

$$F(z) = \begin{bmatrix} f_x(z) \\ f_y(z) \end{bmatrix} = \begin{bmatrix} \frac{1}{K} f_V(Kx, y) \\ f_n(Kx, y) \end{bmatrix}. \tag{3}$$

We note that this scaling is only for the sake of numerical stability and the results that we present later are all in the original $V - n$ coordinates.

3 Optimal control

We consider the system Eq. (2). The objective is to find the optimal control law that would take the system to its phaseless set in some prespecified length of time $[0, T_{\text{end}}]$, while minimizing the cost function

$$J(z, u(t)) = \int_0^{T_{\text{end}}} u^2 dt + \gamma q(z(T_{\text{end}})). \tag{4}$$

This cost function is composed of a time-additive portion, $\int_0^{T_{\text{end}}} u^2 dt$, that characterizes the total input energy being used, and an end-point cost, $q(z(T_{\text{end}}))$, that discriminates between different possible outcomes for the end states. γ is a penalizing scalar. We consider

bounded inputs, i.e., $|u| \leq u_{\text{max}}$, as would be the case in practice due to hardware limitations as well as tissue sensitivity.

To find the optimal control, we employ a Hamilton-Jacobi-Bellman (HJB) approach (Kirk 1970; Hespanha 2007). The idea in this approach is that one finds the optimal control to go from the current point in time and space to the end target point irrespective of how the system has reached the current state. To find the optimal control, one first defines the cost-to-go function, also known as the value function, from state z and time $\tau \in [0, T_{\text{end}}]$. The cost-to-go function $\mathcal{V}(z, \tau)$ gives the *minimum* cost that one needs to pay to go from (z, τ) to the end point subject to the constraints on the optimal control:

$$\begin{aligned} \mathcal{V}(z, \tau) &= \min_{\substack{|u(t)| \leq u_{\text{max}} \\ \forall t \in [\tau, T_{\text{end}}]}} J \\ &= \min_{\substack{|u(t)| \leq u_{\text{max}} \\ \forall t \in [\tau, T_{\text{end}}]}} \left[\int_{\tau}^{T_{\text{end}}} u^2 dt + \gamma q(z(T_{\text{end}})) \right]. \end{aligned} \tag{5}$$

With this definition, following classical optimal control theory (Kirk 1970; Hespanha 2007), we can write

$$\begin{aligned} \mathcal{V}(z, \tau) &= \min_{\substack{|u(t)| \leq u_{\text{max}} \\ \forall t \in [\tau, T_{\text{end}}]}} \left[\int_{\tau}^{\tau+h} u^2 dt + \right. \\ &\quad \left. + \int_{\tau+h}^{T_{\text{end}}} u^2 dt + \gamma q(z(T_{\text{end}})) \right], \end{aligned}$$

where, $h \in [0, T_{\text{end}} - \tau]$. Note that, due to causality, the first integral in this equation is independent of $u(t)$ for $t \in [\tau + h, T_{\text{end}}]$. On the other hand, the second integral and the value of the end point cost are, in general, dependent on $u(t)$ for all $t \in [\tau, T_{\text{end}}]$. The reason for this is that in the general case, the integrand u^2 could be any function of the form $G(t, u(t), z(t))$ and so the $u(t)$ for $t \in [\tau, \tau + h]$ determines the state of the state at $t = \tau + h, z(\tau + h)$, which in turn comes into play when finding $u(t)$ for $t \in [\tau + h, T_{\text{end}}]$. With this, we can write

$$\begin{aligned} \mathcal{V}(z, \tau) &= \min_{\substack{|u(t)| \leq u_{\text{max}} \\ \forall t \in [\tau, \tau+h]}} \left[\int_{\tau}^{\tau+h} u^2 dt + \right. \\ &\quad \left. + \min_{\substack{|u(t)| \leq u_{\text{max}} \\ \forall t \in [\tau+h, T_{\text{end}}]}} \left(\int_{\tau+h}^{T_{\text{end}}} u^2 dt + \gamma q(z(T_{\text{end}})) \right) \right], \end{aligned}$$

where the first (outer) minimum operates on the entire expression, whereas the second (inner) minimum acts only on the second integral and the end point cost. We realize that the inner minimum gives exactly the cost-to-go function from state $z(\tau + h)$ and time $t = \tau + h$: $\mathcal{V}(z(\tau + h), \tau + h)$. By subtracting $\mathcal{V}(z(\tau), \tau)$ from both

sides of the above equation and dividing by $h > 0$, we get

$$0 = \min_{\substack{|u(t)| \leq u_{\max} \\ \forall t \in [\tau, \tau+h]}} \left[\frac{1}{h} \int_{\tau}^{\tau+h} u^2 dt + \frac{\mathcal{V}(z(\tau+h), \tau+h) - \mathcal{V}(z(\tau), \tau)}{h} \right],$$

for any $\tau \in [0, T_{\text{end}}]$ and $h \in [0, T_{\text{end}} - \tau]$. Taking the limit as $h \rightarrow 0$, yields

$$0 = \min_{|u(\tau)| \leq u_{\max}} \left[u(\tau)^2 + \frac{\partial \mathcal{V}}{\partial z}(z(\tau), \tau) \dot{z}(\tau) + \frac{\partial \mathcal{V}}{\partial t}(z(\tau), \tau) \right].$$

Since $\frac{\partial \mathcal{V}}{\partial t}(z(\tau), \tau)$ does not depend on u , one can take it out of the minimization and obtain the well-known HJB equation by substituting $\dot{z}(\tau) = F(z(\tau)) + Bu(\tau)$ from the dynamics:

$$0 = \frac{\partial \mathcal{V}}{\partial t}(z(\tau), \tau) + \min_{|u(\tau)| \leq u_{\max}} \left[u(\tau)^2 + \frac{\partial \mathcal{V}}{\partial z}(z(\tau), \tau) (F(z(\tau)) + Bu(\tau)) \right], \tag{6}$$

which is a partial differential equation (PDE) with the boundary condition

$$\mathcal{V}(z(T_{\text{end}}), T_{\text{end}}) = \gamma q(z(T_{\text{end}})). \tag{7}$$

It should be noted that since we let $h \rightarrow 0$ to obtain Eq. (6), the minimization with respect to the control function in Eq. (6) reduces from choosing values of a curve, $|u(t)| \leq u_{\max}, \forall t \in [\tau, \tau + h]$, to choosing a single value on the curve, $|u(\tau)| \leq u_{\max}$ at $t = \tau$ (Caputo 2005). By defining

$$\mathcal{H}(z, \nabla \mathcal{V}, u) = u^2 + \nabla \mathcal{V}(z(t), t)(F(z(t)) + Bu(t)) \tag{8}$$

as the Hamiltonian for the system, one can rewrite Eq. (6) more succinctly as

$$\frac{\partial \mathcal{V}}{\partial t} + \min_{|u| \leq u_{\max}} \mathcal{H}(z, \nabla \mathcal{V}, u) = 0, \tag{9}$$

where $\nabla \mathcal{V}$ is the gradient of the value function with respect to z , $(\frac{\partial \mathcal{V}}{\partial x}, \frac{\partial \mathcal{V}}{\partial y})^T$. The optimal control that globally minimizes \mathcal{H} is obtained as

$$u^*(t) = \arg \min_{|u| \leq u_{\max}} [u^2 + \nabla \mathcal{V}(z^*(t), t)(F(z^*(t)) + Bu(t))],$$

where $z^*(t)$ represents the optimal trajectory.

In order to find the optimal control, we can set the derivative of the Hamiltonian Eq. (8) with respect to u equal to zero and solve for the extremal u . This is true as long as the magnitude of the control remains smaller

than the predetermined bound u_{\max} . When the magnitude of the optimal control reaches the bound u_{\max} , it saturates in accordance to Pontryagin’s minimum principle (Pontryagin et al. 1962; Kirk 1970). Considering Eq. (8), $\frac{\partial \mathcal{H}}{\partial u} = 0$ results in $u^*(t) = -\frac{1}{2} \nabla \mathcal{V}^T B$ as long as $|u^*(t)| \leq u_{\max}$. Equivalently, one can write $u^*(t) = -\frac{1}{2} \nabla \mathcal{V}^T B$ for when $|\nabla \mathcal{V}^T B| \leq 2u_{\max}$. When $|\nabla \mathcal{V}^T B| = 2u_{\max}$, the optimal control reaches its bound and if $|\nabla \mathcal{V}^T B| > 2u_{\max}$, it gets saturated, in which case, considering Eq. (8) with $|u^*(t)| = u_{\max}$, the minimizing optimal control becomes $u^*(t) = -\text{sign}(\nabla \mathcal{V}^T B)u_{\max}$. So in summary, considering the fact that $B = [\frac{1}{K}, 0]^T$, we get the optimal control as

$$\begin{aligned} u^*(t) &= -\frac{1}{2K} \mathcal{V}_x, & |\mathcal{V}_x| &\leq 2Ku_{\max}, \\ u^*(t) &= -\text{sign}(\mathcal{V}_x)u_{\max}, & |\mathcal{V}_x| &> 2Ku_{\max}, \end{aligned} \tag{10}$$

where $\mathcal{V}_x = \frac{\partial \mathcal{V}}{\partial x}$. With this optimal control, the Hamiltonian can be written as

$$\begin{aligned} \mathcal{H} &= \nabla \mathcal{V}^T F(z) - \frac{1}{4K^2} \mathcal{V}_x^2, & |\mathcal{V}_x| &\leq 2Ku_{\max}, \\ \mathcal{H} &= \nabla \mathcal{V}^T F(z) + u_{\max}^2 - |\mathcal{V}_x| \frac{u_{\max}}{K}, & |\mathcal{V}_x| &> 2Ku_{\max}. \end{aligned} \tag{11}$$

In order to find the optimal control $u^*(t)$ in Eq. (10), we need to find the cost-to-go, $\mathcal{V}(z, t)$ from the HJB PDE Eq. (6) with boundary condition Eq. (7) and Hamiltonian Eq. (11). This is done numerically as explained in the following.

4 Numerical method

The HJB PDE Eq. (6) is a special form of a broader class of equations known as the Hamilton-Jacobi (HJ) equations which for a scalar variable $\mathcal{V}(z, t)$ are given by

$$\frac{\partial \mathcal{V}}{\partial t} + \mathcal{H}(z, t, \mathcal{V}, \nabla \mathcal{V}) = 0,$$

where \mathcal{H} is the Hamiltonian and $\nabla \mathcal{V}$ denotes spatial gradients. These equations frequently appear in different areas of research such as optimal control theory, image processing and computational physics (Kirk 1970; Osher and Fedkiw 2003; Sethian 1999) and thus have been well studied in the past.

The numerical solution of the HJ equation is deeply rooted in the methods that already had existed for the solution of nonlinear hyperbolic conservation laws (HCL) (Osher and Fedkiw 2003). Originally, Crandall and Lions (1984) proposed their first-order accurate numerical algorithm for the solution of the HJ equation

and a few years later, Osher and Sethian (1988) used the connection between HJ and HCL to derive higher order accurate algorithms. For a more complete list of references, one may consult standard texts such as Osher and Fedkiw (2003) and Sethian (1999).

Generally, a convergent, high-order approximation to the HJ equation consists of three steps: (i) Computing the solution gradient $\nabla\mathcal{V}$, which is typically achieved with essentially non-oscillatory (ENO) schemes (Harten et al. 1987; Shu and Osher 1989). These schemes are designed such that they do not produce oscillatory results when the solution gradients are evaluated close to the discontinuities that are inherent to nonlinear HJ equations. (ii) Evaluating the Hamiltonian function, which is straightforward only for linear problems. For nonlinear problems it is necessary to compute the so-called *numerical* Hamiltonian (see $\hat{\mathcal{H}}$ in Osher and Shu 1991) in such a way as to account for the nonlinear shock and rarefaction phenomena; Godunov or local Lax-Friedrichs (LLF) schemes are traditionally used. In this paper, we use the LLF scheme to obtain the numerical Hamiltonian. (iii) Given an initial condition $\mathcal{V}(z, 0)$, time integration is needed to obtain the solution $\mathcal{V}(z, t)$ at later times. To prevent non-physical oscillations in the solution, this is done using a total variation diminishing (TVD) method (Osher and Fedkiw 2003).

The details regarding the careful implementation of these steps are included in the [Appendices](#). In this study we have used the Matlab toolbox, “Level Set Methods Toolbox” written by Mitchell (2007), which is a working example of such implementations.

We set $T_{\text{end}} = 7$ ms and use a 321×321 uniform grid for the states to solve the HJB Eq. (6) for the cost-to-go function $\mathcal{V}(z, t)$. The control bound is set to be $u_{\text{max}} = 10 \mu\text{A}/\mu\text{F}$. We also set the end point cost to be

$$\mathcal{V}(z(T_{\text{end}}), T_{\text{end}}) = \gamma \left(1 - e^{-\left(\frac{(x-x_{pt})^2}{\sigma_x^2} + \frac{(y-y_{pt})^2}{\sigma_y^2} \right)} \right),$$

where $\gamma = 1000$, $\sigma_x^2 = \sigma_y^2 = 0.001$, and $(x_{pt}, y_{pt}) = (\frac{1}{K}V_{pt}, n_{pt})$ where $K = 100$ and $(V_{pt}, n_{pt}) = (-59.6, 0.403)$ is the phaseless target point. This Gaussian end point cost function has a minimum of zero at the phaseless point that encourages the evolution of the controlled system toward this point. We note that we solve the HJB equation backward in time and treat this end point cost as the initial condition for the equations.

Once the solution $\mathcal{V}(z, t)$ is computed, the optimal control is found as a function of the state at all time steps using Eq. (10). Given this data in time and space, the optimal control sequence $u(t)$, and the optimal trajectories can be found by forward integrating Eq. (1)

in the absence of noise and coupling, and for any initial condition including the spiking point $(V_0, n_0) = (44.8, 0.459) \equiv (V_s, n_s)$, as we consider here. A fourth order Runge-Kutta method is used for the integration. We note that since $\mathcal{V}(z, t)$ is available on spatial grid points, a simple bilinear interpolation scheme is used to obtain the input off grid points at each time step. This optimal control sequence is then applied first to the noisy single neuron system to evaluate its performance in randomizing the noisy neuron’s next spiking time, and then, to a population of 100, initially synchronized, coupled neurons to evaluate its performance in desynchronizing the population in the presence of noise and heterogeneities in the coupling. Since introducing noise into the equations makes the problem a stochastic differential equation (SDE) problem, care must be taken in choosing a proper numerical algorithm for integration. A simple and straightforward algorithm is the Honeycutt’s second order stochastic Runge-Kutta method (Honeycutt 1992) which was chosen in this study.

5 Results and discussion

The top panel in Fig. 3 shows the minimum energy control law for the deterministic single neuron. As can be seen in Fig. 3, the control has saturated at the bound value $u_{\text{max}} = 10 \mu\text{A}/\mu\text{F}$ (equivalent to $I_{\text{max}} = 10 \mu\text{A}/\text{cm}^2$). Figure 3 also shows the evolution of the system states (V, n) in time as well as the optimal trajectory in the state space when driven by the control. It is worth pointing out that we set the initial condition to be the spiking state as this is a practical observable which can be used as a trigger for the control, hence producing an event-based control.

5.1 Single neuron level

To evaluate the performance of the control for phase randomization, we apply it to the single neuron in the presence of noise and integrate the noisy system forward in time. Figure 4 shows the results obtained for this case for 100 different numerical realizations. We have included three different cases in this figure for comparison. The top row in this figure shows the case of the neuron under its natural dynamics in the absence of both the noise and the control. As expected, the neuron spikes at its natural period $T_s = 11.85$ ms for all 100 different trials. The second panel shows the case where noise is active, but the control is not. As can be seen, the spiking instant of the neuron varies due to the effect of different noise realizations. In the third panel, both

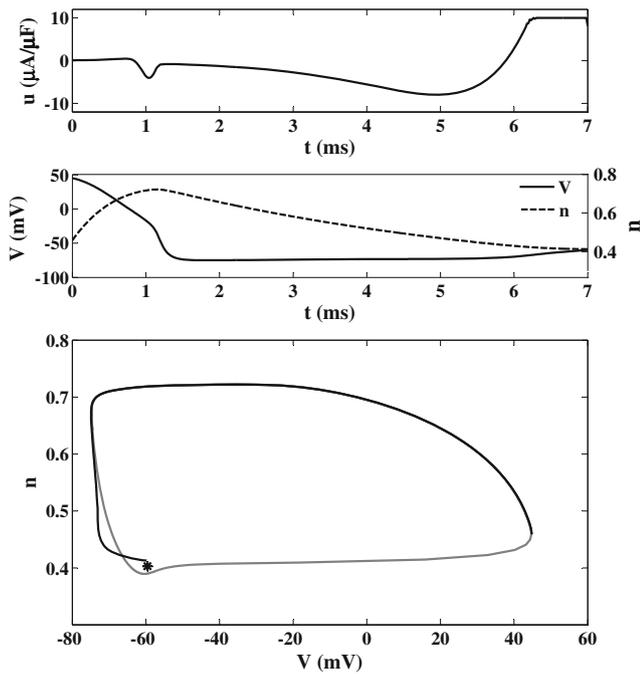


Fig. 3 Results for the deterministic single neuron system (Eq. (1) with $\eta_i(t) \equiv 0$ and $\alpha_{ij} = 0, \forall i, j$). The system is initialized at the spiking point $(V_s, n_s) = (44.2, 0.465)$ and the target point for the control is the phaseless set for the neuron which is its unstable fixed point $(V_{pl}, n_{pl}) = (-59.6, 0.403)$. *Top*: minimum energy control law that is bounded to $|u| \leq 10 \mu\text{A}/\mu\text{F}$. *Middle*: the time evolution of the states of the system with this control. *Bottom*: the state space representation of the trajectory of the system under the control shown in *top panel*. We see that the control has been able to take the system close to the phaseless point shown with asterisk marker

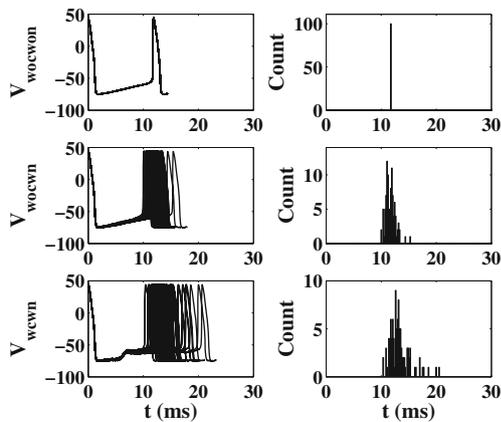


Fig. 4 Results for 100 different simulations for the system Eq. (1) with initial condition $(V_0, n_0) = (44.8, 0.459)$. *Top row*: voltage trace (left) and histogram (right) for the case of without noise and without external control; *middle row*: voltage trace (left) and histogram (right) for the case of 100 different noise realizations, without control; *bottom row*: voltage trace (left) and histogram (right) for the case of 100 different noise realizations, with one cycle of control

noise and the control are acting on the neuron. We see that applying the control causes the next spiking instant of the neuron to randomize over a considerable time interval. We note that the control has only been applied for one cycle and has been set to zero for $t > 7$ ms.

5.2 Population level

We now apply the minimum energy control that is found for a single neuron to the network of $N = 100$ coupled synchronized noisy neurons with common coupling strength $\alpha_{ij} = 0.1$ and i.i.d. noise with $D = 1$. For this system, we define the mean voltage as the observable for the network

$$\bar{V}(t) = \frac{1}{N} \sum_{i=1}^N V_i(t), \tag{12}$$

and set $\bar{V} = -20$ mV as the event that triggers one cycle of control administration. After one cycle, the

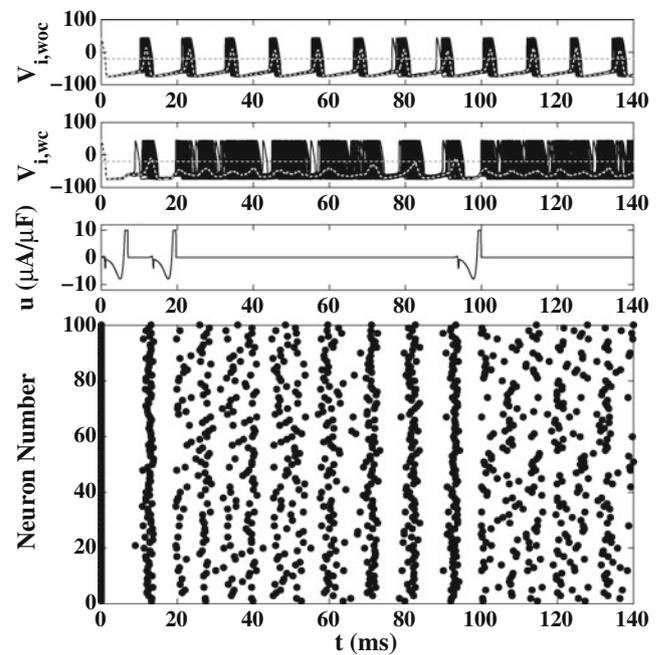


Fig. 5 Results for a population of $N = 100$ coupled neurons with $\eta_i = \sqrt{2D}\mathcal{N}(0, 1)$, $D = 1$, and coupling strength $\alpha = 0.1$. *First panel* shows the result for the noisy network without control. The *second panel* shows the results for the same network with active event-based control. The *dotted gray* traces show the mean voltage for each case and the horizontal dotted lines mark the control activation threshold. We see that the control (shown in *third panel*) has only been applied when the mean voltage has reached the $\bar{V} = -20$ mV threshold, and has been able to substantially desynchronize the network as communicated by the raster plot

control turns off until the next event triggers it. Figure 5 shows the result for this with different noise for each of the neurons in the system. The first panel shows the individual voltages and mean voltage for the coupled system with activated noise, but without control. We see that the mean voltage spikes are always above the -20 mV threshold that is shown as dotted line. The second panel shows the individual voltages and mean voltage for the coupled system when both the noise and the control are present. The control input is shown in the third panel. The desynchronizing effect of the control is clearly seen from the raster plot. The event-based nature of the control is also apparent from the fact that the control has only been turned on when the mean voltage has crossed the threshold line.

It should be noted that when we increase the coupling strength between the neurons to $\alpha_{ij} = 0.2$, we see the same qualitative results, but with more instances of control application due to the network's higher tendency for synchronization (results shown in Fig. 6).

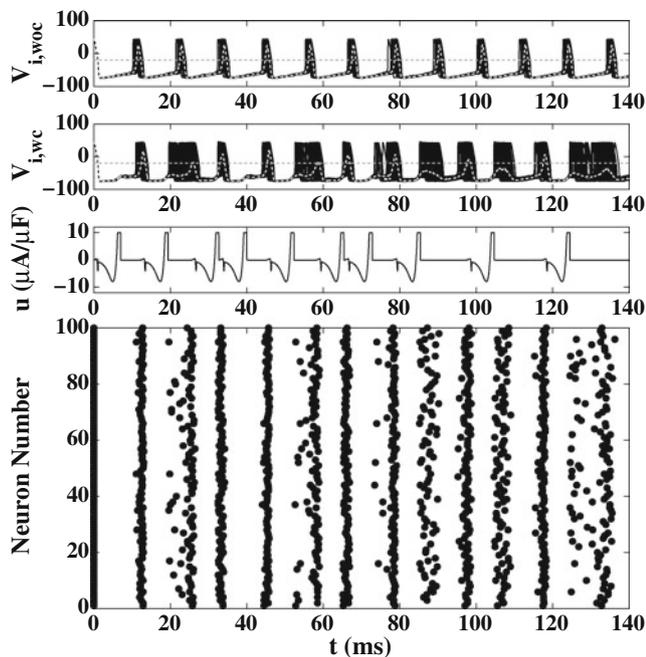


Fig. 6 Results for a population of $N = 100$ coupled neurons with $\eta_i = \sqrt{2D}\mathcal{N}(0, 1)$, $D = 1$, and coupling strength $\alpha = 0.2$. *First panel* shows the result for the noisy network without control. The *second panel* shows the results for the same network with active event-based control. The *dotted gray traces* show the mean voltage for each case and the *horizontal dotted lines* mark the control activation threshold. We see that the control (shown in *third panel*) has only been applied when the mean voltage has reached the $\bar{V} = -20$ mV threshold, and has been able to reasonably desynchronize the network as communicated by the raster plot

6 Robustness analysis

In this section, we investigate the performance of the optimal control found in the preceding section when there are uncertainties in the values of the coupling strengths or in the location of the phaseless set. This induces heterogeneities in the network, making it more realistic. First, instead of a common coupling strength, we draw the coupling strengths from a normal distribution with mean $\bar{\alpha} = 0.1$ and standard deviation $\sigma_\alpha = 0.02$, i.e., $\alpha_{ij} = \alpha_{ji} = \mathcal{N}(\bar{\alpha}, \sigma_\alpha) = \mathcal{N}(0.1, 0.02)$. We note that similar results are found with asymmetric coupling ($\alpha_{ij} \neq \alpha_{ji}$) between neuron pairs. The left panel in Fig. 7 shows this distribution. Simulating the network with this variability in the coupling strengths, we get the result shown in the second panel of Fig. 8. The first panel in this figure is for the case of the network with common coupling strength $\alpha_{ij} = 0.1$ as shown in Fig. 5, and is reillustrated here to facilitate comparison. When the coupling strengths are different, we see very similar behavior as before except that there is a delay in resynchronization after the network is desynchronized. We note that although this difference in timing of the resynchronization is seen, it is dependent on the particular realization of the α values. Next, we used the same realization of the coupling strengths, but randomly set 20 % of them equal to zero so that the network is not an all-to-all coupled network. A further delay in resynchronization is observed, as shown in the third panel of Fig. 8. This is reasonable since now some of the connections are broken and thus the overall drive for resynchronizing the network is smaller. We also considered the baseline current I_b of the neurons to be drawn from a normal distribution, so that the neurons are not all identical. The distribution considered for the I_b values is shown in the right panel of Fig. 7. With this additional variability added to those described before,

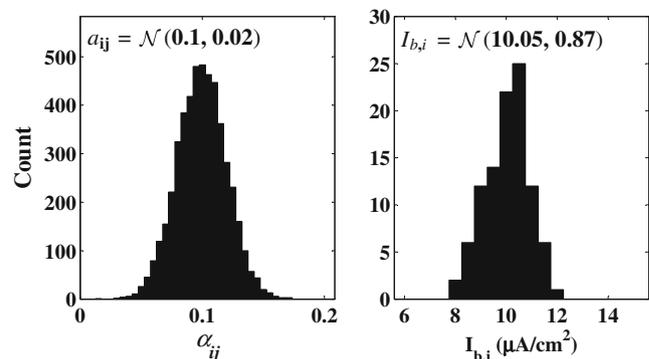


Fig. 7 Normal distributions for the coupling strengths (*left*) and the baseline currents (*right*) for inducing heterogeneity into the network of neurons

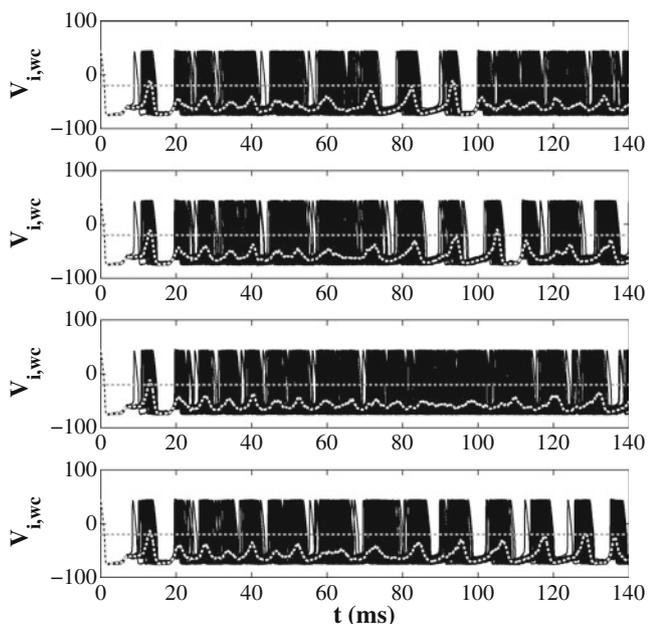


Fig. 8 Results for a population of $N = 100$ coupled neurons with active event-based control and $\eta_i = \sqrt{2D}\mathcal{N}(0, 1)$, $D = 1$; *first panel*: the network has a common coupling strength $\alpha = 0.1$ and the neurons all have the same baseline current $I_b = 10 \mu\text{A}/\text{cm}^2$; *second panel*: the coupling strengths are drawn from the distribution shown in Fig. 7, but the baseline current is the same for all neurons $I_b = 10 \mu\text{A}/\text{cm}^2$; *third panel*: the same coupling strengths and baseline current values are used as in the *second panel*, except that 20 % of the coupling strengths have been randomly chosen and set equal to zero; *fourth panel*: the coupling strengths are exactly the same as those used to produce panel three, but the baseline current values have been drawn from the normal distribution shown in Fig. 7. The seed for the random number generators have been set such that the same values for the random vectors were produced across all four experiments to facilitate comparison. The dotted gray traces show the mean voltage for each case and the *horizontal dotted lines* mark the control activation threshold. The control is only active when the mean voltage has crossed the threshold

we get the result shown in the fourth panel of Fig. 8. We see that the network seems to be synchronizing more often in this case. The reason for this is that when the baseline currents are not the same, the location of the phaseless point for each neuron is slightly different and so the efficiency of the optimal control is reduced.

With common coupling strength $\alpha_{ij} = 0.1$, we next consider heterogeneous I_b values for the neurons in a population of 100 coupled neurons. We draw the I_b values from normal distributions of the form $I_{b,i} = \mathcal{N}(\bar{I}_b, \sigma_{I_b})$ where the mean $\bar{I}_b = 10 \mu\text{A}/\text{cm}^2$ and the standard deviation $\sigma_{I_b} \in \{0, 0.2, 0.4, \dots, 3\}$. In order to have a measure of the performance of the controller, we count the number of control applications over an extended simulation time. We set the simulation time to be 60 times the natural period of a single noise-

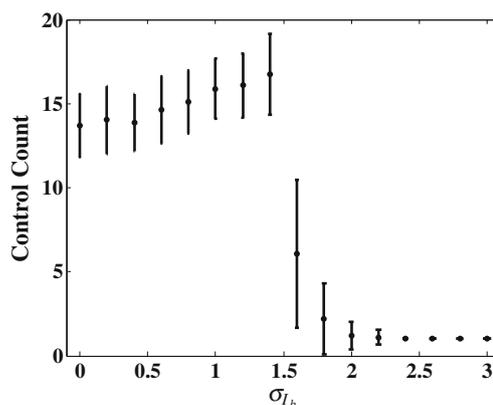


Fig. 9 Number of control applications in a network of 100 coupled neurons with homogeneous coupling strengths $\alpha_{ij} = 0.1$ in which the baseline current for each neuron is drawn from a normal distribution of the form $I_{b,i} = \mathcal{N}(\bar{I}_b, \sigma_{I_b})$. The mean is $\bar{I}_b = 10 \mu\text{A}/\text{cm}^2$ and the standard deviation $\sigma_{I_b} \in \{0, 0.2, 0.4, \dots, 3\}$. The total simulation time for each data point is set to be 60 times the natural period of the noiseless neuron with $\bar{I}_b = 10 \mu\text{A}/\text{cm}^2$, i.e., $T_{\text{sim}} \simeq 60 \times 11.85 \simeq 710$ ms. The control waveform that is applied to the system is that obtained from solving the HJB equation shown in Fig. 3. The statistics have been obtained for 100 different realizations of the noise for each value of σ_{I_b}

less neuron under $I_b = 10$. More instances of control application indicates that the controller has been less effective in desynchronizing the population. Figure 9 shows the statistical simulation results for 100 different noise realizations for each value of σ_{I_b} .

In Fig. 9, we see that by holding the mean value of the baseline current constant at $I_b = 10 \mu\text{A}/\text{cm}^2$, the number of control applications increases with increasing standard deviation of the $I_{b,i}$'s at first, but after a certain point, $\sigma_{I_b} \simeq 1.5$, there is a rapid drop in the number of control applications. The reason for this behavior is that by increasing the standard deviation from zero, the location of the phaseless point changes more from neuron to neuron and so the control becomes progressively less effective. This results in less desynchronization per control application and thus more instances of application. However, for $\sigma_{I_b} \simeq 1.5$ and higher, the heterogeneity in the I_b values has increased so much that the neurons become desynchronized after the first few control applications and do not resynchronize. This means that the heterogeneity overrides the weak coupling in the system.

7 Conclusion

We have considered the design of an event-based, minimum energy, desynchronizing control stimulus for a network of pathologically synchronized coupled

neurons. The control drives the neurons to their phaseless set, and lets the phases of the neurons randomize under intrinsic background noise. The minimum energy optimality criterion is desirable for practical purposes, as it may increase the battery life of implanted stimulus generators in patients with Parkinson’s disease treated by DBS. By employing the minimum energy formulation, the total input energy for the control input shown in Fig. 3 is computed to be $\int_0^{T_{\text{end}}} u^2 dt \approx 194$ which is about 70 % less than the minimum time approach considered in Danzl et al. (2009). Moreover, this approach was shown to be robust to heterogeneity in the coupling strengths and the baseline current.

Acknowledgment This work was supported by the National Science Foundation grants NSF-1000678 and CHE-1027817.

Appendix A: Essentially Non-Oscillatory (ENO) schemes

Finite difference approximations of derivatives of a function $\mathcal{V} : \mathbb{R}^n \mapsto \mathbb{R}$, are essentially equivalent to choosing an interpolation polynomial for the function \mathcal{V} , and performing exact differentiation. In traditional finite difference methods, the polynomial stencil is fixed, i.e., to approximate $\mathcal{V}_x \equiv \frac{\partial \mathcal{V}}{\partial x}$, at the grid point x_i , one assumes

$$\mathcal{V}_x|_i \approx f(\mathcal{V}_{i-m}, \mathcal{V}_{i-m+1}, \dots, \mathcal{V}_i, \dots, \mathcal{V}_{i+n-1}, \mathcal{V}_{i+n}),$$

where constants m and n are fixed in space and are the same for all points (except maybe at the boundaries). Here, the subscript indices refer to the grid points, i.e., $\mathcal{V}_i \equiv \mathcal{V}(x_i)$. Problems arise when the function \mathcal{V} is not sufficiently smooth and this interpolation, when combined with time integration, results in spurious oscillations and even divergence.

To remedy this problem, Harten et al. (1987) first introduced the idea of *essentially non-oscillatory* (ENO) schemes. Unlike traditional finite difference methods, in ENO schemes the polynomial stencil is not fixed, and at each point, one chooses the smoothest possible polynomial. When combined with a TVD time integration method (see below), the scheme is guaranteed not to produce any spurious solutions.

This idea was further improved, from the implementation point of view, by Shu and Osher (1989) and later applied to the numerical solution of the HJ equation (Osher and Shu 1991). Here we merely consider the

method in one spatial dimension. Extension of the method to higher spatial dimensions is possible through a dimension-by-dimension approach and we leave the details to the appropriate references mentioned above.

Consider the one-dimensional HJ equation written as

$$\frac{\partial \mathcal{V}}{\partial t} + \mathcal{H}(\mathcal{V}_x) = 0,$$

where the explicit dependence on other variables have been dropped for brevity. To construct a polynomial function, $Q_j^{j+\mathcal{M}}(x)$, of degree \mathcal{M} that interpolates through points $x_j, x_{j+1}, \dots, x_i, \dots, x_{j+\mathcal{M}}$, one first needs to define the so-called m th-order Newton divided differences coefficients, f_j^{j+m} , for $m = 0, 1, \dots, \mathcal{M}$. This is done via the recursive formula

$$f_j^{j+m} = \frac{f_{j+1}^{j+m} - f_j^{j+m-1}}{x_{j+m} - x_j},$$

with $f_j^j = \mathcal{V}_j = \mathcal{V}(x_j)$. Here the subscript index, j , and superscript index, $j + m$, refer to the lower and upper bounds of the interval used to compute the coefficient, i.e., to compute f_j^{j+m} , $m + 1$ grid points, x_j through x_{j+m} , are required. Using the divided differences coefficients, the polynomial $Q_j^{j+\mathcal{M}}(x)$ may be written as,

$$Q_j^{j+\mathcal{M}}(x) = \sum_{m=0}^{\mathcal{M}} f_j^{j+m} \phi_j^m(x), \tag{13}$$

where, by definition, $\phi_j^0(x) = 1$ and, for $m \geq 1$,

$$\phi_j^m(x) = \prod_{n=0}^{m-1} (x - x_{j+n}).$$

Once the interpolating polynomial, $Q_j^{j+\mathcal{M}}(x)$, is known, Eq. (13) may be differentiated with respect to x , to obtain the following approximation to the spatial derivative:

$$\mathcal{V}_x|_i = \left. \frac{dQ_j^{j+\mathcal{M}}}{dx} \right|_{x_i} + \mathcal{O}(h^{\mathcal{M}}),$$

where $h \equiv \max_{1 \leq n \leq \mathcal{M}} |x_{j+n} - x_{j+n-1}|$ is the maximum grid spacing in the interpolation interval and determines the order of truncation error.

Constructing the polynomial $Q_j^{j+\mathcal{M}}(x)$ also requires the knowledge of the interpolation interval, $\mathcal{I} \equiv [x_j, x_{j+1}, \dots, x_i, \dots, x_{j+\mathcal{M}}]$. To build the interval, and starting at point x_i , one has two possible options for choosing the next point, x_{i-1} or x_{i+1} . In fact, these

are both valid, and as we shall see, required for the next step of the algorithm. Thus at each point, x_i , two different approximations for \mathcal{V}_x exist which we denote as \mathcal{V}_x^+ and \mathcal{V}_x^- based on whether x_{i+1} or x_{i-1} are chosen, respectively. In fact, if one chooses $\mathcal{M} = 1$, these are simply the classical first-order upwind methods.

What comes next is the core idea of the ENO scheme. At each remaining step, toward finding the interval, one will face two options in choosing the next point. The idea is to choose the point that will result in the smoothest polynomial. In fact, it is easy to note that the divided differences coefficients are a good measure of the variations in the interpolating function. A coefficient with a large magnitude alerts the existence of a rapid change, or even a discontinuity in the function \mathcal{V} around point x_i , and thus should be avoided in constructing the polynomial.

For example, let's consider the second-order correction to \mathcal{V}_x^- . The two possible options are x_{i-2} and x_{i+1} . There are also two second-order divided differences coefficients, f_{i-2}^i and f_{i-1}^{i+1} . As a result one either chooses x_{i-2} if

$$|f_{i-2}^i| \leq |f_{i-1}^{i+1}|,$$

or x_{i+1} otherwise. This same idea is repeated for all higher-order coefficients until all remaining points in the interval are found, after which the interpolation polynomial is uniquely determined. Note that in this algorithm, one first determines the interval and then constructs the polynomial. It is possible to combine the two in one single pass as suggested in Osher and Fedkiw (2003).

Finally let us briefly mention that, since the ENO scheme always chooses the smoothest polynomial with minimal variations, it may dampen out even slightest gradients in the solution where no shock or discontinuity exist. To remedy this problem, Liu et al. (1994) introduced the idea of the weighted ENO (WENO) schemes in which one obtains a higher order numerical approximation to \mathcal{V}_x by appropriate weighting of all possible \mathcal{M} th-order ENO schemes. The weighting coefficients are usually chosen so that they inversely depend on the smoothness of corresponding ENO scheme in a potential interval. This idea then results in a WENO scheme that automatically switches to an ENO scheme in parts of the domain that the solution is non-smooth while obtaining higher-order approximations in the smooth part of the domain. We do not present the algorithm in detail here and refer the interested reader to Osher and Fedkiw (2003) for a discussion of the implementation and also the literature review on WENO schemes.

Appendix B: Numerical Hamiltonian

The second part in constructing a high-order algorithm for solving the HJ equation is the high-order construction of a *numerical Hamiltonian*, $\hat{\mathcal{H}}$. As noted in the previous section, at each point there are two approximations to the gradient, denoted by \mathcal{V}_x^+ and \mathcal{V}_x^- , depending on the initial stencil bias. As such, in general, the numerical Hamiltonian may be written as

$$\hat{\mathcal{H}} = \hat{\mathcal{H}}(\mathcal{V}_x^+, \mathcal{V}_x^-).$$

To get a numerical Hamiltonian that correctly accounts for the nonlinear shock and rarefaction phenomena (see Osher and Shu 1991), three criteria must be satisfied. First, the Hamiltonian needs to be Lipschitz continuous in both \mathcal{V}_x^+ and \mathcal{V}_x^- . Second, the numerical Hamiltonian needs to be a non-increasing function of \mathcal{V}_x^+ and a non-decreasing function of \mathcal{V}_x^- . Symbolically, this is usually denoted as $\hat{\mathcal{H}}(\downarrow, \uparrow)$. Third, the numerical Hamiltonian needs to be consistent with the analytical Hamiltonian, i.e., $\hat{\mathcal{H}}(\mathcal{V}_x, \mathcal{V}_x) = \mathcal{H}(\mathcal{V}_x)$.

Usually, what makes one numerical Hamiltonian better than the other is the degree of numerical dissipation it adds to the problem. Different constructs have been proposed over the years that vary, not only in the degree of dissipation they add, but also how hard they are to implement. One of the easiest ones, though more dissipative, is the Lax-Friedrichs Hamiltonian. In one spatial dimension, this is written as

$$\hat{\mathcal{H}}(\mathcal{V}_x^+, \mathcal{V}_x^-) = \mathcal{H}\left(\frac{\mathcal{V}_x^+ + \mathcal{V}_x^-}{2}\right) - \frac{1}{2}\alpha_x(\mathcal{V}_x^+ - \mathcal{V}_x^-), \quad (14)$$

where $\alpha_x = \max |\partial\mathcal{H}/\partial\mathcal{V}_x|$. If the maximum is computed globally through the whole computational domain, the method is usually termed global Lax-Friedrichs (LF). This, however, is usually unnecessarily too restrictive and the maximum, at any point, is usually computed locally for adjacent grid points and the method is termed local Lax-Friedrichs (LLF).

Although both LF and LLF Hamiltonians are very easy to implement, they usually over-dampen the solution and distort sharp gradients. Better results may be obtained via the Godunov's Hamiltonian. The Godunov's Hamiltonian, unfortunately, is usually hard to obtain for complicated Hamiltonians and may be quite computationally expensive. We do not go into the details of finding these, and other, numerical Hamiltonians and refer the interested reader to Osher and Fedkiw (2003) and Osher and Shu (1991) for more details. Finally we note that to avoid the computational cost of evaluating the Godunov's Hamiltonian, and to obtain solutions that are not overly damped when using the LLF Hamiltonian, one can generally use a

high-order ENO scheme along with a high order TVD Runge-Kutta (see below) scheme and use a sufficiently refined grid.

Appendix C: Total Variation Diminishing (TVD) Runge-Kutta schemes

A TVD Runge-Kutta (TVD-RK) method is merely a Runge-Kutta method that is ensured to decrease the total variation (TV) in the solution as integrated in time. By definition, total variation of a differentiable function, $f(x)$, is defined as,

$$TV(f) = \int |f_x| dx,$$

while for a discrete function, u_j , this definition changes to

$$TV(u) = \sum_j |u_{j+1} - u_j|.$$

In both cases, TV of a function simply is a measure of the amount of variation in the function. To have a convergent solution, one requires a time integration method that decreases the total variation in the function since otherwise it may lead to non-physical oscillatory results. A TVD method is then any time integration scheme that satisfies the following condition:

$$TV(u^{n+1}) \leq TV(u^n).$$

Many different TVD methods exist in the literature, from both the Runge-Kutta (Shu and Osher 1988) and the linear multi-step families (Gottlieb et al. 2001). Without going into much detail, here we present the third-order TVD-RK method and leave other TVD methods to the references mentioned above.

Just like normal Runge-Kutta methods, the third-order TVD-RK method consists of three consecutive forward Euler parts. For the semi-discrete HJ equation, i.e., already discretized in the space variable, x , using the ENO scheme,

$$\frac{\partial \mathcal{V}}{\partial t} = -\hat{\mathcal{H}}(\mathcal{V}_x^+, \mathcal{V}_x^-),$$

the third-order TVD-RK method is written as

$$\mathcal{V}^{n+1/3} = \mathcal{V}^n - \Delta t \hat{\mathcal{H}}^n,$$

$$\mathcal{V}^{n+2/3} = \frac{3}{4}\mathcal{V}^n + \frac{1}{4}\mathcal{V}^{n+1/3} - \frac{1}{4}\Delta t \hat{\mathcal{H}}^{n+1/3},$$

$$\mathcal{V}^{n+1} = \frac{1}{3}\mathcal{V}^n + \frac{2}{3}\mathcal{V}^{n+2/3} - \frac{2}{3}\Delta t \hat{\mathcal{H}}^{n+2/3}.$$

Note that to ensure stability, and the TVD property, it is required to impose a restriction on the time step according to

$$\frac{\Delta t}{h_{\min}} \alpha_{\max} \leq c,$$

where $\alpha_{\max} = \max |\partial \mathcal{H} / \partial \mathcal{V}_x|$ and $h_{\min} = \min |x_{i+1} - x_i|$. These can also be local evaluations (i.e., for each grid point, x_i , within the stencil used to compute \mathcal{V}_x^\pm). Here, $0 < c \leq 1$ is called the CFL number (Osher and Shu 1991; Osher and Fedkiw 2003).

References

- Brown, E., Moehlis, J., Holmes, P. (2004). On the phase reduction and response dynamics of neural oscillator populations. *Neural Computation*, 16, 673–715.
- Caputo, M.R. (2005). *Foundations of dynamic economic analysis: Optimal control theory and applications*. Cambridge: Cambridge University Press.
- Crandall, M.G., & Lions, P.L. (1984). Two approximations of solutions of Hamilton-Jacobi equations. *Mathematics of Computation*, 43(167), 1.
- Danzl, P., Hespanha, J., Moehlis, J. (2009). Event-based minimum-time control of oscillatory neuron models. *Biological Cybernetics*, 101, 387–399.
- Danzl, P., Nabi, A., Moehlis, J. (2010). Charge-balanced spike timing control for phase models of spiking neurons. *Discrete and Continuous Dynamical Systems Series A*, 28, 1413–1435.
- Dasanayake, I., & Li, J.-S. (2011). Optimal design of minimum-power stimuli for phase models of neuron oscillators. *Physical Review E*, 83, 061916.
- Feng, X.J., Shea-Brown, E., Greenwald, B., Kosut, R., Rabitz, H. (2007a). Optimal deep brain stimulation of the subthalamic nucleus - a computational study. *Journal of Computational Neuroscience*, 23, 265–282.
- Feng, X.J., Greenwald, B., Rabitz, H., Shea-Brown, E., Kosut, R. (2007b). Toward closed-loop optimization of deep brain stimulation for Parkinson's disease: concepts and lessons from a computational model. *Journal of Neural Engineering*, 4, L14–21.
- Gottlieb, S., Shu, C.-W., Tadmor, E. (2001). Strong stability-preserving high-order time discretization methods. *SIAM Review*, 43, 89.
- Guckenheimer, J. (1975). Isochrons and phaseless sets. *Journal of Mathematical Biology*, 1, 259–273.
- Harten, A., Engquist, B., Osher, S., Chakravarthy, S. (1987). Uniformly high order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics*, 303, 231–303.
- Hespanha, J. (2007). An introductory course in noncooperative game theory. Available at <http://www.ece.ucsb.edu/~hespanha/published>. Accessed Oct 2011
- Hodgkin, A.L., & Huxley, A.F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology*, 117, 500–544.
- Honeycutt, R.L. (1992). Stochastic runge-kutta algorithms. I. white noise. *Physical Review A*, 45, 600–603.
- Johnston, D., & Wu, S. M.-S. (1995). *Foundations of cellular neurophysiology*. Cambridge, MA: MIT Press.

- Keener, J., & Sneyd, J. (1998). *Mathematical physiology*. New York: Springer.
- Kirk, D.E. (1970). *Optimal control theory: an introduction*. Dover Publications Inc.
- Kiss, I.Z., Rusin, C.G., Kori, H., Hudson, J.L. (2007). Engineering complex dynamical structures: sequential patterns and desynchronization. *Science*, *316*, 1886–1889.
- Liu, X., Osher, S., Chan, T. (1994). Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, *115*(1), 200–212.
- Mitchell, I. (2007). *A toolbox of level set methods*. Technical Report UBC CS TR-2007-11, University of British Columbia.
- Moehlis, J. (2006). Canards for a reduction of the Hodgkin-Huxley equations. *Journal of Mathematical Biology*, *52*, 141–153.
- Moehlis, J., Shea-Brown, E., Rabitz, H. (2006). Optimal inputs for phase models of spiking neurons. *ASME - Journal of Computational and Nonlinear Dynamics*, *1*, 358–367.
- Nabi, A. & Moehlis, J. (2009). Charge-balanced optimal inputs for phase models of spiking neurons. In *Proceedings of the 2009 ASME dynamic systems and control conference*. Hollywood, CA.
- Nabi, A. & Moehlis, J. (2010). Nonlinear hybrid control of phase models for coupled oscillators. In *Proceedings of the 2010 American Control Conference* (pp. 922–923). MD: Baltimore.
- Nabi, A., & Moehlis, J. (2011a). Single input optimal control for globally coupled neuron networks. *Journal of Neural Engineering*, *8*, 065008. doi:10.1088/1741-2560/8/6/065008.
- Nabi, A., & Moehlis, J. (2011b). Time optimal control of spiking neurons. *Journal of Mathematical Biology*, *64*, 981–1004.
- Nabi, A., Mirzadeh, M., Gibou, F., Moehlis, J. (2012). Minimum energy spike randomization for neurons. In *Proceedings of the 2012 American Control Conference* (pp. 4751–4756). Canada: Montreal.
- Nini, A., Feingold, A., Sloviter, H., Bergman, H. (1995). Neurons in the globus pallidus do not show correlated activity in the normal monkey, but phase-locked oscillations appear in the MPTP model of Parkinsonism. *Journal of Neurophysiology*, *74*(4), 1800–1805.
- Osher, S., & Fedkiw, R. (2003). *Level set methods and dynamic implicit surfaces* (1st ed.). New York: Springer.
- Osher, S., & Sethian, J.A. (1988). Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, *79*(1), 12–49.
- Osher, S., & Shu, C. (1991). High-order essentially nonoscillatory schemes for Hamilton-Jacobi equations. *SIAM Journal on Numerical Analysis*, *28*(4), 907–922.
- Osinga, H., & Moehlis, J. (2010). A continuation method for computing global isochrons. *SIAM Journal on Applied Dynamical Systems*, *9*, 1201–1228.
- Pare, D., Curro'Dossi, R., Steriade, M. (1990). Neuronal basis of the Parkinsonian resting tremor: a hypothesis and its implications for treatment. *Neuroscience*, *35*, 217–226.
- Pontryagin, L., Tririgoff, K.N., Neustadt, L. (1962). *The mathematical theory of optimal processes*. Wiley New York.
- Popovych, O.V., Hauptmann, C., Tass, P.A. (2006). Control of neuronal synchrony by nonlinear delayed feedback. *Biological Cybernetics*, *95*(1), 69–85.
- Schiff, S., Jerger, K., Duong, D., Chang, T., Spano, M., Ditto, W., et al. (1994). Controlling chaos in the brain. *Nature*, *370*(6491), 615–620.
- Schiff, S. (2010). Towards model-based control of Parkinson's disease. *Philosophical Transactions of the Royal Society A*, *368*, 2269–2308.
- Schiff, S., & Sauer, T. (2008). Kalman filter control of a model of spatiotemporal cortical dynamics. *Journal of Neural Engineering*, *5*, 1–8.
- Schöll, E., Hiller, G., Hövel, P., Dahlem, M.A. (2009). Time-delayed feedback in neurosystems. *Philosophical Transactions of the Royal Society A*, *367*, 1079–1096.
- Sethian, J.A. (1999). *Level set methods and fast marching methods* (2nd ed.). Cambridge University Press.
- Shu, C., & Osher, S. (1988). Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, *77*(2), 439–471.
- Shu, C., & Osher, S. (1989). Efficient implementation of essentially non-oscillatory shock-capturing schemes, II. *Journal of Computational Physics*, *83*(1), 32–78.
- Stigen, T., Danzl, P., Moehlis, J., Netoff, T. (2011). Controlling spike timing and synchrony in oscillatory neurons. *Journal of Neurophysiology*, *105*, 2074–2082.
- Tass, P.A. (1999). *Phase resetting in medicine and biology*. New York: Springer.
- Volkman, J., Joliot, M., Mogilner, A., Ioannides, A.A., Lado, F., Fazzini, E., Ribary, U., Llinàs, R. (1996). Central motor loop oscillations in Parkinsonian resting tremor revealed magnetoencephalography. *Neurology*, *46*(5), 1359.
- Wilson, C., Beverlin II, B., Netoff, T. (2011). Chaotic desynchronization as the therapeutic mechanism of deep brain stimulation. *Frontiers in Systems Neuroscience*, *5*, Art. No. 50.
- Winfree, A. (2001). *The Geometry of biological time* (2nd ed.). New York: Springer.