Optimal Control of Inter-Spike Intervals given minimal feedback

aviolov, alongtin, sditlevsen

UofO, KU

November 26, 2012

1. **Problem Background**
   - Leaky Integrate-and-Fire Basics
   - Problem Objective
2. Closed-loop stochastic control - dynamic programing
3. Open-loop Stochastic Control

1. Problem Background
   - Leaky Integrate-and-Fire Basics
   - Problem Objective
2. Closed-loop stochastic control - dynamic programing
3. Open-loop Stochastic Control

1 Problem Background
- Leaky Integrate-and-Fire Basics
- Problem Objective

2 Closed-loop stochastic control - dynamic programing

3 Open-loop Stochastic Control

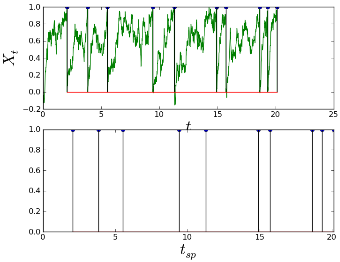
THE UNDERLYING MODEL

A voltage, X_t autonomously decays to zero, is driven and spikes when it hits threshold:

$$dX_s = \left(\alpha(t) - \frac{X_s}{\tau_c}\right)\mathrm{d}s + \beta\,\mathrm{d}W_s,$$

$$X(0) = .0,$$

$$X(t_{\mathrm{sp}}) = x_{th} \implies \left\{ X({t_{\mathrm{sp}}}^{+}) = .0 \right.$$



$\alpha(.) \sim$ the control

$\alpha(t) \in [\alpha_{\min}, \alpha_{\max}]$

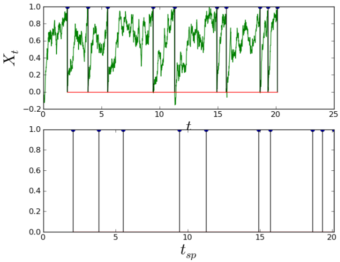
THE UNDERLYING MODEL

A voltage, X_t autonomously decays to zero, is driven and spikes when it hits threshold:

$$dX_s = \left(\alpha(t) - \frac{X_s}{\tau_c}\right)\mathrm{d}s + \beta\,\mathrm{d}W_s,$$

$$X(0) = .0,$$

$$X(t_{\mathrm{sp}}) = x_{th} \implies \left\{ X(t_{\mathrm{sp}}^{+}) = .0 \right.$$



$$\alpha(.) \sim \text{the control}$$
$$\alpha(t) \in [\alpha_{\min}, \alpha_{\max}]$$

OUR GOAL

Impose a pre-specified spike sequence, $\{t_n\}_1^N$, on the neuron.

ASSUMPTION: We know when actual spikes, $\{t_{sp,n}\}$ occur

$\implies$ we can work sequentially:

$$T^* = t_n - t_{sp,n-1}$$

OUR GOAL

Impose a pre-specified spike sequence, $\{t_n\}_1^N$, on the neuron.

ASSUMPTION: We know when actual spikes, $\{t_{sp,n}\}$ occur
$\implies$ we can work sequentially:

$$T^* = t_n - t_{sp,n-1}$$

OUR GOAL

Given a target T^* find $\alpha(t)$ minimizing $\mathbb{E}[(T^* - t_{\rm sp})^2]$

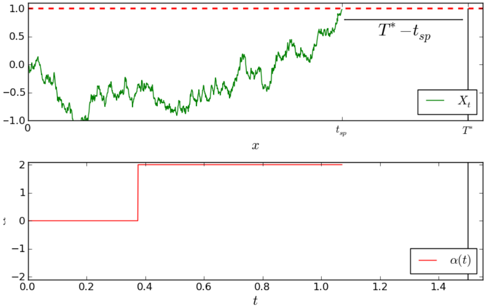


FIGURE: Observed vs. Desired spike time - Controlled Trajectory Example

Formal Problem Statement

Given, τ_c, β, T^* and $X_0 = .0$, find

$$\alpha(t) = \arg\min_{\alpha(t)} \left\{ J[\alpha] = \mathbb{E}\left[\left(t_{\text{sp}} - T^*\right)^2 + \varepsilon \int_0^{t_{\text{sp}}} \alpha^2 \, \mathrm{d}s \right] \right\} \qquad (1)$$

Dynamic Programing Solution

If! we had state observations / feedback

$\Longrightarrow$ Dynamic Programming / Hamilton-Jacobi-Bellman (HJB) Equations

$$\partial_t v(x,t) + \frac{\beta^2}{2}\partial_x^2 v + \frac{(\partial_x v)^2}{\varepsilon} - \frac{x}{\tau_c}\partial_x v = 0$$

$$\alpha(t) = \frac{-\partial_x v(x(t),t)}{2\varepsilon}$$

$$\begin{cases} v(x_{th},t) = (t - T^*)^2 & \text{upper BC} \\ \partial_x v(x_-,t) = .0 & \text{lower BC} \\ v(x,T^*) = \mathbb{E}[(\tau^2 : X_{T^*+\tau} = x_{th}) | X_{T^*} = x, \alpha(t) = \alpha_{\max}] & \text{TC} \end{cases}$$

Dynamic Programing Solution

If! we had state observations / feedback
$\Longrightarrow$ Dynamic Programming / Hamilton-Jacobi-Bellman (HJB) Equations

$$\partial_t v(x,t) + \frac{\beta^2}{2}\partial_x^2 v + \frac{(\partial_x v)^2}{\varepsilon} - \frac{x}{\tau_c}\partial_x v = 0$$

$$\alpha(t) = \frac{-\partial_x v(x(t),t)}{2\varepsilon}$$

$$\begin{cases} v(x_{th},t) = (t - T^*)^2 & \text{upper BC} \\ \partial_x v(x_-,t) = .0 & \text{lower BC} \\ v(x,T^*) = \mathbb{E}[(\tau^2 : X_{T^*+\tau} = x_{th}) | X_{T^*} = x, \alpha(t) = \alpha_{\max}] & \text{TC} \end{cases}$$

Dynamic Programing Solution

If! we had state observations / feedback

$\Longrightarrow$ Dynamic Programming / Hamilton-Jacobi-Bellman (HJB) Equations

$$\partial_t v(x,t) + \frac{\beta^2}{2}\partial_x^2 v + \frac{(\partial_x v)^2}{\varepsilon} - \frac{x}{\tau_c}\partial_x v = 0$$

$$\alpha(t) = \frac{-\partial_x v(x(t),t)}{2\varepsilon}$$

$$\begin{cases} v(x_{th},t) = (t-T^*)^2 & \text{upper BC} \\ \partial_x v(x_-,t) = .0 & \text{lower BC} \\ v(x,T^*) = \mathbb{E}[(\tau^2 : X_{T^*+\tau} = x_{th})|X_{T^*} = x, \alpha(t) = \alpha_{\max}] & \text{TC} \end{cases}$$

But we do not!

Simplest solution $\sim$ ignore the noise!
$\Longrightarrow$ Deterministic Optimal Control (in 1-D)

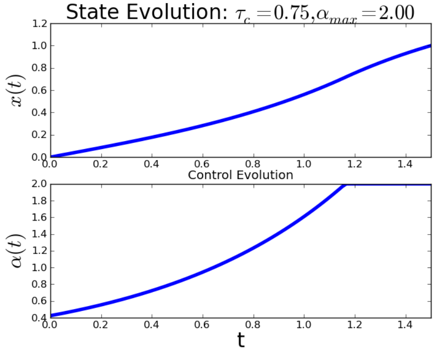


FIGURE: An example trajectory for the deterministic dynamics.

AN UNFAIR COMPARISON

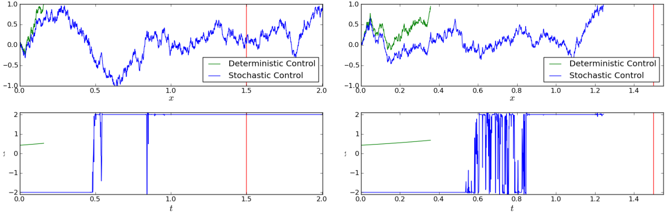


FIGURE: Controlled trajectories using deterministic vs. stochastic control approaches. T^* is indicated with the red vertical line, $\tau_c, \beta = [.75, 1.25]$.

Can we do better?

An Unfair Comparison

Control Law	Squared Error
Open-Loop Deterministic	0.647
Closed-Loop Stochastic	0.336
Theory (HJB Value function at $(0,0)$)	0.285

TABLE: Realized performance of the different control laws and the theoretical performance of the stochastic law (last row)

Can we do better?

An Unfair Comparison

Control Law	Squared Error
Open-Loop Deterministic	0.647 $\swarrow$
Closed-Loop Stochastic	0.336 $\nwarrow$
Theory (HJB Value function at $(0,0)$)	0.285

TABLE: Realized performance of the different control laws and the theoretical performance of the stochastic law (last row)

Can we do better?

Consider, $f(x,t)$, the forward transition density for X_t

then

$$\mathbb{E}\left[\left(t_{\mathrm{sp}} - T^*\right)^2\right] = \int_{x_-}^{x_{th}} \mathbb{E}[\tau^2 | x, \alpha_{\max}] \cdot f(x, T^*)\,\mathrm{d}x + \int_0^{t_{\mathrm{sp}}} (s - T^*)^2 \cdot \left(-\frac{\beta^2}{2} \partial_x f(x_{th}, s)\right) \mathrm{d}s$$

and

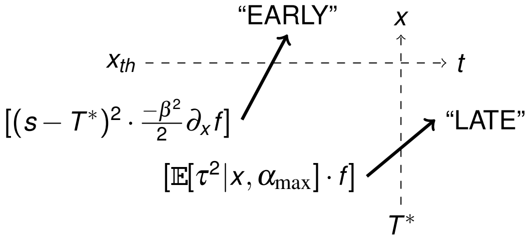
$f(x,t) \sim$ deterministic (it satisfies a Fokker-Planck equation)

$\implies$ optimize using f

But how do we optimize with PDEs?

Consider, $f(x,t)$, the forward transition density for X_t

then



and

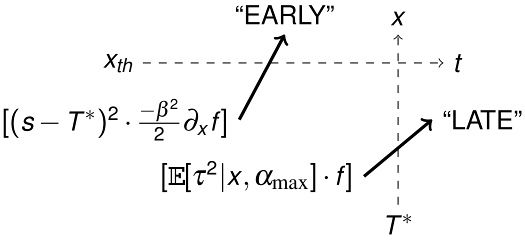
$f(x,t) \sim$ deterministic (it satisfies a Fokker-Planck equation)

$\implies$ optimize using f

But how do we optimize with PDEs?

Consider, $f(x,t)$, the forward transition density for X_t

then



and

$f(x,t) \sim$ deterministic (it satisfies a Fokker-Planck equation)

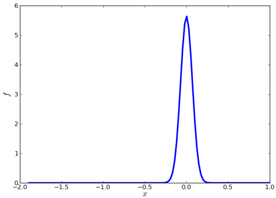
$\implies$ optimize using f

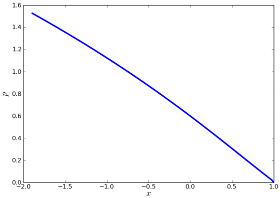
But how do we optimize with PDEs?

AN INFINITE-DIMENSIONAL MINIMUM PRINCIPLE

f

FORWARD





ADJOINT

p

$$\left\{ \varepsilon 2 \cdot \alpha(t) + pf\Big|_{x_-} - \int_{x_-}^{x_{th}} p \cdot \partial_x f \,\mathrm{d}x \right\} = 0 \quad \forall t \in [0, T]$$

Gradient Descent

$$\nabla_\alpha [J] = \varepsilon 2 \cdot \alpha(t) + pf\Big|_{x_-} - \int_{x_-}^{x_{th}} p \cdot \partial_x f \, \mathrm{d}x$$

We can use $\nabla_\alpha J$ for gradient-based numerical optimization

A MORE FAIR COMPARISON

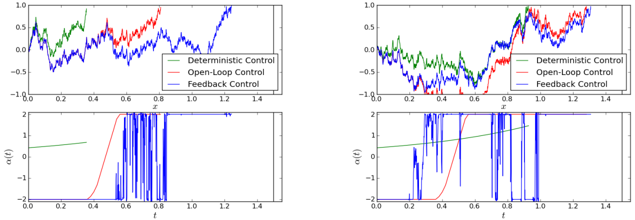


FIGURE: Controlled trajectories using deterministic vs. open-loop stochastic vs. closed-loop stochastic control laws. T^* is indicated with the red vertical line, $\tau_c, \beta = [.75, 1.25]$.

A more fair comparison

Control Law	Squared Error
Open-Loop Deterministic	0.621
Open-Loop Stochastic	0.356
Closed-Loop Stochastic	0.288
Open-Loop Theory (J)	0.382
Closed-Loop Theory (Value function)	0.285

TABLE: Realized performance of the different control laws and the theoretical expected performance for the stochastic laws (last 2 rows)

SUMMARY

HAS BEEN SHOWN:

- A stochastic control problem with minimal observations can be re-stated as a deterministic optimal control problem using the transition density.
- The deterministic optimal control problem can be solved using an infinite-dimensional Pontryagin-type Principle and a simple gradient descent procedure.

TO BE ADDRESSED:

- Can this be done very fast (real-time for real neurons)?
- Can this be done for several coupled neurons simultaneously?