An Optimal Control Model of Zebra Finch Vocalization

Mike Schachter [*]
Helen Wills Neuroscience Institute
University of California, Berkeley
Berkeley, CA 94720
mike.schachter@gmail.com

Abstract

In this work, a nonlinear oscillator modeling the syringeal folds of the Zebra Finch is controlled by the state of higher level linear dynamical system. We formulate an optimal control cost function and solution for the learning of bird song.

1   Introduction

First we'll discuss the Zebra Finch vocalization system and it's mathematical formulation, and then the control of that system. All code used to generate figures and results in this paper can be found at:

http://github.com/mschachter/birdy

1.1   The Zebra Finch Vocalization System

Like human speech, Zebra Finch song is generated at it's source by oscillations in airflow generated by vibrating vocal cords. A model for these oscillations is given as:

$$\dot{x} = v$$
$$\dot{v} = \gamma^2\alpha + \gamma^2\beta x - \gamma^2 x^3 - \gamma x^2 v + \gamma^2 x^2 - \gamma x v$$

The control parameters of the model are α and β. $\gamma = 23500$ is a constant.

1.2   Control of the Syrinx Model

The goal is to control the parameters α and β in order to produce an observed vocalization. Let $\phi(t) = [\alpha(t)\ \beta(t)]^T$ be the state vector.

Assume that the temporal evolution of $\phi(t)$ is defined by a controlled linear dynamical system:

$$\dot{\phi} = A\phi(t) + \boldsymbol{u}(t)$$

where A is a 2x2 matrix, chosen to make the passive dynamics of the control system decay to some physiologically relevant rest state.

---

[*]With a metric ton of help and code from Hédi Soula: hsoula@gmail.com

We discretized time to simplify the analysis. Let $\Delta\tau$ be the time step for simulation of the control system, and define $t_k = k\Delta\tau$. Also define $\phi_k = \phi(t_k)$. Note that the time step for the simulation of the control is significantly larger than that of the oscillator. The discrete time map representing the control system is given by the forward Euler step:

$$
\begin{aligned}
\phi_{k+1} &= (A\phi_k + u_k)\,\Delta\tau + \phi_k \\
&= (\Delta\tau A + I)\,\phi_k + \Delta\tau u_k
\end{aligned}
$$

In optimal control theory, a cost function is specified and is to be minimized over time to produce an optimal control law. But what form should the cost function take? We'll make the following assumptions:

1. The control wants to keep the system's instantaneous energy low: $\phi_k^T \phi_k$.

2. The control wants to keep it's own instantaneous energy low: $u^T u$

3. The control wants to produce an instantaneous fundamental frequency that matches that of a stored template.

To elaborate on the last assumption, say we are given the time-varying fundamental frequency of a song syllable that we would like to learn, represented as the function $F(t_k)$. Let $\hat{F} = g(\phi_k)$ be the function that gives the steady state fundamental frequency for a given control ϕ_k. The function g can be empirically estimated through simulation and approximated through interpolation. To follow this frequency means to keep the quantity $(F(t_k) - g(\phi_k))^2$ small.

Given these assumptions, the cost at time t_k of applying control u is:

$$
\ell_k(\phi_k, u) = \phi_k^T \phi_k + u^T u + (F(t_k) - g(\phi_k))^2
$$

Let N be the number of time points we want to control, and let $\pi_j = \{u_j(\phi_j), ..., u_N(\phi_N)\}$ be the control law applied from time j to time N. Note that the control law is a sequence of functions! Each control uses feedback information about most recently observed state. The total cost of applying a control law π_0 given initial state ϕ_0 is:

$$
v(\phi_0, \pi_0) = \sum_{k=1}^{N} \ell_k(\phi_k, u_k)
$$

1.3 Dynamic Programming Solution to Optimal Control

The optimal cost-to-go is a function that represents the total cost from a time j to time N given that the optimal control function is applied:

$$
v_j^*(\phi_j) = \min_{\pi_j} \sum_{k=j}^{N} \ell_k(\phi_k, u_k)
$$

Dynamic programming is typically used to solve for the optimal cost-to-go. When applying DP, we work backwards in time, creating a recursive algorithm that determines the functional form of the entire optimal control policy π_0^*. To illustrate this, we are going to take $N = 2$ with a specified initial condition ϕ_0.

First the solution must be found for $N = 2$:

$$
v_2^*(\phi_2) = \min_{u_2} \ell_2(\phi_2, u_2)
$$

In order to solve this, we have to perform a search over a grid of values for ϕ_2. For each value on the grid, we then will use gradient descent to minimize $v_2^* (\phi_2)$ and find an optimal $\boldsymbol{u}_2$. The end result is that we have a lookup table of values, giving us a function $\boldsymbol{u}_2(\phi_2)$, and a function $v_2^* (\phi_2)$.

The optimal cost-to-go at time $N - 1 = 1$ is

$$v_1^* (\phi_1) = \min_{\boldsymbol{u}_1} \left[\ell_1 (\phi_1, \boldsymbol{u}_1) + v_2^* (\phi_2) \right]$$

We know the functional form of $\ell_1 (\phi_1, \boldsymbol{u}_1)$ and the numerical form of $v_2^* (\phi_2)$. We also know that:

$$\phi_2 = (A\phi_1 + \boldsymbol{u}_1) \Delta\tau + \phi_1$$

so we can again use gradient descent for a grid on ϕ_1 to create a lookup table for $\boldsymbol{u}_1(\phi_1)$ and $v_1^* (\phi_1)$. We now have a functional form for the optimal control policy: $\boldsymbol{\pi}_0^* = \{\boldsymbol{u}_1(\phi_1), \boldsymbol{u}_2(\phi_2)\}$

Now going foward in time, with the knowledge of a start state ϕ_0, we can then compute ϕ_1, and then the optimal control $\boldsymbol{u}_1 (\phi_1)$. Given $\boldsymbol{u}_1$, we can then compute ϕ_2 and then $\boldsymbol{u}_2$. The techinque, albeit expensive, is generally applicable for any N.

1.4 Temporal Hierarchy of Representation

Generically, we want to determine a joint probability between ϕ and a set of variables that are related to the acoustic representation. These features may span several time scales. For example, instead of just relating $F(t)$ and ϕ, i.e. looking at the joint distribution $p(F(t), \phi)$, we might want to look at the running variance of $F(t)$ for a specified time window.

Let $\{\sigma_1, ..., \sigma_m\}$ be a set of statistics for some acoustic variables. Construct this set so that they are ordered by time scale. By this, we mean that computing σ_i requires a larger window of time than computing σ_j if $i < j$.

Finding the optimal control requires maximizing a conditional probability:

$$\operatorname*{argmin}_{\boldsymbol{u}} C_f = \operatorname*{argmax}_{\phi(\boldsymbol{u})} p(\phi(\boldsymbol{u})|\sigma_1, ..., \sigma_m)$$

There may be a "telescoping" algorithm to maximizing $p(\phi(\boldsymbol{u})|\sigma_1, ..., \sigma_m)$ quickly. Start with σ_1, which is the statistic with the longest time scale. There should be many instantaneous values of $\boldsymbol{u}(t)$ that give a nonzero probability of occurance with σ_1. But not all the values will, so restrict the search for all $\boldsymbol{u}$ to that space. Do the same for σ_2, which will reduce the size of the space even further. Continue this process until the space of actual $\boldsymbol{u}$ is small enough to do a more efficient optimization, and then peform that optimization to find the optimal control.

Acknowledgments

References