A different view of gestural activation: learning gestural parameters and activation with an RNN

Sam Tilsen

June 20, 2020

Abstract

The gestural scores of the Articulatory Phonology / Task Dynamics model are useful for conceptualizing how speakers control the geometry of the vocal tract. However, the parameters and inputs of the model—gestural targets, stiffnesses, and activation states—cannot be measured directly. Instead, they must be estimated from empirical data, and this estimation requires assumptions about which gestures occur, when they occur, what their targets are, and how their activation functions may vary over time. I present a new approach to learning gestural parameters and activation states using a recurrent neural network that takes gestural activation as input and that outputs tract variable positions. The network can be trained to generate tract variable trajectories which closely fit empirical data. The training is accomplished using backpropagation through time to adjust both gestural activation functions and target/stiffness parameters—hence gestural activation functions are treated as learnable parameters of the model. An important consequence of the approach is that it does not impose temporal bounds on gestures.

Introduction

A foundational concept in the framework of Articulatory Phonology / Task Dynamics is the concept of an *articulatory gesture*. But what *IS* a gesture? The term is used in a variety of ways: sometimes it is a movement, sometimes a task; sometimes it is a system that exerts a force, sometimes it is the period of time in which a force is exerted. Sometimes it is an elemental unit of a communicative code. In a basic sense, gestures are *events*, and crucially, this entails that gestures have *temporal bounds*.

What does it mean for an event to be "bounded" in time? We usually think of time metaphorically as a 1-dimensional space, as shown in Fig. 1A. Gestures are associated with finite regions of this space. When we say that a gesture occurred *here* or *there* in time, it entails that the spatial location is *delimited*: the gesture had a beginning and end:

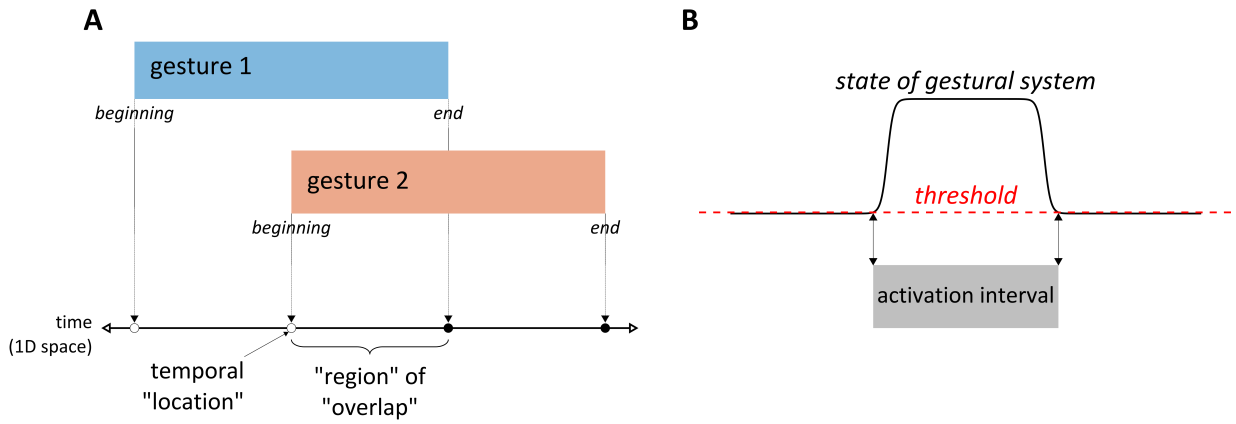

Fig. 1. Comparison of event and system conceptions of gestures. (A) Event conception in which gestures are events with temporal bounds. (B) System conception in which gestures are systems with an activation state variable.

Crucially, it is possible in the standard picture of Fig. 1A to say that two gestures do or do not "overlap", by which it is meant that the beginning of one precedes the end of the other. This sort of entailment is not possible without an image schema in which there are identifiable temporal "locations" or "landmarks" which bound the periods of time associated with gestures.

What is happening in these *periods of time*? The Task Dynamic (TD) model (Saltzman & Munhall, 1989, henceforth SM89) tells us that during the period of time associated with a gesture (called an *activation interval*), the gesture exerts a force on the state of the vocal tract (more on these forces later). Specifically, the activation interval of a gesture corresponds to the time when its activation state is greater than a threshold value of zero, as shown Fig. 1B.

Yet there is nothing fundamental to the equations of the SM89 TD model that requires us to impose an activation threshold. Furthermore, the idea that a gesture is a temporally bounded event is entirely a consequence of imposing the threshold: the SM89 model in actuality defines gestural activation states for all gestures at all points in time. Hence we can also think of gestures not as bounded events, but rather as *systems* which are continuously exerting forces on the vocal tract state. In this view, gestures do not have beginnings or ends, and they are not even the sorts of things that can have beginnings or endings.

Why would we want to think of gestures as systems? In general, we want to be able to estimate quantities of our theoretical models from empirical data. This estimation process will always require some arbitrary decisions about how information is reduced in mapping from empirical data to model quantities. I argue that the advantages of the systems-view of articulatory gestures are that it (i) allows us to

reinterpret some of the arbitrariness in the estimation process, and (ii) facilitates the analysis of information that might otherwise be ignored.

Rethinking the estimation procedure

The standard approach to estimating gestural "onsets" and "offsets" is illustrated in Fig. 2A. The top panel shows a lip aperture time series from a single trial of an experiment. In the standard approach we use extremal values of first or second derivatives (i, ii) or thresholds of derivatives (iii, 20% of max velocity) of the tract variable time series. These are then taken as estimates of the beginnings and ends of gestural activation intervals. This procedure—a transformation from the tract variable to the activation interval signal—greatly reduces information. This reduction is generally a good thing, because the tract variable time series itself contains too much information to be useful as the basis of a communicative code. In general we want our theoretical conceptualization to be low-dimensional.

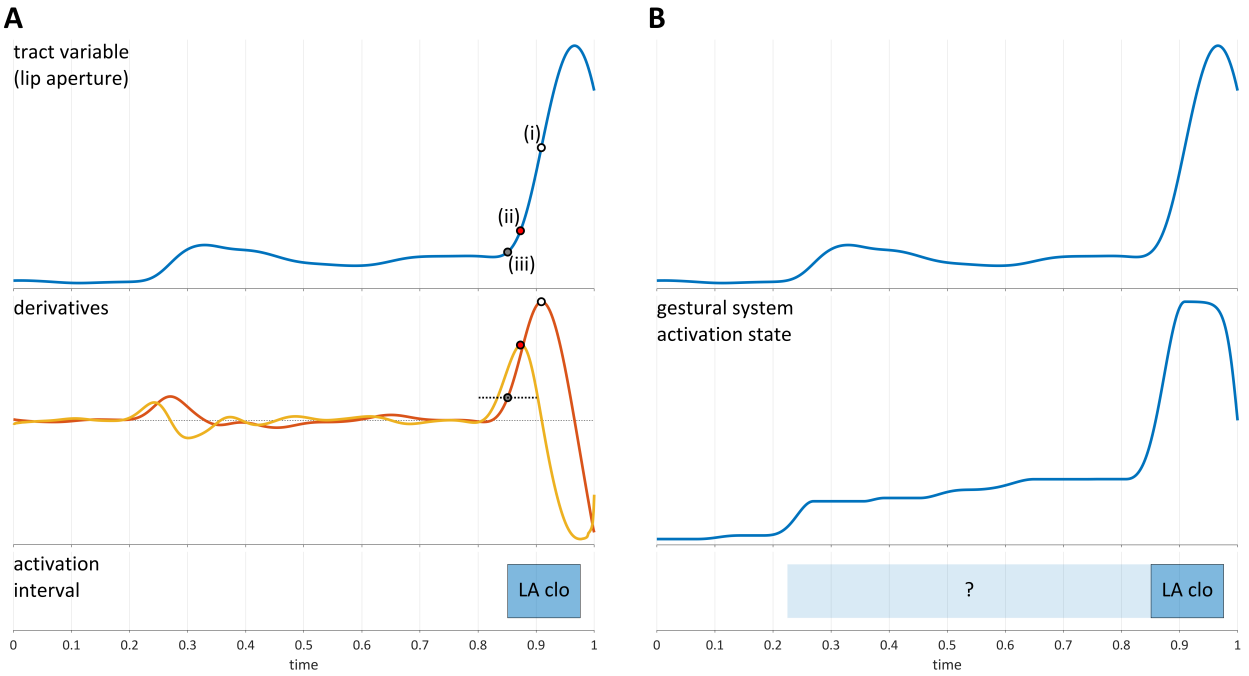

Fig. 2. Comparison of approaches to estimating gestural onsets. (A) Standard approach. (B) Use of gestural system activation state as an intermediate representation.

In the alternative approach proposed here Fig. 2B, we optimize a continuously varying gestural activation function (i.e. *a gestural system state trajectory*) to generate the empirically observed values of the tract variable. This signal may have similar information content to the original tract variable, but it is a quantity of our theoretical model. It has features which may of be interest to us, such as the subtle transient increase around a time of 0.2 s, or the presence of an extended non-zero component before the global acceleration maximum. Subsequently, if we want to recover the standard gestural "onset" and "offset" landmarks, these can be derived via extrema identification or thresholding of derivatives of the gestural activation function, rather than from the tract variable. Crucially, the interpretation of the landmarks is necessarily different: instead of being the beginnings and ends of gestural activation intervals, they delimit intervals of time when there is a *relatively strong* gestural force exerted on a tract variable. The qualifier *relatively* reinforces the idea that these intervals may have no special status and depend on somewhat arbitrary decisions.

Both approaches result in a similar reduction of information from the tract variable to the interval signal. In the standard approach, the reduction occurs because we have arbitrarily presupposed an "activation interval" and used this to transform the tract variable signal. In the alternative approach, the reduction occurs because we have arbitrarily imposed criteria on the gestural activation function in order to demarcate intervals of time with relatively strong gestural forces. The difference is that in the standard approach, some of the arbitrariness in the estimation process is associated with the presupposition of the theoretical model that gestures are bounded events; in contrast, in the alternative, the arbitrariness derives from a decision to identify periods of time in which there is a strong gestural force. In this sense, the interpretation of the arbitrariness is quite different. Moreover, by deriving an intermediate signal of theoretical interest—the gestural activation—we allow for new opportunities of analysis.

To think of gestural delimitation as arbitrary is entirely consistent with the Task Dynamic model. For instance, SM89 justified their use of step functions of activation on the basis of simplicity, and did not offer any further rationale for that decision. Indeed, they stated a plan "to generalize the shapes of the activation waves and to allow activations to vary continuously". The step function (black line in Fig. 3A) is a transition from 0 activation to maximal activation, this transition being associated with the initiation of gesture. The same applies to the transition from maximal activation to 0 activation, associated with termination of a gesture (not shown). Some obvious generalizations involve linear or logistic/sigmoidal ramping, also illustrated in Fig. 3A. It does not seem possible to empirically resolve which of these is correct, because with appropriate parameterization, all of them can be optimized to generate the velocity profiles that we observe empirically. Moreover, the effects of these functional forms depend on the structure of the TD model, which is also not a uniquely correct solution to the problem of generating speech movements. The mere fact that we do not have principled ways of differentiating between these alternatives calls into question whether gestures should be conceptualized in this way.

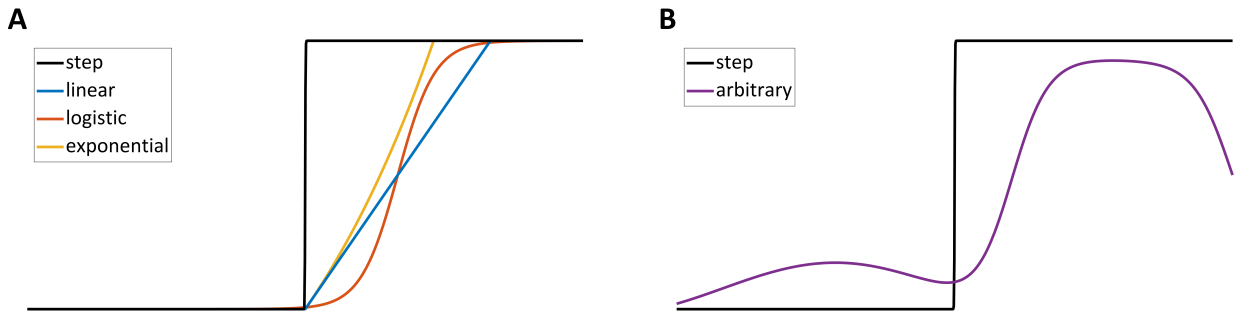

Fig. 3. Examples of gestural activation dynamics. (A) comparison of activation functions. (B) an arbitrary activation function.

Reasoning about activation from the microscale

How can we motivate our decisions regarding the forms of activation functions? One approach is to develop additional hypotheses about the nature of gestures. From the systems perspective, one such hypothesis is that the entities we call *gestural systems* correspond to neural populations in premotor cortex. In this view, gestural activation is a macroscale variable that integrates over many microscale variables, which are the spiking rates of the neurons in the populations. The integration is assumed to apply over time scales that are long enough to result in smooth activation functions, yet short to enough to capture variation relevant to gestural control of the vocal tract. This hypothesis has been elaborated in detail in previous work (Tilsen, 2017, 2018, 2019b).

Given the above multiscale conception of gestures, we can motivate an argument that gestural activation cannot have functional forms of the sort in Fig. 3A, and instead we should allow for more general, arbitrary functional forms. The argument is as follows. Neuronal spiking occurs when the membrane potential of a neuron reaches a voltage threshold, and generally neuronal membrane potentials are maintained near this threshold. Because of this, small electrochemical fluctuations in the surroundings or in the states of pre-synaptic neurons can conspire to induce action potentials. If we assume that (under standard environmental conditions[1]) the amplitudes of stochastic fluctuations in the population are sufficiently large, then the spiking rate is never actually zero—it can only be very close to 0. Hence gestural activation is never 0. Indeed, if we assume that the population interacts sufficiently strongly with other populations (e.g. lexical/conceptual systems), it is fair to assume that the spiking rate may sometimes be a substantial proportion of its maximum, even when there may be no readily apparent constriction task associated with a gesture.

The presence of stochastic fluctuations leads to a conundrum for our temporally delimited concept of a gesture—we cannot use 0 as our threshold for determining when a gesture begins and ends, because gestural activation will always be above 0, in the multiscale conception. Instead we must impose some non-zero threshold. Yet this opens the door to the possibility that we might allow for more complicated forms of "below-threshold" variation in activation, as in Fig. 3B. Ultimately this begs the question of why we would impose a threshold in the first place. Since the threshold is not essential to the TD model, its use seems to be driven entirely by a desire—perhaps aesthetic in nature—to delimit gestures in time.

A less presumptive conceptualization of the TD model is one in which there is no inherent delimitation; instead, there is just a set of gestural systems and tract variable systems, both continuously varying in time. The gestural systems exert forces on the tract variable systems, and these forces influence the states of tract variables and are modulated by the states of gestural systems. In the language of SM89, gestural activation "serves to define or 'tune' the current set of dynamic parameter values in the model" and "can be interpreted as the strength with which the associated tract-variable dynamical system attempts to shape vocal-tract movements at any given point in time". The parameters referenced in these quotes describe damping and elastic forces on a tract variable system. SM89 also cite the similarity of this *tuning* conception to an idea from Fowler (1983), where segmental "prominence" relates to the "extent to which vocal tract activity is given over to the production of a particular segment" (1983: 392).

To summarize, we have so far argued that the concept of a temporally delimitation of gestures is merely an aesthetic appendage. Perhaps its popularity is attributable to its facilitation of an image schema in which speech is viewed a spatial arrangement of objects in time, which is a hallmark of segmental phonological representations. The alternative schema which rejects delimitation is one in which there is a continuous trajectory (or flow) of gestural system states in a gestural activation state space, driving a continuous trajectory of tract variable states in a tract variable state space. This continuous activation flow alone may not be entirely adequate for reasoning about a communicative code: we *do* seem to observe that in many cases there are rapid changes in our measures of vocal tract states—i.e. movements—and it is quite valuable to localize these in time. But this localization is ultimately arbitrary and does entail the existence of entities with temporal bounds.

The RNN approach

The RNN implementation of gesture and tract variable dynamics developed here is inspired by recent observations that layer skipping in residual networks can be viewed as one step of Euler's numerical method for solving differential equations (Chen et al., 2018; Dupont et al., 2019; Weinan, 2017). In the limit of small time steps, such networks can be used to provide an accurate numerical solution for a system

[1] The speaker is alive and not in cryogenic stasis.

of differential equations. The model presented below uses this idea to implement a neural network which generates the dynamics of the forced, damped mass-spring equations for tract variables in the Task Dynamic model. Backpropagation can be used to "learn"—i.e. optimize—gestural targets and stiffnesses, and even gestural activation functions, by treating them as high-dimensional parameters. Before describing the TD RNN model, we first examine a toy example using a simpler system.

How can an RNN solve a differential equation?

To illustrate how an ODE can be "solved" by an RNN, we use the non-autonomous ODE in (Eq. 1) as an example, where $D(t) = sin(t)$ is a time-dependent input to the system, and $x(t)$ is the state variable of the system.

$$\dot{\mathrm{x}} = \frac{\mathrm{dx}}{\mathrm{dt}} = -\mathrm{x} + \mathrm{D(t)}, \qquad \text{where } D(t) = \sin(t) \tag{Eq. 1}$$

With initial condition x_0, this system has the following analytic solution:

$$x(t) = \left(\frac{1}{2} + x_0\right) \mathrm{e}^{-\mathrm{t}} - \frac{\sqrt{2}cos\left(t + \frac{pi}{4}\right)}{2} \tag{Eq. 2}$$

A numeric solution can be obtained for times $\mathrm{t} = 0, \mathrm{dt}, \ldots, \mathrm{N}$ using Euler's method, where we specify a value for initial condition and iteratively add $\frac{dx}{dt}dt$ to the previous timestep, as illustrated in the Matlab code below. As shown in Fig. 4, the numeric solution is very close to the analytic solution. The system has a decaying transient due to the exponential decay, and a harmonic oscillation from the $\sin(t)$ input.

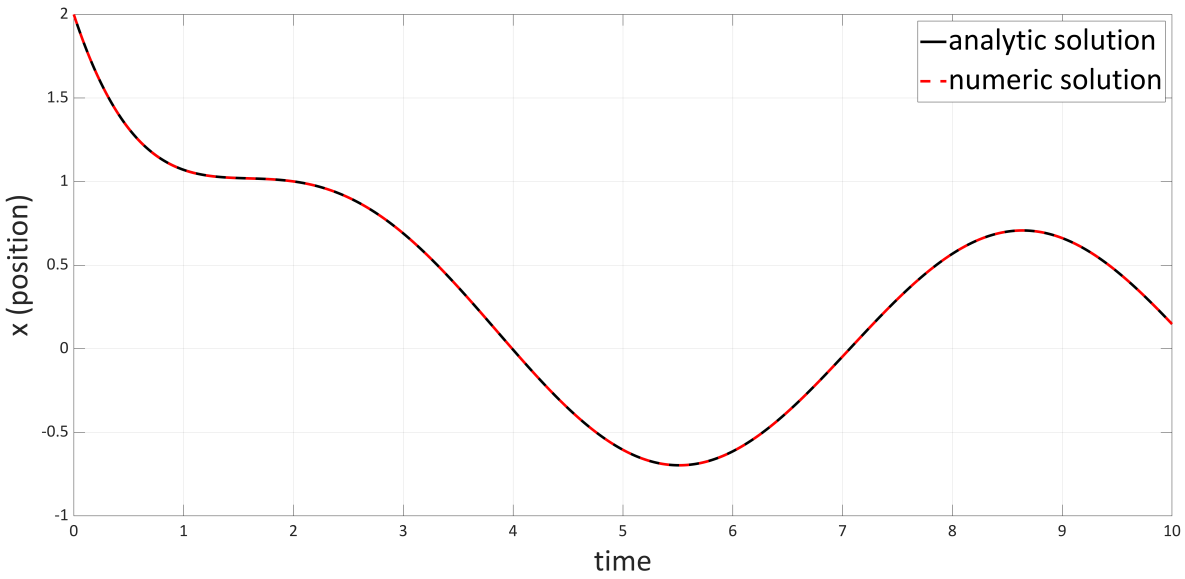

Fig. 4. Analytic and numeric solutions of the system in (Eq. 1).

The code snippet below shows how Euler's method can be implemented. Here t is a vector of times which is indexed by *i.* Observed that $-\mathrm{x}(i) + sin\, t(i)$ in the solution loop is the right hand side of (Eq. 1) and is

the slope of the tangent line at time $t(i)$. By multiplying this slope by a small time step dt and adding it to $x(i)$ we obtain an approximation of $x(i+1)$.

```
dt = 0.001; %time step
t = 0:dt:20;

x = zeros(size(t));
x(1) = 2; %initial condition

%solution
for i=1:length(t)-1
    x(i+1) = x(i) + ( -x(i) + sin(t(i)) ) *dt;
end
```

We can interpret Euler's numeric solution of the system as the RNN in Fig. 5A, where the state variable is the output layer, the $\sin(t)$ term is an input layer, and the $-x$ term is a recurrent connection from the output layer to itself at the next time step. The figure also shows a node that represents the loss at each timestep, $L(t)$. If we think of the analytic solution from (Eq. 2) as $y(t)$, a target output of the solution, then $L(t)$ is half the squared difference of the analytic solution and numeric solution. If the timestep is sufficiently small and the system is non-stiff, the loss will be small as well. A common convention in depicting RNNs is to "unfold" the network, as in Fig. 5B. In this depiction, each time step is associated with a "copy" of the RNN, and these copies are viewed as layers of a potentially very deep feed-forward network.

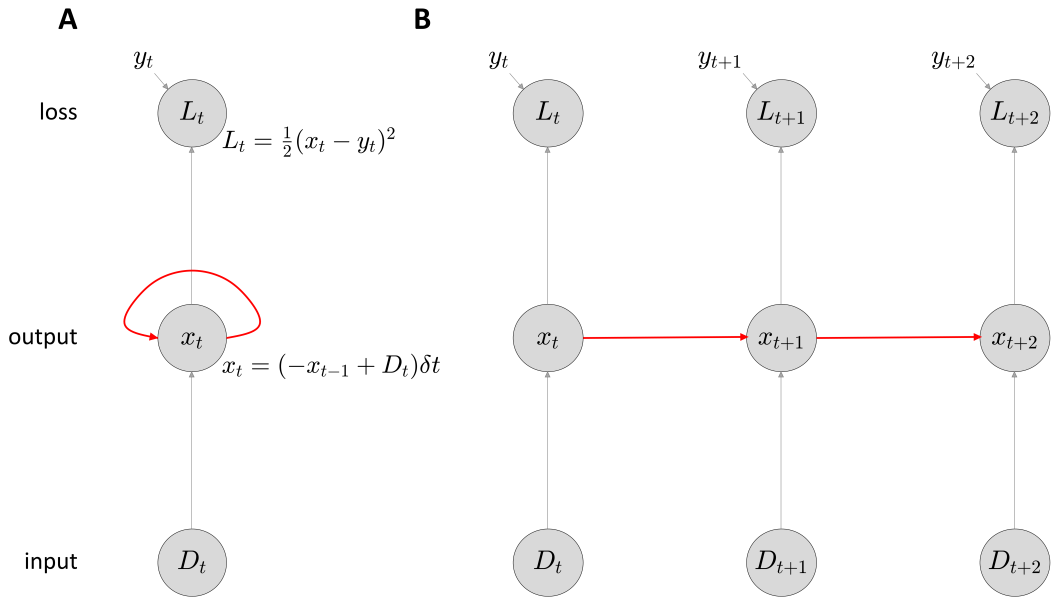

Fig. 5. RNN implementation of Euler's method. (A) RNN with input layer D(t), output layer x(t), recurrent connection from the output layer to itself, and loss L(t). (B) Unfolded network.

Learning ODE input with an RNN

In the above example, we assumed that we knew the input to the system, $D(t)$. But what if we do not know the input? In this case, we can use the network to optimize the input. To do this, we think of the input at each time step as a separate parameter, and we use backpropagation (i.e. gradient descent) to find an input which minimizes the loss. Why might this be useful? Think of an empirically observed tract variable time series as a target output, and consider that the TD model generates these outputs with gestural activation as input. We can use a network to infer gestural activation functions from empirical data.

Before developing the TD RNN model, let us consider how backpropagation through time works for the toy example above. We begin by making a guess at the input. Our guess will be $D(t) = 0$, i.e. there is no input. We will also set an initial condition $x_0 = y_0$, i.e. the initial state of the system is the initial value of the target output. We run a forward pass of the network, which is almost identical to Euler's method shown above, except that we replace the known driving force $\sin(t)$ with our guess $D(t) = 0$. The total loss is shown in (Eq. 3), and the error at each time step is shown in (Eq. 4):

$$L = \sum_{t=1}^{N} \frac{1}{2}[x(t) - y(t)]^2 \quad \text{(Eq. 3)}$$

$$E(t) = x(t) - y(t) \quad \text{(Eq. 4)}$$

The target output, network output, and error are shown in Fig. 6. Note that the units of *x* and the input are arbitrary in this example. Clearly the initial guess for the input was not very good, since the error is nearly as large in magnitude as the target output. We need to figure out how much to adjust the input at each time step to decrease the total loss.

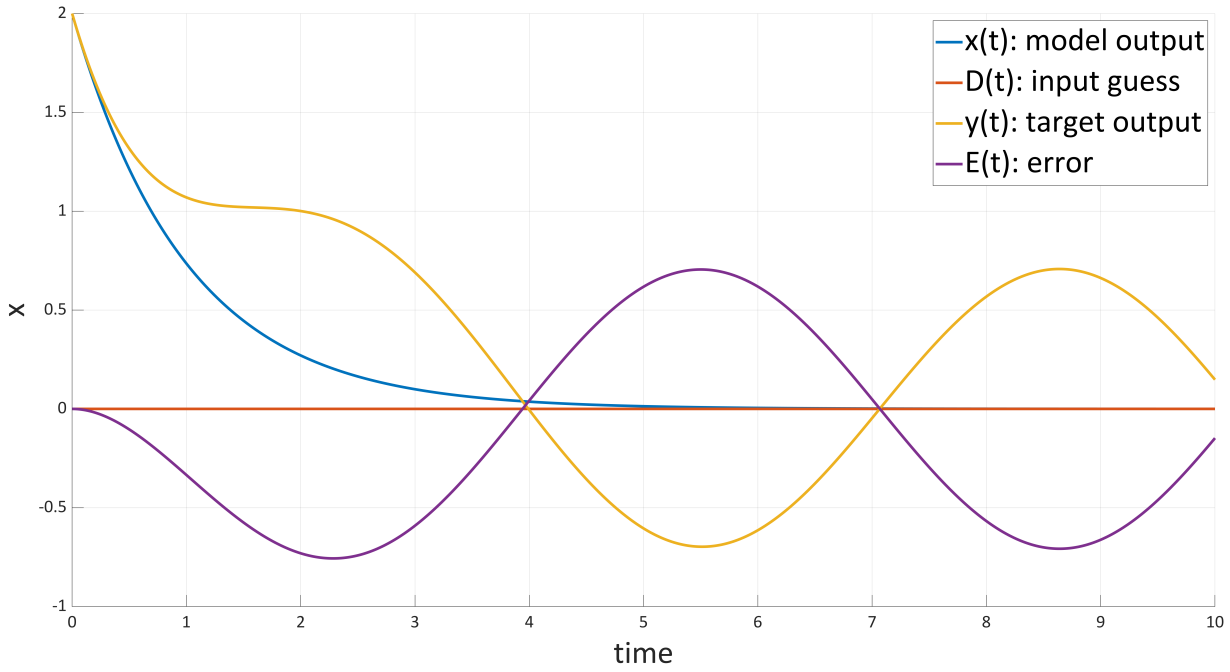

Fig. 6. Initial forward pass of RNN. The error (purple line) is the difference between the network output (blue line) and target output (yellow line). The input guess is also shown (red line).

To determine how much to adjust $D(t)$ at each time step, we need to know how much each value of $D(t)$ contributes to the total loss. To determine this, we need to know the gradient (partial derivative) of the total loss with respect to the input values, i.e. $\frac{\partial L}{\partial D(t)}$, which tells us the ratio of change in loss to change in input. If there were no recurrent connection, we could define this quantity as below, using the chain rule:

$$\frac{\partial \mathrm{L}}{\partial \mathrm{D(t)}} = \sum_{\mathrm{t=1}}^{\mathrm{N}} \frac{\partial \mathrm{L(t)}}{\partial \mathrm{D(t)}} = \sum_{\mathrm{t=1}}^{\mathrm{N}} \frac{\partial \mathrm{L(t)}}{\partial \mathrm{x(t)}} \frac{\partial \mathrm{x(t)}}{\partial \mathrm{D(t)}} \qquad \text{(Eq. 5)}$$

And since we know that $\frac{\partial x(t)}{\partial D(t)} = \delta t$, the gradient would be:

$$\frac{\partial \mathrm{L}}{\partial \mathrm{D}(t)} = \sum_{t=1}^{N} \frac{\partial \mathrm{L}(t)}{\partial \mathrm{D}(t)} = \sum_{t=1}^{N} \frac{1}{2} [x(t) - y(t)] \delta \mathrm{t} \qquad \text{(Eq. 6)}$$

But in fact, $L(t)$ depends not only on the current output $x(t)$ but also indirectly on the previous output $x(t-1)$, because the current output depends on the previous output. Alternatively, this entails that the output $x(t)$ influences not only $L(t)$ but also $L(t+1), L(t+2), \ldots, L(n)$. Thus the gradient of the total loss with respect to the output at any given time is:

$$\frac{\partial L}{\partial x(t)} = \frac{\partial L(t)}{\partial x(t)} + \frac{\partial L(t+1)}{\partial x(t)} + \cdots + \frac{\partial L(N)}{\partial x(t)} = \sum_{\tau=t}^{N} \frac{\partial L(\tau)}{\partial x(t)} \qquad \text{(Eq. 7)}$$

(Eq. 7) reflects the fact that the value of $x(t)$ affects the loss at time t and all subsequent times. Note that the gradient of $x(t+1)$ with respect to the previous value for our system is:

$$\frac{\partial x(t+1)}{\partial x(t)} = -\mathrm{x}(t)\delta t \qquad \text{(Eq. 8)}$$

For all but the first term in the sum in (Eq. 7), we can expand with the chain rule to include intermediate partial derivatives, e.g.

$$\frac{\partial L(t+1)}{\partial x(t)} = \frac{\partial L(t+1)}{\partial x(t+1)}\frac{\partial x(t+1)}{\partial x(t)} = E(t+1)\cdot -x(t)\delta t \qquad \text{(Eq. 9)}$$

$$\frac{\partial L(t+2)}{\partial x(t)} = \frac{\partial L(t+2)}{\partial x(t+2)}\frac{\partial x(t+2)}{\partial x(t+1)}\frac{\partial x(t+1)}{\partial x(t)} = E(t+2)\cdot -x(t+1)\delta t\cdot -x(t)\delta t$$

…

This means that the gradient of the total loss with respect to $x(t)$ is:

$$\frac{\partial L}{\partial x(t)} = E(t) + \sum_{\tau=t+1}^{T} E(\tau)\cdot\left[\prod_{n=t}^{\tau-1} -x(n)\delta t\right] \qquad \text{(Eq. 10)}$$

Because δt is small, the contribution of $x(t)$ to the loss at times $t+1, t+2, \ldots$ becomes smaller and smaller. The error associated with $x(t)$ can be backpropagated to each of the inputs $D(t)$ via:

$$\frac{\partial \mathrm{x(t)}}{\partial \mathrm{D(t)}} = \delta \mathrm{t} \qquad \text{(Eq. 11)}$$

To implement the calculation of the partial derivatives in (Eq. 10) and (Eq. 11), we use the backpropagation algorithm. We start from the final time step and use the partial derivatives to assign error to each node of the unfolded network, allowing the error to "flow" backwards through the unfolded network. The algorithm for this is shown in the Matlab code below. In each time step, we store the back-propagated error in variables dx and dD. Note that we do not pass error back to the initial input, because the initial position is fixed.

```
for i=length(x):-1:2
    % add current error to total error
    % associated with current position:
    dx(i)=dx(i)+E(i);

    % error to current input (dL_dx(t) * dx(t)_dD(t)):
    dD(i)=dx(i)*dt;

    % pass error back to previous timestep:
    dx(i-1)=dx(i)*-x(i)*dt;
end
```

Notice also that by passing the error associated with $x(t)$ backwards in time to $x(t-1)$, using the factor $\frac{\partial x(t+1)}{\partial x(t)} = -x(t)\delta t$, we are implementing the sum over products of (Eq. 10), but doing so in reverse temporal order. After stepping backward through time we have a vector of gradients $\frac{\partial L}{\partial D}$. We use these gradients to adjust the input, with the update rule in (Eq. 12), where $\lambda > 0$ controls how much the gradient is adjusted.

$$D = D - \lambda \frac{\partial L}{\partial D} \qquad \text{(Eq. 12)}$$

We then iterate the steps above many times, i.e. (i) run a forward pass with the updated $D(t)$, (ii) calculate the loss, (iii) backpropagate the error, and (iv) update $D(t)$. The iterations are stopped when the loss stops changing substantially. Fig. 7 shows the results of the iterations. The top panel shows the gradient of the loss with respect to the input after the first and last iterations: initially the gradient is relatively large but by the end of the optimization it is close to zero. The middle panel shows the final model output and target output—they are nearly identical. The bottom panel shows the optimized input, which is very close to the actual input $sin(t)$ that generated the target output.

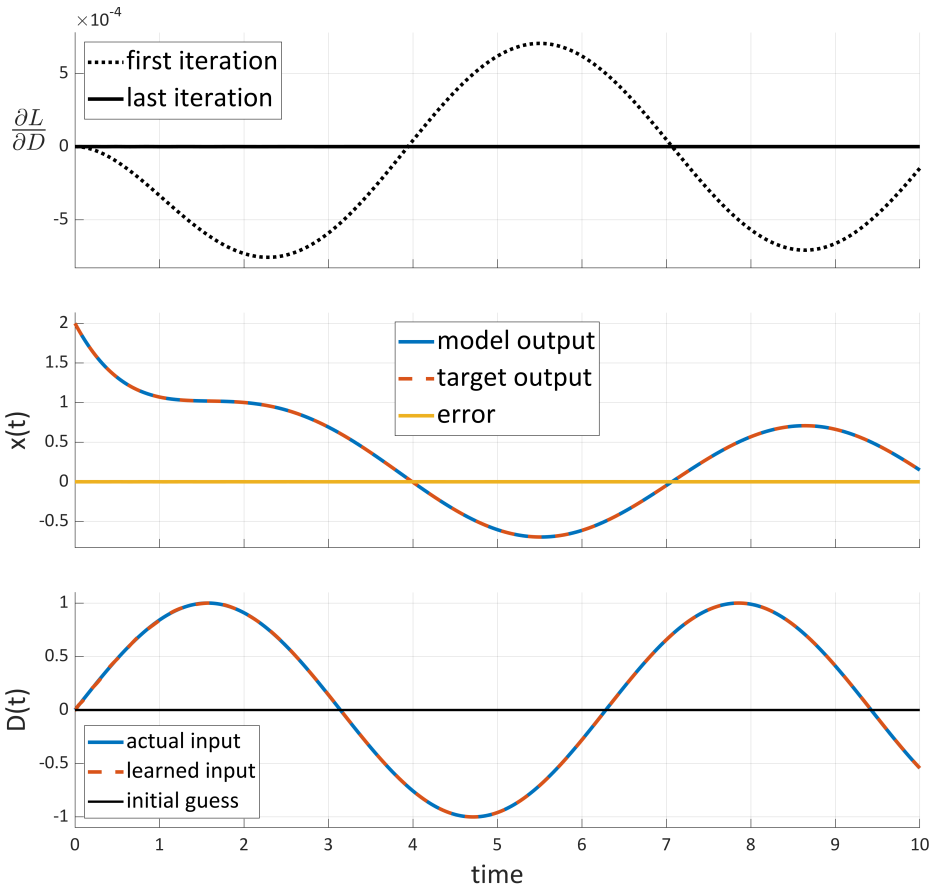

Fig. 7. Results of iterated backpropagation. Top: comparison of gradients after first and last iterations. Middle: the error (yellow) is the difference between model output (blue) and target (orange). Bottom: The initial and final inputs, compared to the actual input, sin(*t*).

The toy example above illustrates several key aspects of the method that extend to the model we implement below. First, we solve a differential equation (or system of differential equations) with a recurrent neural network which implements Euler's method, using small time steps. Second, we reconceptualize the "input" to the system (gestural activation functions) as high-dimensional parameters which can be optimized. Note that in the TD RNN models below, there are additional non-time-dependent parameters—gestural stiffnesses and targets—which are "shared" across the copies of the unfolded network. Although we did not incorporate any shared parameters in the above example, the gradients of the loss with respect to these parameters are determined in a similar fashion to those of the time-varying input, except that in each time step the backpropagated error is accumulated to the static parameters rather than being assigned to time-index of the parameter. Third, the dependence of the system state on its past state is implemented via a recurrent connection in the network. With more complicated system states and update rules, such as a damped, driven harmonic oscillator with both a position and velocity, there will be multiple recurrent connections representing the dependencies of the system state variables on their past values.

Differences between current model and Task Dynamic model

The Task Dynamic model of Saltzman & Munhall (1989) describes both the current states of gesturally-driven tract variable systems, such as lip aperture (LA), and the current states of *model articulators*, such

as the jaw (JAW), lower lip (LL), and upper lip (UL), all of which contribute to the tract variable LA. Each tract variable is modeled with a second order differential equation (Eq. 13) which is analogous to a driven, damped mass-spring system. The reader should be clear that the damped mass-spring system is *not* a model of a *gesture*; rather, it is a model of how the state of a TV system evolves continuously in time. The *x* in the equation is a tract variable, and so in the full TD model there is a set of such equations, one for each tract variable.

$$m\ddot{x} + \beta(t)\dot{x} + k(t)\big(x - T(t)\big) = 0 \qquad \text{(Eq. 13)}$$

$$m\ddot{x} = \underbrace{-\beta(t)\dot{x}}_{\text{damping}} - \underbrace{k(t)x}_{\text{restoring}} + \underbrace{k(t)T(t)}_{\text{driving}}$$

To better understand the equation, consider that the term $m\ddot{x}$ is the product of a mass *m* and an *acceleration* $\ddot{x}$. Then recall the Newtonian relation force equals mass times acceleration, $F = ma$. The equation can be rearranged to describe three forces on a tract variable: a damping force, a restoring/elastic force, and a driving force; the latter of these two forces share the same proportionality factor, $k(t)$, and hence can be combined into a single term. Here we separate them to clarify their origins and meaning. The driving force $k(t)T(t)$ is determined from the stiffness and target parameters of gestures which influence a tract variable. We refer to $k(t)$ and $T(t)$ as the *dynamic stiffness* and *dynamic target* of a tract variable. One sensible way to determine these quantities is to take the weighted average of the static gestural parameters, where the weighting term is gestural activation. If gestural activation functions are assumed to be step functions, then all active gestures have equally weighted contributions to the dynamic parameters, and all inactive gestures have no contribution. Note that critical damping is imposed by setting $\beta(t) = 2\sqrt{k(t)}$, and that the inertial coefficient is set to $m = 1$ and hence can be omitted.

When conducting numerical simulations with the second order ODE in (Eq. 13), it is convenient to convert the second order equation into two coupled first order differential equations (Eq. 14), accomplished by defining the variable $y = \dot{x}$. Note that if x is interpreted as a position, then y is a velocity.

$$\dot{x} = y \qquad \text{(Eq. 15)}$$

$$\dot{y} = -2\sqrt{k(t)}y - \mathrm{k(t)x} + \mathrm{k(t)T}(t)$$

In the SM89 model, changes in tract variable positions and velocities are mapped to changes in model articulator positions and velocities via weighted matrices of partial derivatives. To obtain realistic articulator trajectories, the weighting of "receptivities" of changes in articulator states to changes in tract variable states must be based on the set of currently active gestures (1989: 379). Hence the TD model includes parameters which describe the relative influences of gestures as well as parameters which describe the interactions between tract variables.

Earlier we stated that the TD model does not require temporally delimited gestures. This statement is only specifically true in relation to the role that gestural activation functions play in exerting forces on the states of tract variable systems: the TD model of tract variable dynamics does not require any temporal delimitation of the gestural activation functions for this purpose, because when their values are zero, they exert no forces. However, there is a way in which temporal delimitation is incorporated into the SM89 model. Consider a set of gestures G which influence a given model articulator. When the sum of the current activation values of the gestures in G is zero, there is no influence on that model articulator.

Instead, the model articulator is governed by a neutral attractor. Indeed, the neutral attractor might be conceptualized as just another gesture, one whose activation is determined by the sum of activations of G with the rule in (Eq. 16).

$$g_{neutr} = \begin{matrix} 1, & \sum_j G_j = 0 \\ 0, & \sum_j G_j > 0 \end{matrix} \quad \text{(Eq. 16)}$$

Note that the SM89 formulation determines an influence of the neutral attractor on an articulator, rather than a tract variable. Nonetheless, in that formulation there is a threshold of zero for the total activation of gestures which influence an articulator, and the threshold determines when the neutral attractor exerts an influence. In other words, the influences of active gestures and of the neutral attractor are mutually exclusive. The threshold is implicitly a consequence of the notion that gestures "turn on" and "turn off", with the off-state being zero activation.

In contrast, in the TD RNN implementation, the models developed below only generate tract variable states; they do not represent model articulators and nor do they model interactions between tract variables. Thus the models are not nearly as powerful as the full TD model. In later discussion we consider how the current approach might be extended to be more comprehensive.

Despite omitting a model articulator level, the network models below do have a gestural weighting mechanism and a neutral attractor, although these are conceptualized and implemented in way that differs substantially from the TD model. Specifically, the following differences obtain in the TD RNN model:

(i) Neutral attractors specify a default (equilibrium) state for a *tract variable*, rather than a model articulator.

(ii) Neutral attractors, one for each TV system, have a constant, low level of activation.

(iii) Dynamic parameters of TV systems are weighted averages of gestural and neutral attractor parameters—gestures and neutral attractors are treated identically, with the only difference being that neutral attractor activations are fixed, rather than optimized.

The above aspects of the current approach eliminate entirely the need for temporal delimitation of gestures, either directly or indirectly via an activation threshold.

The basic model

The first model we present adheres more closely to the TD model with regard to how the dynamic stiffness and target are implemented. Fig. 8 shows the network architecture. The blue boxes are input vectors of time-varying gestural activation, with one dimension for each gesture. The yellow boxes are vectors of static gestural stiffness and target parameters. Both the gestural activation and target/stiffness parameters are optimized via backpropagation of error. The nodes labeled (1) and (2) are vectors of dynamic stiffness and target parameters, one for each tract variable. The nodes labeled (3) and (4) are vectors for the time varying velocities and positions of the tract variables. (5) shows the loss at time step *t*, which is half the squared difference between the current TV position and the target value.

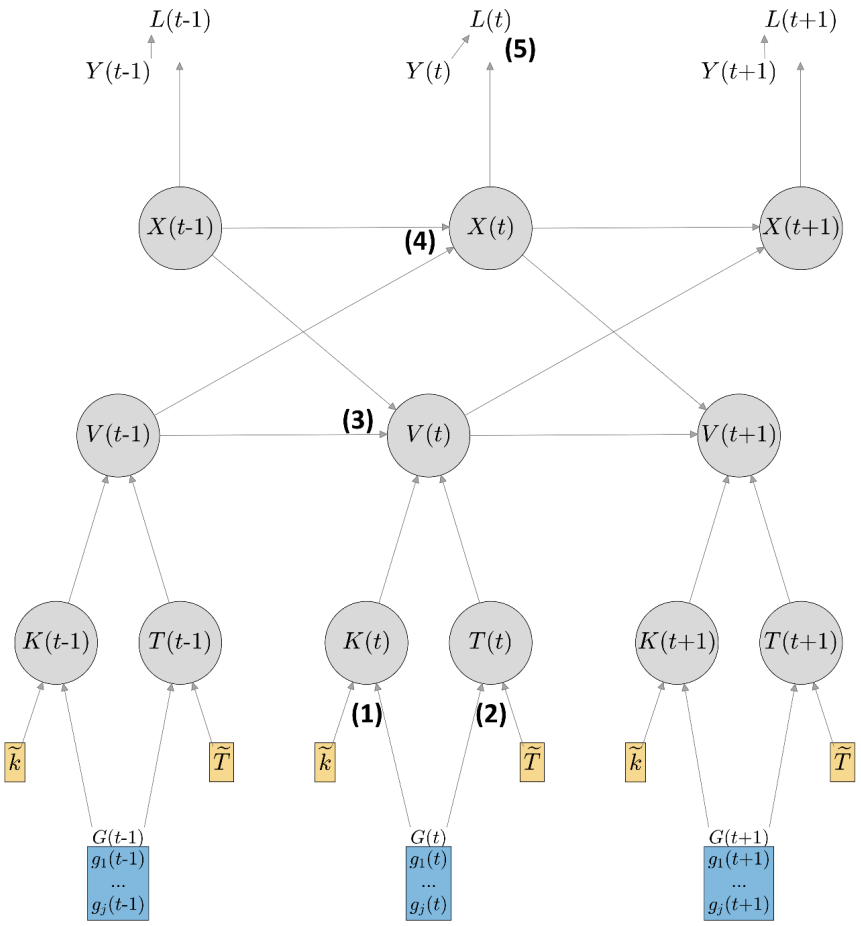

Fig. 8. Basic model network architecture. The model elements are gestural activation (blue boxes), gestural stiffness and target parameters (yellow boxes), dynamic stiffness and target parameters (labels 1 and 2), tract variable velocities and positions (3 and 4), and loss (5).

The equations for the forward dynamics of the model are shown below. For tract variable (TV) i, the dynamic stiffness $K_i(t)$ and target $T_i(t)$ are the gestural activation-weighted averages of the gestural stiffness and target parameters, $\tilde{k}_j$ and $\tilde{T}_j$ (Eq. 17 and 18). W_{ij} is a binary matrix which specifies which gestures *j* are associated with which TVs i; we refer to this as the *gesture-TV map*. To interpret this equation, consider that the gestural stiffness and target parameters are *intrinsic,* long-timescale memories. They are associated with the way that a gestural system interacts with a TV system. The activations of gestures, $G_j(t)$, determine their relative influences on the dynamic stiffness and target of the tract variable.

$$K_i(t) = \frac{\sum_j W_{ij}\, G_j(t)\tilde{k}_j}{\sum_j W_{ij} G_j(t)} \qquad \text{(Eq. 17)}$$

$$T_i(t) = \frac{\sum_j W_{ij}\, G_j(t)\tilde{T}_j}{\sum_j W_{ij} G_j(t)} \qquad \text{(Eq. 18)}$$

The current velocity of each tract variable, $V_i(t)$, is determined by an Euler update of the TD ODE for velocity, as shown in (Eq. 19). The current position of each tract variable, $X_i(t)$, is the position defined by an Euler update of the TD ODE for position, as shown in (Eq. 20). The loss $L_i(t)$ at the current timestep for TV i is the antiderivative of the error (Eq. 21).

$$V_i(t) = V_i(t-1) + \Delta t\left[-2\sqrt{K_i(t)}V_i(t-1) - K_i(t)\big(X_i(t-1) - T_i(t)\big)\right] \quad \text{(Eq. 19)}$$

$$X_i(t) = X_i(t-1) + \Delta t V_i(t-1) \quad \text{(Eq. 20)}$$

$$L_i(t) = (1/2)[X_i(t) - Y_i(t)]^2 \quad \text{(Eq. 21)}$$

For the simple case of one gesture, Fig. 9 illustrates how variation of the parameters and activation function influence the variables of the model. Note that the tract variable $x(t)$ and dynamic target $T(t)$ are expressed in normalized units from [0,1], as is the gestural target; for most tract variables we will assume that 0 is least constricted and 1 is most constricted (for lip aperture this will be reversed). The neutral gesture in all cases has a target of 0.5 and a constant activation of 0.1. Notice that the dynamic target never quite reaches the gestural target because it is a weighted average of the active and neutral gestural targets. Decreasing the gestural target from 0.95 to 0.75 (blue vs. orange lines) decreases the dynamic target when the gesture has non-zero activation. Furthermore, decreasing the maximal gestural activation to 0.5 (purple line) also decreases the dynamic target, but the effect is more subtle. These changes in the dynamic target are manifested not only in the positional extremum achieved by the tract variable, but also its velocity profile and maximal velocity. Another effect to observe involves sigmoidal activation ramping (cf. blue vs. yellow lines). The sigmoidal ramping of gestural activation does not affect the position extremum of the tract variable but causes subtle changes in the position and velocity trajectories.

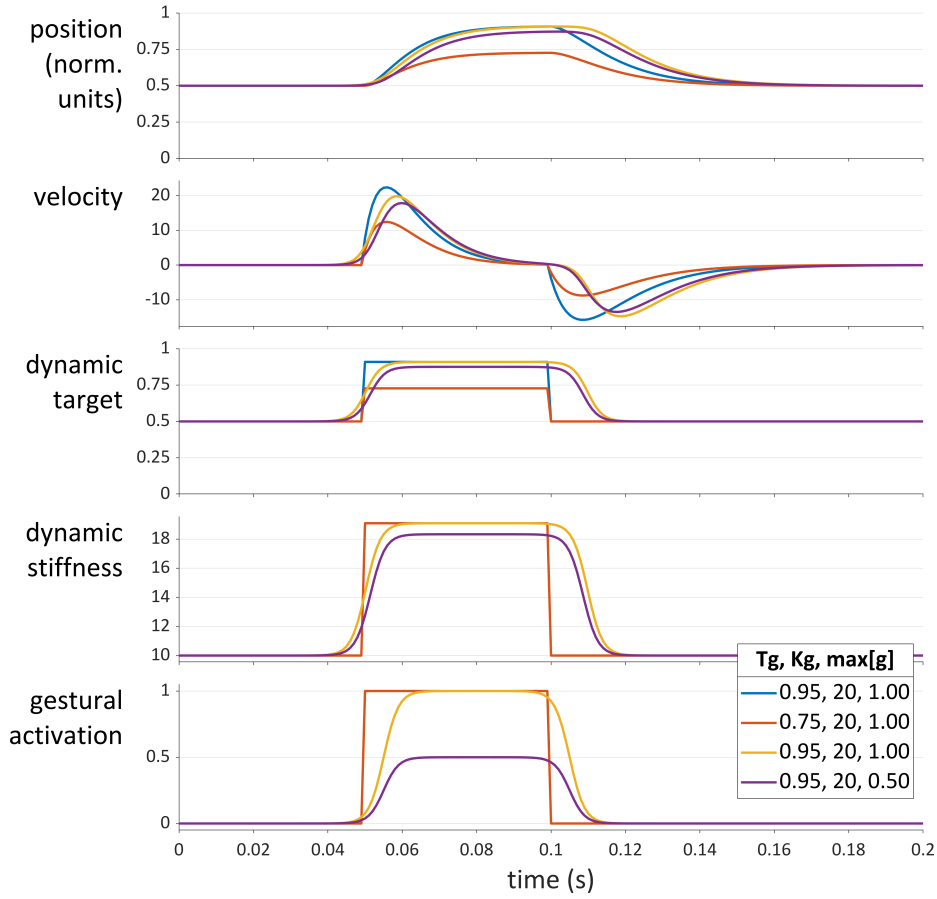

Fig. 9. Illustration of how changes in gestural parameters and activation influence model output.

Ultimately, the output of the model is very similar to what the standard TD model generates for a tract variable influenced by a single gesture. This is not surprising given that we have imposed standard gestural activation functions, and that the RNN architecture mimics the TD equation for tract variable dynamics. Our real interest is in optimizing these functions, given empirical data. The details of backpropagation of error to optimize gestural parameters for the basic model are provided in the Appendix. Results of model optimizations on an articulatory dataset are presented further below.

The field model

The field model differs from the basic model in a number of ways. First, there is a new entity, a *target planning field*, associated with each tract variable. The target planning field (also called an intentional planning field, see Tilsen (2018, 2019a)) is conceptualized as a scalar field, with an activation value defined over a continuous, bounded interval of tract variable values. In actual implementation the field values are only defined on a grid of equally spaced points, which we will index with k. The tract variable value associated with index k is τ_k, and here we use n=50 equally spaced points on the interval [0,1].

The dynamic target of a tract variable $T_i(t)$ is the centroid of field activation, i.e. the activation-weighted average value of τ (Eq. 22). The current stiffness $K_i(t)$ is proportional to the sum of field activation (Eq. 23), with the proportionality constant being $\hat{k}\Delta t^{-1}$.

$$T_i(t) = \frac{\sum_k F_{ik}(t)\tau_k}{\sum_k F_{ik}(t)} \quad \text{(Eq. 22)}$$

$$K_i(t) = \frac{\hat{k}}{\Delta t}\sum_k F_{ik}(t) \quad \text{(Eq. 23)}$$

The field activation has two components, excitatory input $F_{ik}^{+}(t)$ and inhibitory input $F_{ik}^{-}(t)$. The activation is the rectified difference of these (Eq. 24). Both of these inputs are derived from gestural force distributions. In other words, gestures are imagined to exert an excitatory force on an intentional planning field, and an inhibitory force on that same field. The force distributions are Gaussian functions with central values of $\tilde{T}_j^{+}$ and $\tilde{T}_j^{-}$, for the excitatory and inhibitory forces respectively (Eq. 25 and 26). In the current implementation, all force distributions have a fixed standard deviation of 0.10. The amplitudes of the force distributions are determined by the current gestural activation, $G_j(t)$. Note that the matrix W_{ij} is the gesture-TV map, and ensures that only gestures *j* associated with TV i have an influence on the field for TV i.

$$F_{ik}(t) = \mathrm{ReLU}[F_{ik}^{+}(t) - F_{ik}^{-}(t)] \quad \text{(Eq. 24)}$$

$$\mathrm{F_{ik}^{+}(t)} = \sum_{\mathrm{j}} \mathrm{W_{ij}\, G_j(t)}\mathcal{N}\left(\tau_{\mathrm{k}}, \tilde{\mathrm{T}}_{\mathrm{j}}^{+}\right) \quad \text{(Eq. 25)}$$

$$F_{ik}^{-}(t) = \sum_{j} W_{ij}\, G_j(t)\mathcal{N}\left(\tau_k, \tilde{T}_j^{-}\right) \quad \text{(Eq. 26)}$$

The relevant aspects of field model are illustrated in Fig. 10 for a single gesture with TV target 0.95 and sigmoidal activation ramping. The gesture has substantial activation over the time interval [0.050, 0.100]. Heatmaps show the time-evolution of the field and its inputs, with lighter shades indicating higher amplitudes. The values of the field and inputs at time 0.075 s, which is in the middle of the activation interval, are shown in the panels on the right. Note that the centroid of the target planning field (green line) is shown both in the time-slice and superimposed in the field heatmap. The dynamic stiffness is proportional to the area under the activation function in the planning field. Throughout the simulation, the neutral attractor (with target 0.5) has an activation of 0.1.

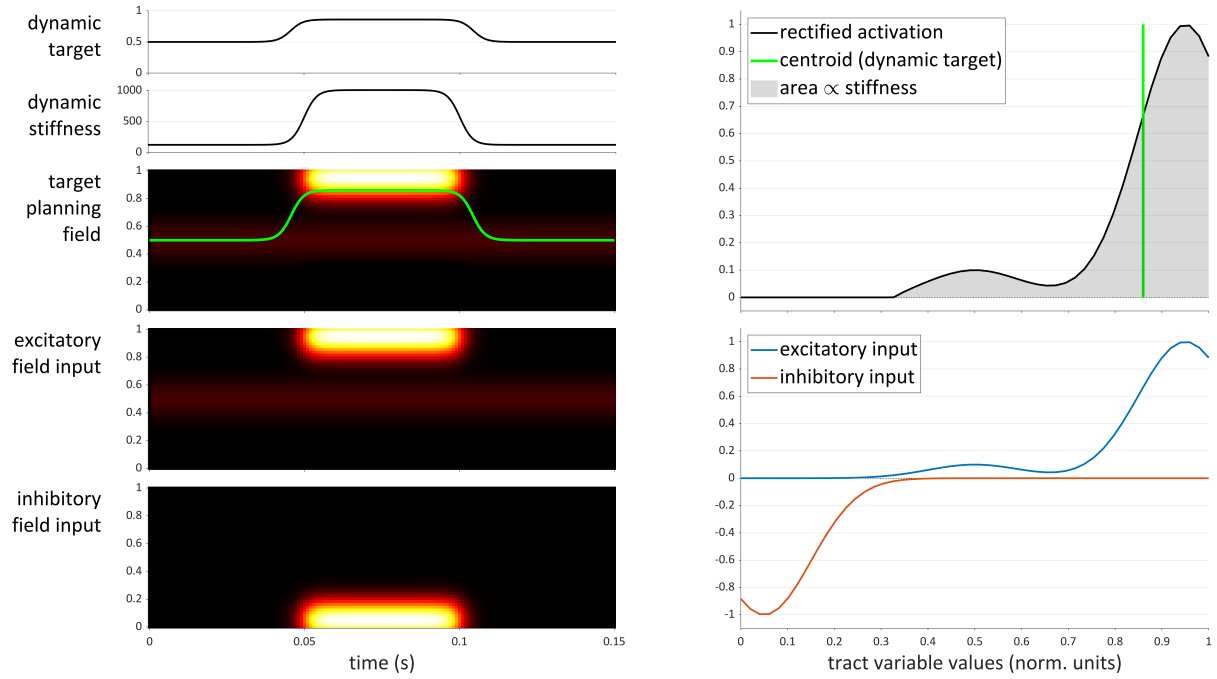

Fig. 10. Example of target planning field dynamics for a single gesture with logistic activation ramping. Gestural activation is substantial in the interval t=[0.05, 0.100]; gestural target is 0.95. Left (top-to-bottom): dynamic target, dynamic stiffness, and heatmaps of field activation, excitatory field input, and inhibitory field input. Right: excitatory and inhibitory input and rectified field activation at time 0.075 s. The dynamic target (centroid) is shown with a green line.

To simplify the model implementation, the dynamics of the planning field are not modeled explicitly—there is no representation of the first time derivative of the current field activation—hence the "force" distributions do not play the role of a true force by governing the change in the first derivative. Instead, a weighted sum of the activation force distributions determines the current value of the intentional planning field. This is conceptually equivalent to thinking of the gestural forces as target activation values, defined over the field, with the field dynamics evolving very rapidly (within each timestep) to an equilibrium; in this case, the equilibrium is the gestural-activation weighted average of the excitatory values minus the gestural-activation weighted average of the inhibitory values.

There are several important points to mention here. First, the rationale for the rectification of the field (i.e. the ReLU nonlinearity) comes from a microscopic conception of an intentional planning field in which the gestural systems are neural ensembles which project to both excitatory and inhibitory neurons in somatotopically organized neural populations (possibly located in primary motor cortex). The inhibitory neurons in the intentional planning field exert local inhibitory forces on excitatory neurons in the field, whereas the excitatory neurons are the source of efferent projections to brainstem systems that control muscle tension. Hence only the excitatory neurons of the primary motor cortex fields directly determine current target positions. The linear rectification implements these assumptions.

Second, the approach to determining the dynamic stiffness in the field model makes gesture-specific stiffness parameters unnecessary. In the field model, the strength of the force experienced by a TV system is proportional to the total activation of the field, with just one parameter $\hat{\mathrm{k}}\Delta t^{-1}$. On the microscale this entails that the spiking rates of excitatory neurons in the field determine the strength of the driving force on a given tract variable. To the extent that we associate different movement velocities with different types of gestures (e.g. consonantal vs. vocalic), the model can generate differences via the relative activation of gestures and the neutral attractor for each intentional planning field. Eliminating gestural

stiffnesses as independent parameters is desirable because it is unclear how to derive them from the microscale conception. It is more intuitive that the strength of the force experienced by a TV system is determined by the spiking rates of neurons that encode targets.

Methods

Dataset. The dataset used here is from an experiment that has previously been analyzed using other methods (Tilsen, 2020). There were six participants in the experiment, and articulatory data were collected with electromagnetic articulography (EMA). More detail regarding data collection procedures can be found in the aforementioned paper. Each trial of the experiment elicited a production of a CVC syllable cued by a visual signal, where C = {/p/,/t/,∅} and V = /a/. Each production was also preceded by a prolonged /i/ vowel (for about 1.5 s), during which time the participant was informed to the identity of the target CVC syllable. Hence there is an opportunity for the pre-response vowel /i/ to be colored by sub-threshold influences of gestures which are components of the target response but have not been "initiated" or "activated" in the conventional sense.

Network output. The target outputs of the network are time series of tract variables. Specifically, three tract variables: lip aperture (LA), which is the Euclidean distance between the lips; tongue tip constriction degree (TT), which is derived from the vertical position of a sensor on the tongue tip, and tongue body constriction degree (TB), which is derived from the vertical position of a sensor on the tongue body. All three of these variables were normalized to the interval [0,1] on a by-participant basis. Notice that these tract variables are the rows of Table 1, which shows the mapping between gestures and tract variables.

Parameter initialization and constraints. Given the pre-response vowel and the nine target syllables of the experiment (orthographically cued with: *pop*, *pot*, *pah*, *top*, *tot*, *tah*, *op*, *ot*, *ah*), a standard analysis holds that there are up to six unique oral articulatory gestures that could be present on a given trial: LA clo, LA op, TT clo, TT op, TB [i], and TB [a], i.e. constriction and release gestures associated with /p/ and /t/, and vocalic gestures associated with /i/ and /a/. The same gesture may occur twice in a trial, as in *pop* and *tot*. Assuming that a gesture which occurs in both onset and coda is controlled by "the same system", there are six gestural systems in the network. The initial target parameters of these gestural systems are shown in the table below. Neutral attractor targets were fixed at 0.5 for all tract variables. Note that there is a tract variable coordinate direction difference between LA and TT/TB, such that maximal bilabial closure of /p/ is defined as an LA (lip aperture) value of 0, whereas maximal alveolar closure of /t/ is defined as 1 (and similarly for TB)—in other words, the TV coordinate scale of LA is reversed compared to the scales of TT or TB. This potentially confusing inconsistency is a consequence of preserving the meaning of "aperture" as a degree of opening (with 1 being maximal opening), in contrast with a constriction degree, which is a degree of closure (with 1 being maximal closure).

Table 1. Initial values of target parameters of gestural systems

		gestural system						
		LA		TT		TB		NEUT
		clo	op	clo	op	[i]	[a]	
tract variable system	LA	0	1					0.5
	TT			1	0			0.5
	TB					1	0	0.5

The initial stiffnesses of all gestural systems were 1 x 10^5 s^{-2}. The units of stiffness of a physical spring are force/length, where length is a displacement from equilibrium; hence stiffness relates displacement to force. The SI base units of force are $kg \cdot m \cdot s^{-2}$, and for the TD model we ignore mass and analogize the normalized tract variable coordinate to length. The frequency of an undamped harmonic oscillator with mass of 1 is $\omega = \sqrt{k}$, with period $T = \frac{2\pi}{\omega}$, and so the initial stiffness of 10^5 would correspond to an oscillation period of about 0.020 s, if the system were not damped.

All of the neutral attractor parameters in the model are fixed, with activations of 0.1 (10% of maximum), targets of 0.5, and stiffnesses of 0.5 x 10^5. Because the neutral attractors are always active in the current model, their fixed activations and stiffnesses influence the optimized values of the gestural system stiffnesses: gestural system stiffnesses must remain sufficiently large to outweigh the influence of neutral attractor stiffnesses; the relative activations of gestures to neutral attractor activation also plays a key role in this balance. Note that in analyses below we graph the relative stiffness, k/k_N, where $k_N = 0.5 \times 10^5$ is the fixed stiffness of the neutral attractors.

For each optimization, the initial positions and velocities for each tract variable are set to the empirical values for that tract variable. Due to the network architecture (see Fig. 8), the initial positions generated by the model never change. Furthermore, the backpropagation implementation never adjusts the initial gestural activations, and so the velocities at the first time step also remain fixed.

Two different approaches to initializing gestural activation functions (i.e. network input values) were investigated. Consider that in typical deep learning contexts, network parameter values are often randomly initialized. In our context however, there are more direct relations between the gestural activation functions and the network outputs. Hence it makes sense to consider an initialization strategy which makes use of these relations. One approach, which we refer to as *gradient-based initialization*, did this by setting the initial gestural activation functions to be a transformation of the empirical tract variable time derivatives. Examples are shown with dashed lines in the middle and bottom panels of Fig. 11. Specifically, for each gesture, the derivative of the corresponding tract variable was calculated. If the gestural target was 0 (as for the |LA clo| gesture in Fig. 11), the sign of the derivative was reversed. The reason for this reversal is that gestures which decrease a tract variable are active when the derivative of the tract variable is negative. Next, the initial activation functions were rectified and then rescaled to a maximum of 1, because gestural activation is limited to the range [0, 1]. A simpler initialization approach, which we refer to as *zero-initialization*, involved setting the values of all gestural activation functions at all times to 0 (blue lines in Fig. 11).

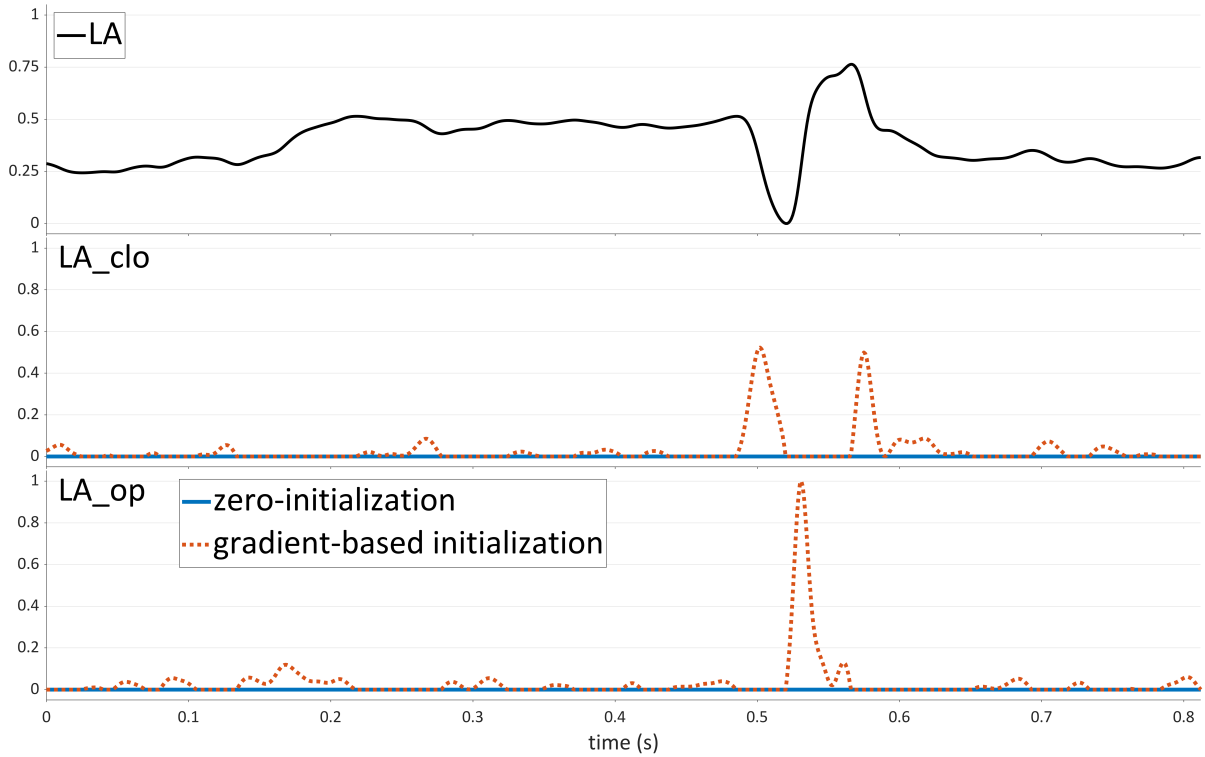

Fig. 11. Comparison of zero- and gradient-based initialization. Top: empirical lip aperture time series. Middle, bottom: initial activation functions for |LA clo| and |LA op| gestures.

For the field model, the standard deviations of the excitatory and inhibitory Gaussian force distributions for all gestures were fixed at 0.1 normalized tract variable units. The proportionality constant for the relation between field activation and dynamic stiffness, $\hat{k}$, was fixed at 100. The initial modes of the excitatory Gaussian force distributions were the same as the gestural system targets in Table 1. The initial modes of the inhibitory force distributions were set to $\hat{T}^- = 1 - \hat{T}^+$, i.e. they were on the opposite ends of the tract variable coordinate space, such that, for example, the mode of the inhibitory force distribution of |LA clo| coincided with the mode of the excitatory force distribution of |LA op|. Neutral attractors exerted no inhibitory forces. Only the gradient-based initialization was used with the field model.

Optimization algorithm. The models were optimized by using a fairly simple approach in which the error gradients were weighted by the learning rate parameters ε_g, ε_k, and ε_T, for gestural activation, stiffness, and target parameters, respectively. For the basic model optimizations, $\varepsilon_g = 0.1$ and $\varepsilon_k = \varepsilon_T = 1.0 \times 10^{-5}$. Different values for these parameters may result in different optimized parameters/activation functions, and indeed, the relative value of ε_g to $\varepsilon_k/\varepsilon_T$ represents a decision about whether to prioritize variation in gestural activation or gestural targets/stiffnesses. The current values prioritize gestural activation. The following stopping criterion was used: if all decreases in the loss for 20 consecutive trials were less than 1.0×10^{-5}, the optimization was terminated.

Analysis methods. The following analyses are conducted below. First, we compare the zero- and gradient-based initialization strategies. To do this, we consider loss distributions, target and stiffness parameter distributions and correlations, and gestural activation functions. Second, we examine qualitative features gestural activation functions of the zero-initialization optimizations. For all analyses other than those

involving loss distributions, trials with a loss greater than the upper 98th percentile of the loss distributions were excluded. These exclusions were made in order to avoid drawing inferences based on potentially abnormal trials.

Analyses

Loss distributions

The zero-initialization and gradient-based initialization strategies produced high-quality fits of the empirical data and had very similar loss distributions. The field model (with gradient-based initialization) exhibited substantially larger loss, indicative of poor fits of empirical data. For each of the models/initialization strategies, the loss distributions over all trials are shown in Fig. 12. Note that loss is measured in units of ½ squared error. As we discuss later, the poor fits of the field model may be due to the absence of free parameters for adjusting stiffness.

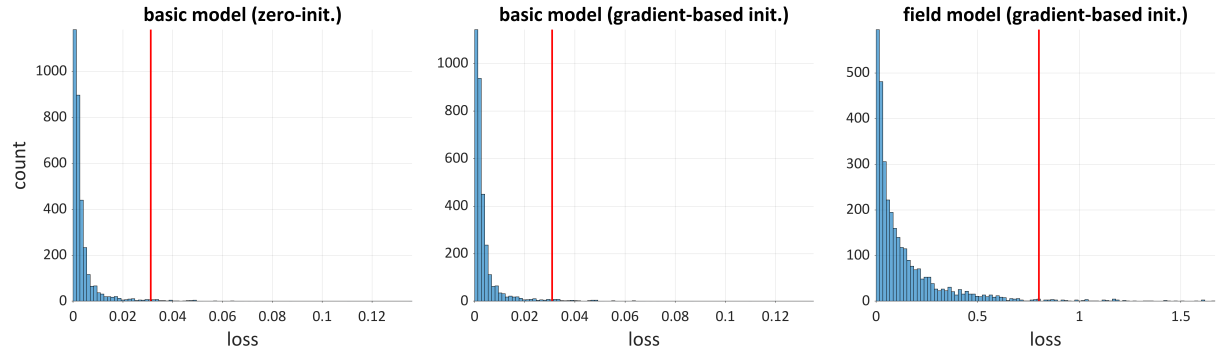

Fig. 12. Loss distributions for the basic model and field model. Red lines show 98% threshold used for data exclusion in subsequent analyses.

The correlation of individual-trial losses between the two initialization strategies for the basic model was very high (r=0.9987), as shown in Fig. 13. This suggests that the overall quality of the fits were similar. It also indicates that when the basic models did have low-quality fits (which was not often), it was due to idiosyncrasies of individual trials, rather than inappropriate initialization.

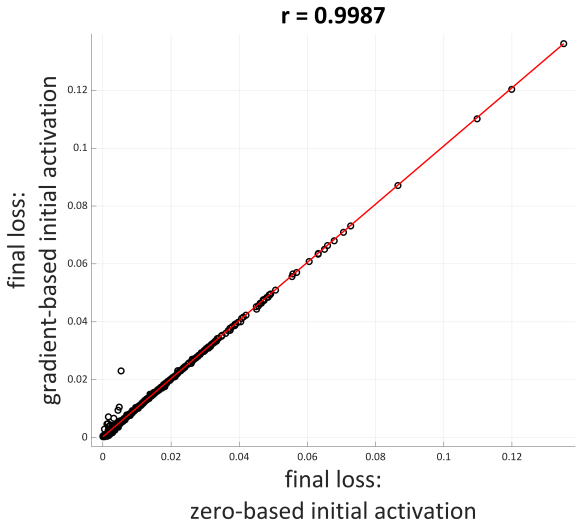

Fig. 13. Scatterplot of by-trial losses of zero- and gradient-based-initialization strategies. Red line is the function x = y. The proximity of points to this line indicates a high degree of correlation.

Target and stiffness parameters

The similarity of the loss distributions and their high correlations might lead one to infer that the optimized parameters of the models from both initialization schemes were similar, but this was far from the case. Instead, the gradient-based initialization resulted in substantially more variation within and between response categories (i.e. /pap/, /pat/, /pa/, /tat/, etc.). This difference is illustrated in Fig. 14 and Fig. 15, which show boxplots of across-participant target and relative stiffness parameter distributions, from gradient-based and zero-initialization, respectively. The response categories (horizontal axes) are sorted by their mean relative stiffness for each gesture. The figures show that targets and stiffness remained fairly close to their initial values, especially for zero-initialization. This may not be surprising, given the difference in learning rates that was imposed on the parameters.

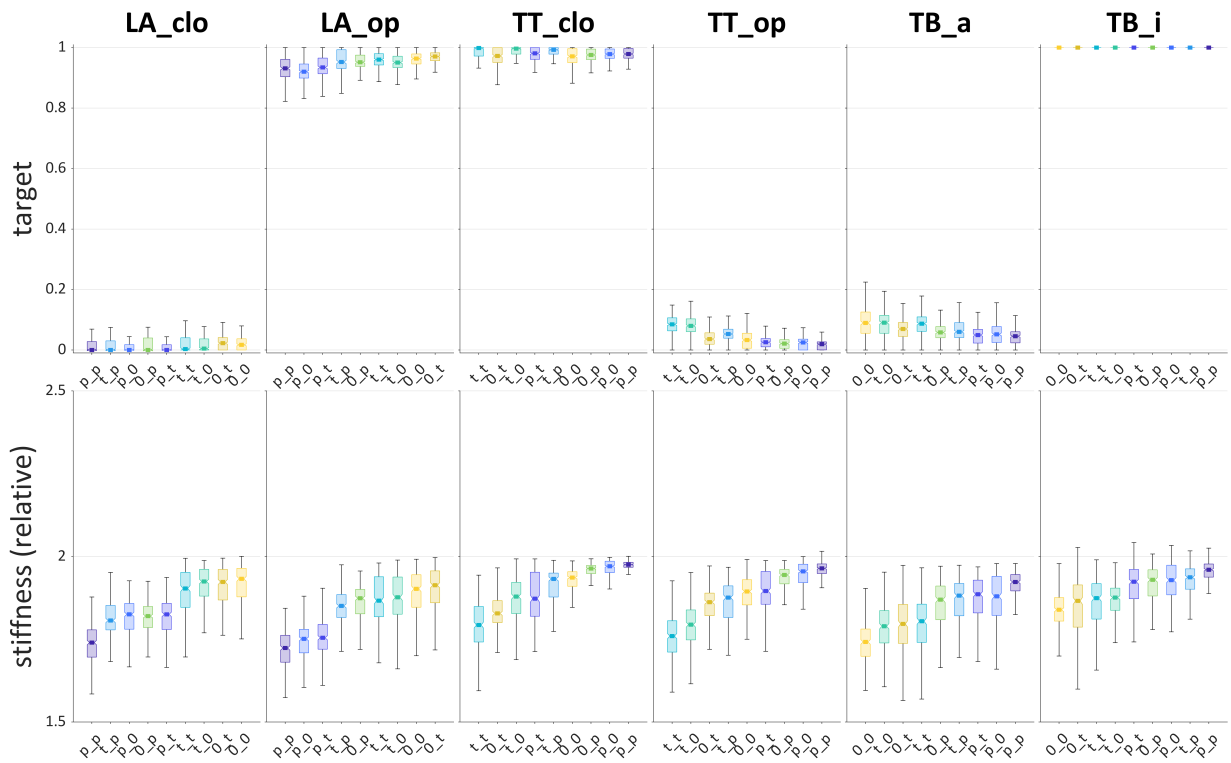

Fig. 14. Boxplots of across-participant target and relative stiffness parameter distributions from gradient-based initialization. Response categories are sorted by their mean relative stiffness for each gesture.

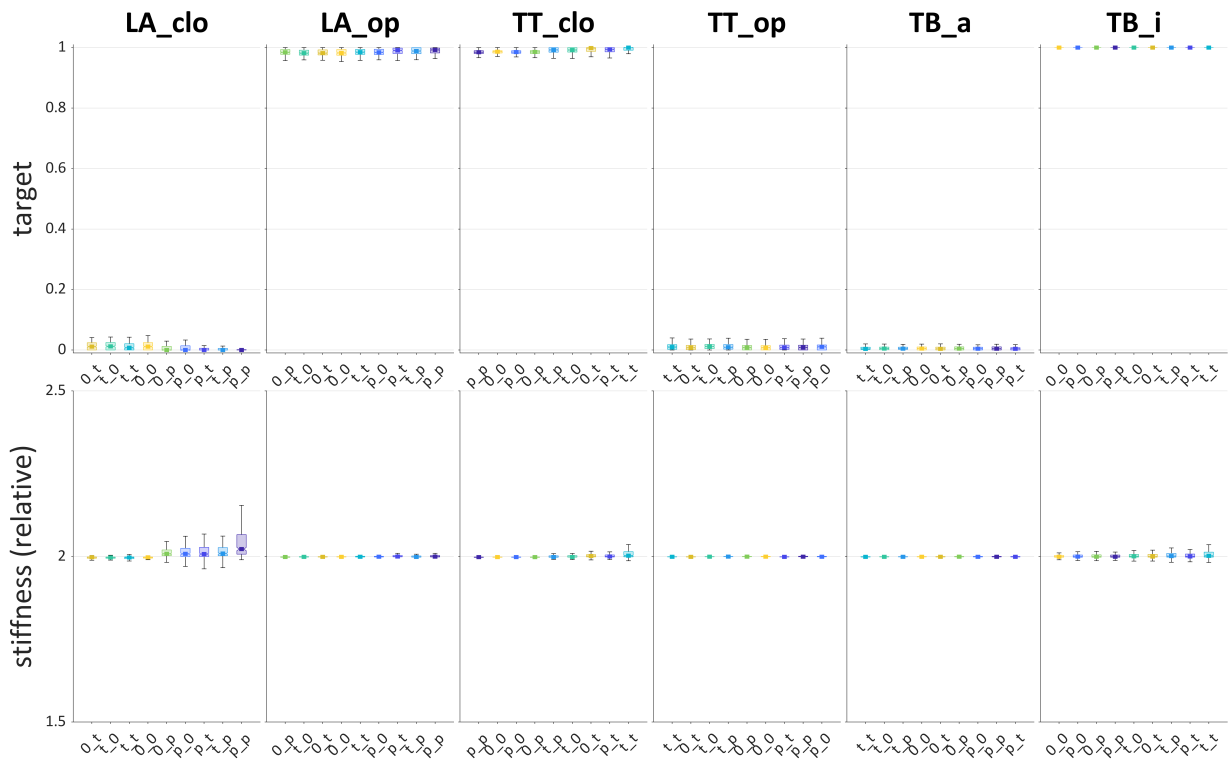

Fig. 15. Boxplots of across-participant target and relative stiffness parameter distributions from zero-initialization. Scales are identical to those of Fig. 14. Response categories are sorted by their mean relative stiffness for each gesture.

The same parameter distributions as above are shown in Fig. 16 and Fig. 17 below, after the parameters have been recentered for each participant/gesture. These allow for a clearer visual impression of patterns in between response-category variation. Response categories are sorted by their mean relative stiffness for each gesture. Particularly for the gradient-based initialization, it is evident from the sorting of the response categories that there is variation conditioned by the presence/absence of expected active gestures in the optimized values of LA and TT gestural target and stiffness parameters. For example, Fig. 16 shows that the stiffness parameters obtained with gradient-based initialization for |LA clo| were lower when the response included a /p/ (i.e. an active |LA clo| gesture), and lowest when two active |LA clo| gestures are expected, as in /pap/. In contrast, with zero-initialization, Fig. 17 shows that |LA clo| stiffness was higher when the response included a /p/.

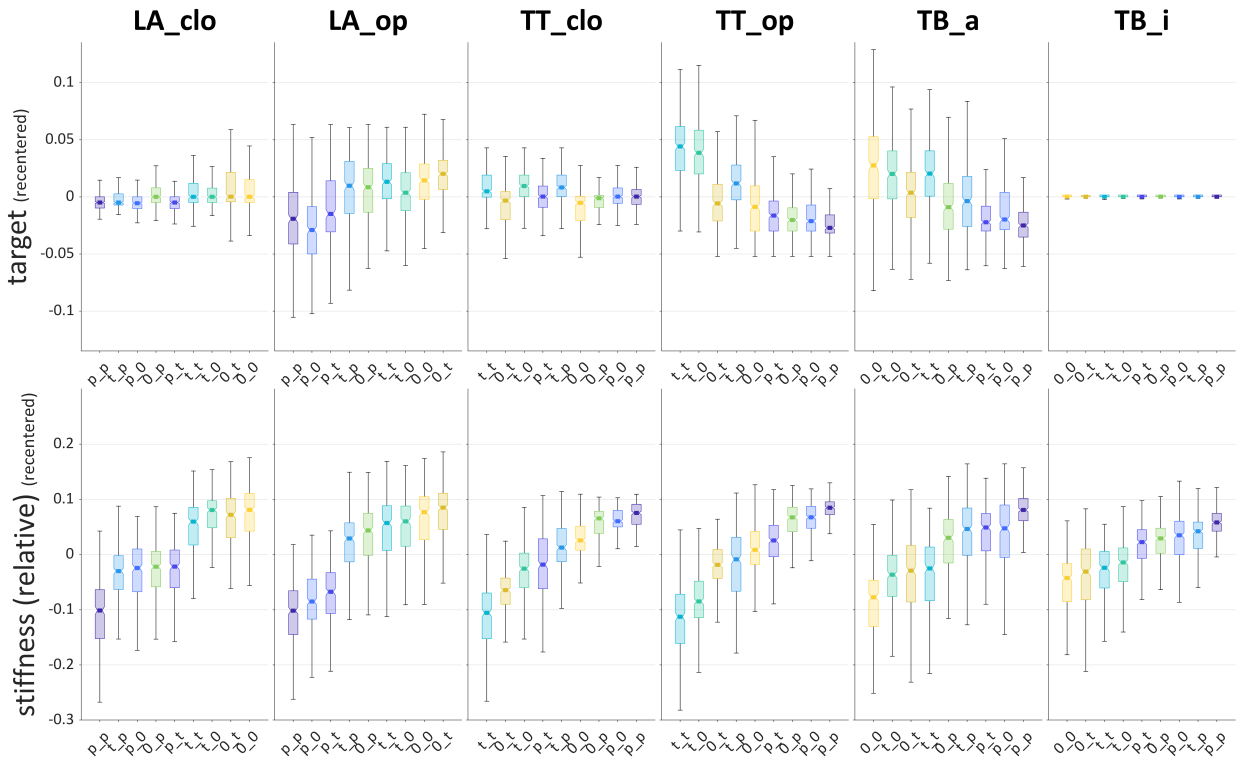

Fig. 16. Boxplots of target and relative stiffness parameter distributions from gradient-based initialization, after recentering by participant and gesture. Response categories are sorted by their mean relative stiffness for each gesture.

The aforementioned relation between stiffness and response category for the gradient-based initialization is somewhat unexpected. Specfically, it is curious that when the segmental identity of a response leads us to assume that an active gesture is present (e.g. an active LA clo gesture should be present when there is a /p/ onset or coda), the stiffnesses of corresponding gestures are *lower*. The relation for the zero-initialization scheme is more sensible: presence of an active gesture is associated with higher stiffness of that gesture, i.e. a stronger driving force from the gesture.

Another noteworthy pattern of the gradient-based activation is that the least stiff vocalic gestures were those in the onsetless and codaless environment. This pattern is absent from the zero-initialization optimizations, where effectively no variation was observed in stiffness parameters of gestures other than |LA clo|.

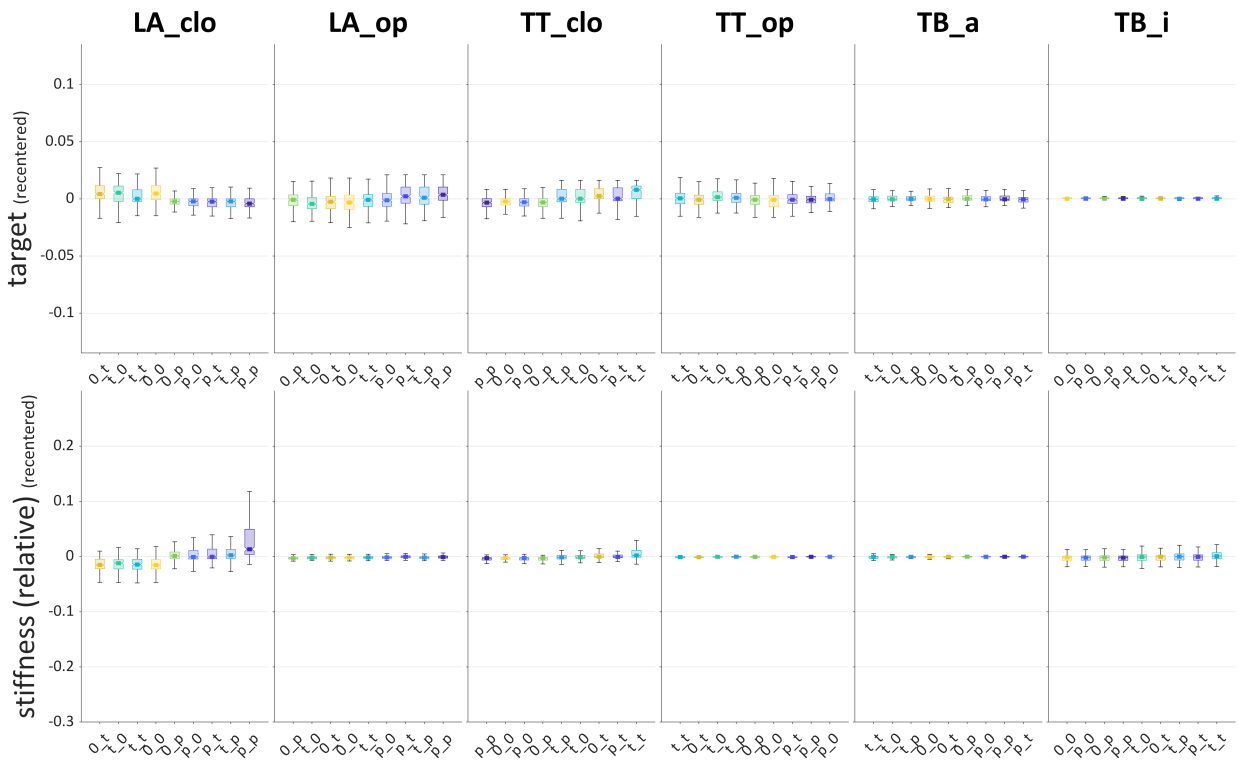

Fig. 17. Boxplots of target and relative stiffness parameter distributions from zero- initialization, after recentering by participant and gesture. Scales are identical to those of Fig. 16. Response categories are sorted by their mean relative stiffness for each gesture.

The target parameters of the basic model optimizations show some notably different patterns between participants/responses. These are illustrated for zero-initialization for |LA clo| and |LA op| in Fig. 18. In some cases, all or almost all of optimized parameter values remained at the extremes of the normalized tract variable scale, which ranges from 0 to 1. This arises when the backpropagation gradients are always pushing the target parameter toward a scale extremum, a phenomenon which we will refer to as *pinning*—i.e. a parameter value gets pushed against a floor or ceiling that is imposed in the gradient update step of each iteration of the optimization. Fig. 18 shows that pinning was prevalent for |LA clo| target parameters for all response categories for P01, and for several of the response categories for P03, P04, and P05. In other cases, the distributions were highly skewed toward the extremal value, suggestive of pinning on some but not all trials. In other cases the distributions were more normally distributed at some value close to the extremum. The same generalizations apply to the |TT clo| and |TT op| gestures (not shown), and interestingly, participant P01 also exhibited the strongest pinning of the |TT clo| gesture. Further investigation is required to determine when pinning occurs, and to what extent it has theoretical or empirical significance. It could be an artefact of how the empirical tract variables are scaled or the bounds imposed on gestural targets.

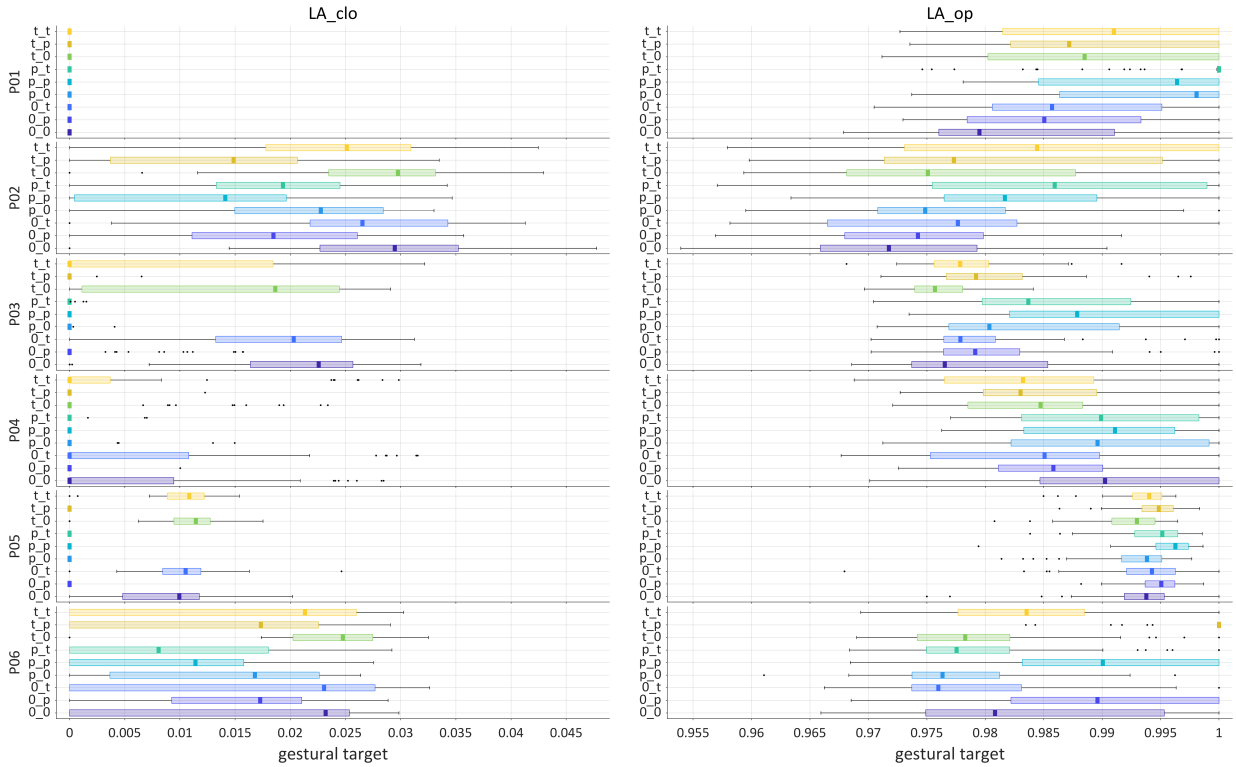

Fig. 18. Boxplots of target parameters for |LA clo| and |LA op| gestures obtained from zero-initialization, for each subject and response. Boxes are 25th-75th percentile ranges, medians are thick lines, outliers are dots.

Overall it was quite rare that the optimized parameters for any of the LA or TT gestures was further than 0.05 from the extremal value, i.e. 5% of the full range, and for each participant (the range was larger, about 0.15, for the gradient-based optimizations). This indicates that the optimized values are capturing some degree of regularity in the target parameters of gestures, which is desirable if we suppose that gestural targets are context-invariant. However, there were a fair number of statistically significant differences within participants between the target parameter distributions for some responses, which could suggest a context-specificity of targets. These differences could on the other hand be attributed to coarticulatory effects which are not captured by the model.

The most strongly pinned (and therefore most consistent) target parameter was the target of the |TB [i]| gesture, which was pinned at 1 for nearly all of the participants/responses (Fig. 19). Interestingly, the |TB [a]| target was the least pinned of all six gestures.

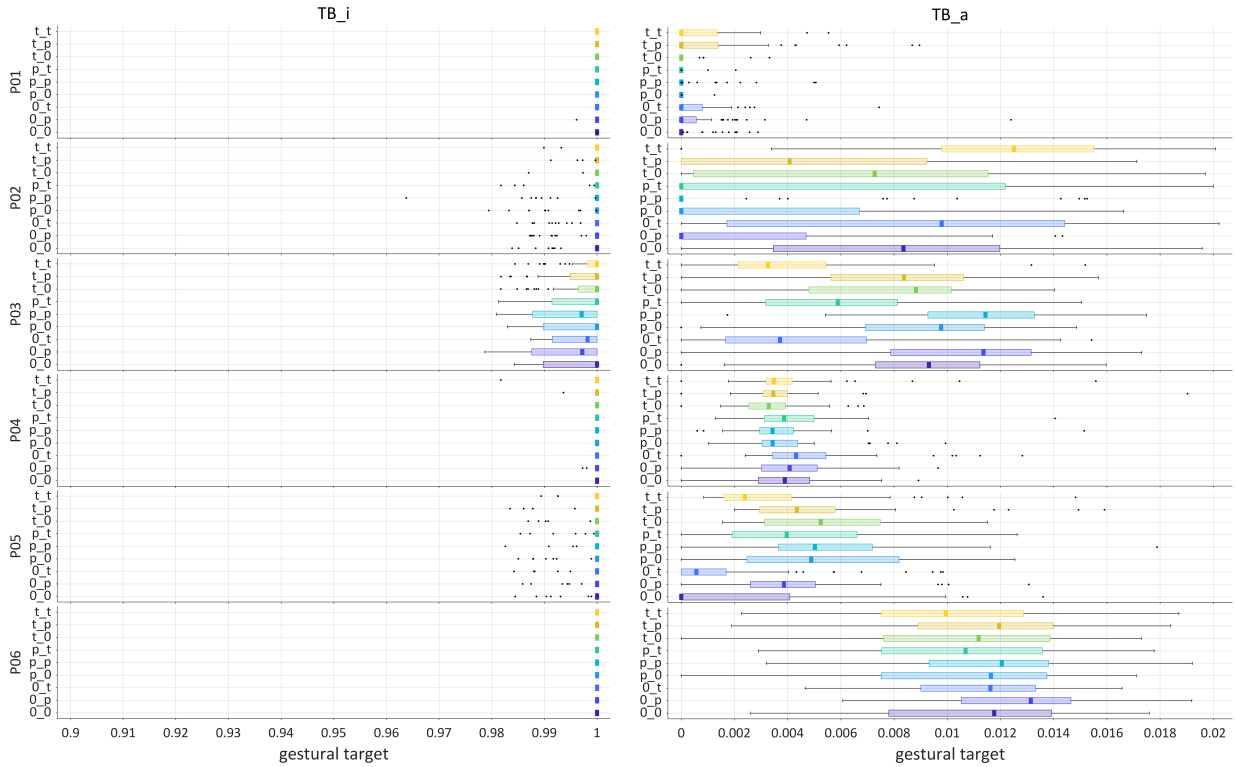

Fig. 19. Boxplots of target parameter distributions for |TB [i]| and |TB [a]| gestures obtained with zero-initialization, for each participant and response. Boxes are 25th-75th percentile ranges, medians are thick lines, outliers are dots.

Stiffness parameters were never pinned by the optimization because they had no ceiling and a floor of 0, which is very low relative to the values necessary produce good results. Fig. 20 shows stiffness parameter distributions for each participant/response for the |LA clo| and |LA op| gestures, obtained with zero-initialization. The distributions for the TT and TB gestures (not shown) are in similar ranges. As shown in Table 2, the mean relative stiffnesses (rKg) for all six gestures with gradient-based initialization were in the range of 1.83-1.90, which means they were a bit less than twice as stiff as the neutral attractor. Observe that the *clo* gestures have higher mean stiffness than the *op* gestures, and TB [i] has higher stiffness than TB [a]. This might suggest that constriction formation gestures are more stiff than constriction release gestures. For zero-initialization, the optimized values were always very close to the initial value, which was twice the stiffness of the neutral attractor. This could be an artefact of the relatively low learning rates for stiffness and high error associated with initial activation functions.

Table 2. Mean stiffnesses for each gesture and 95% distribution intervals

	gradient-based-init.		zero-initialization	
	rKg	rKg_ci	rKg	rKg
LA_clo	1.85	[1.65, 2.05]	2.01	[1.94, 2.08]
LA_op	1.83	[1.61, 2.05]	2.00	[1.98, 2.03]
TT_clo	1.90	[1.72, 2.08]	2.00	[1.97, 2.04]
TT_op	1.88	[1.68, 2.07]	2.00	[1.98, 2.02]
TB_a	1.83	[1.62, 2.04]	2.00	[1.99, 2.01]
TB_i	1.89	[1.72, 2.07]	2.00	[1.97, 2.04]

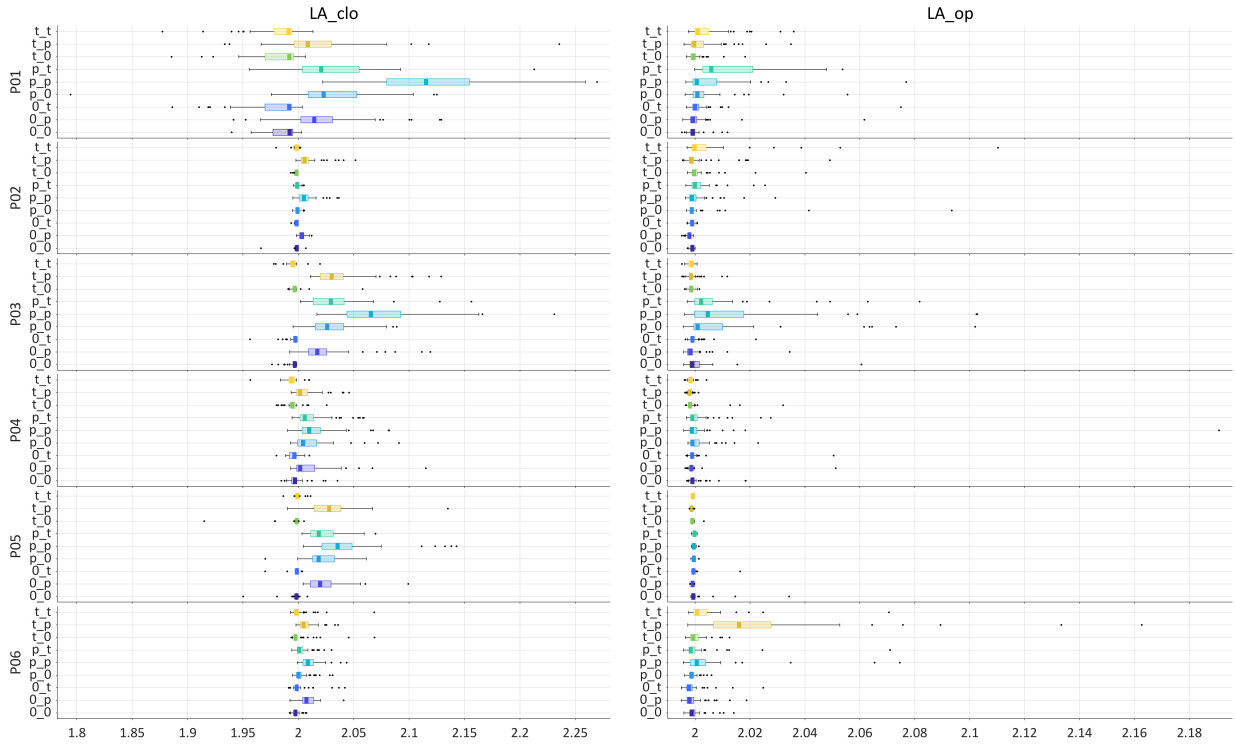

Fig. 20. Boxplots of stiffness parameters for |LA clo| and |LA op| gestures for each subject and response, with zero-initialization. Boxes are 25th-75th percentile ranges, medians are thick lines, outliers are dots.

Stronger correlations between parameters were observed with the gradient-based initialization than with zero-initialization. The correlations are shown in Fig. 21. In particular, the stiffnesses of pairs of gestures which share the same tract variables with opposing targets (i.e. antagonistic gesture pairs such as |LA clo| and |LA op|) are highly correlated ($r \approx 0.85$) in the gradient-based scheme. This means that for trials in which the optimization procedure resulted in lower/higher stiffness for one gesture in a pair, the other also had lower/higher stiffness. These stiffness correlations are likely a consequence of co-activation patterns that we examine below.

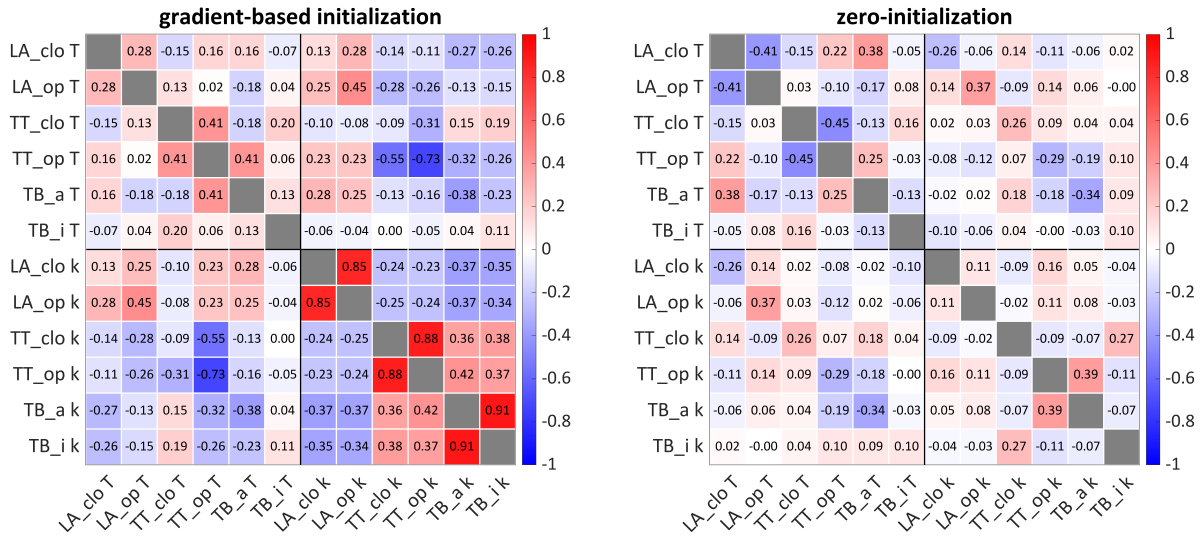

Fig. 21. Parameter correlations for the basic model with gradient-based- and zero-initialization.

Another instance of a strong correlation involves the |TT op| target and stiffness: these are negatively correlated (r = -0.73), entailing that when the TT release gesture had a less constricted target, it was more stiff. The analogous correlation was observed betwee the LA release gesture and its stiffness, but was somewhat more moderate (r = 0.45).

Gestural activation functions: comparison of initialization strategies

There are two main differences between the optimized activation functions obtained from zero-initialization vs. gradient-based initialization: (i) Antagonistic gestural pairs (i.e. LA clo/op, TT clo/op, and TB [i]/[a]) had complementary activation peaks in the zero-initialization scheme; in contrast, these pairs had positively correlated activation peaks in the gradient-based scheme. (ii) Levels of gestural activation associated with activation peaks were higher in the gradient-based scheme than in the zero-initialization scheme.

These differences are illustrated in Fig. 22 and Fig. 23, which show optimized gestural activation functions for an example utterance of /pat/, obtained from zero- and gradient-based initialization respectively. To see the complementary vs. co-activation patterns, examine the |LA clo| and |LA op| gestural activation functions obtained from zero-initialization (Fig. 22). During the |LA clo| peak associated with the onset of the target response (around time 0.475 s), |LA op| activation falls to zero. Subsequently, when the bilabial closure is released with the |LA op| gesture, |LA clo| activation falls to zero. These same patterns can be observed for the TT and TB antagonists. (Note that the activation functions reflect not only active gestural control, but passive mechanical effects, a phenomenon which we discuss later.) In contrast, the |LA clo| and |LA op| gestural activation functions obtained from gradient-based initialization of the same utterance (Fig. 23) show that the bilabial closure is associated with not only a peak in |LA clo| activation but also a smaller peak in |LA rel|. The same holds for the bilabial release and for other antagonist pairs.

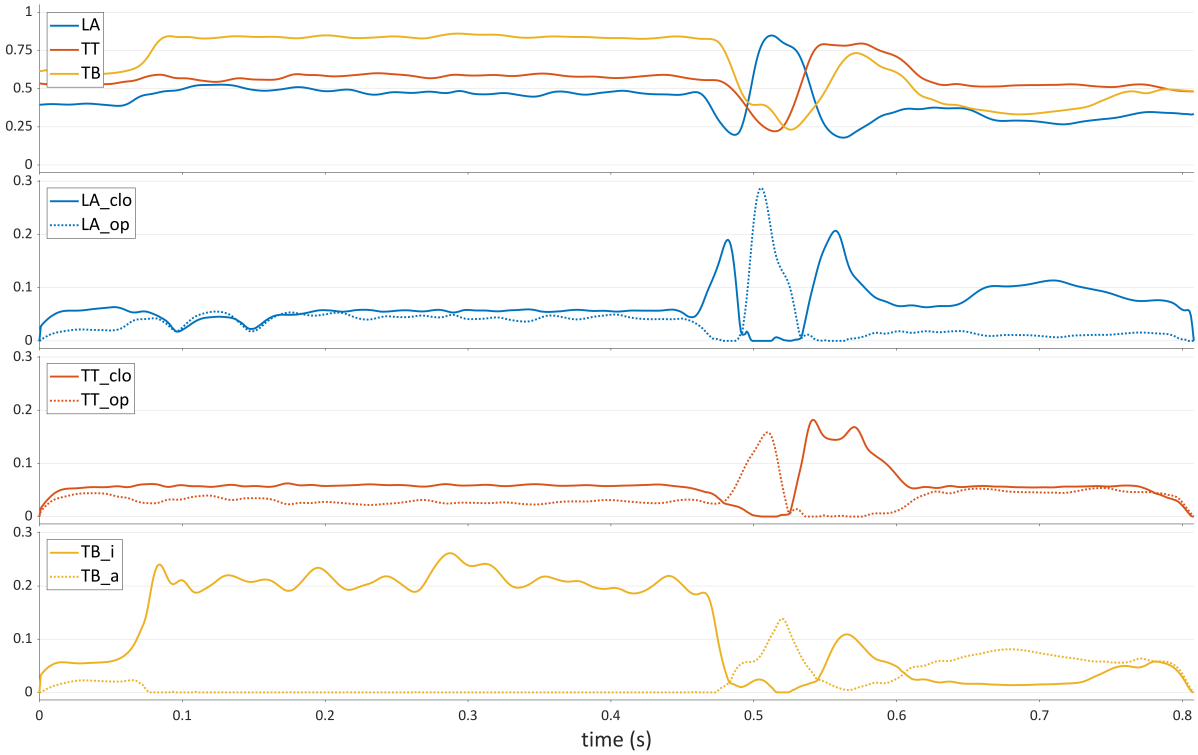

Fig. 22. Example of optimized gestural activation functions obtained from zero-initialization, with the utterance /pat/. Top panel: empirical tract variables. Bottom three panels: gestural activation functions for antagonist pairs.

The second main difference is that the activation values of peaks in the optimized gestural activation functions are larger for the gradient-based strategy than for the zero-initialization strategy. This can be seen by examing the gestural functions which are overlayed in Fig. 24. It is likely that this difference is a consequence of complementary activation vs. co-activation: because antagonistic gestures are co-active in the gradient-based scheme, the gesture which is primarily responsible for a change in the relevant tract variable (e.g. |LA clo| for the bilabial closure at the onset of the target response) must have relatively higher activation to overcome the effects of the co-active antagonist.

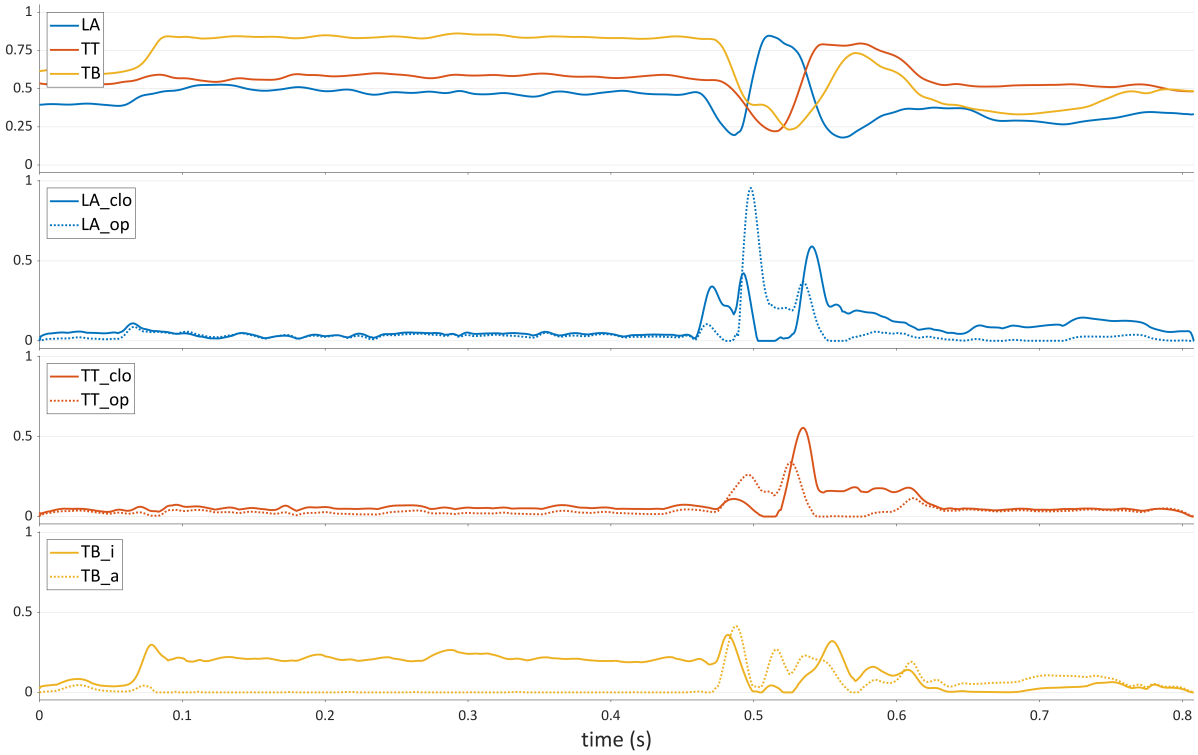

Fig. 23. Example of optimized gestural activation functions obtained from gradient-based initialization, with same token of /pat/ from Fig. 22. Top panel: empirical tract variables. Bottom three panels: gestural activation functions for antagonist pairs.

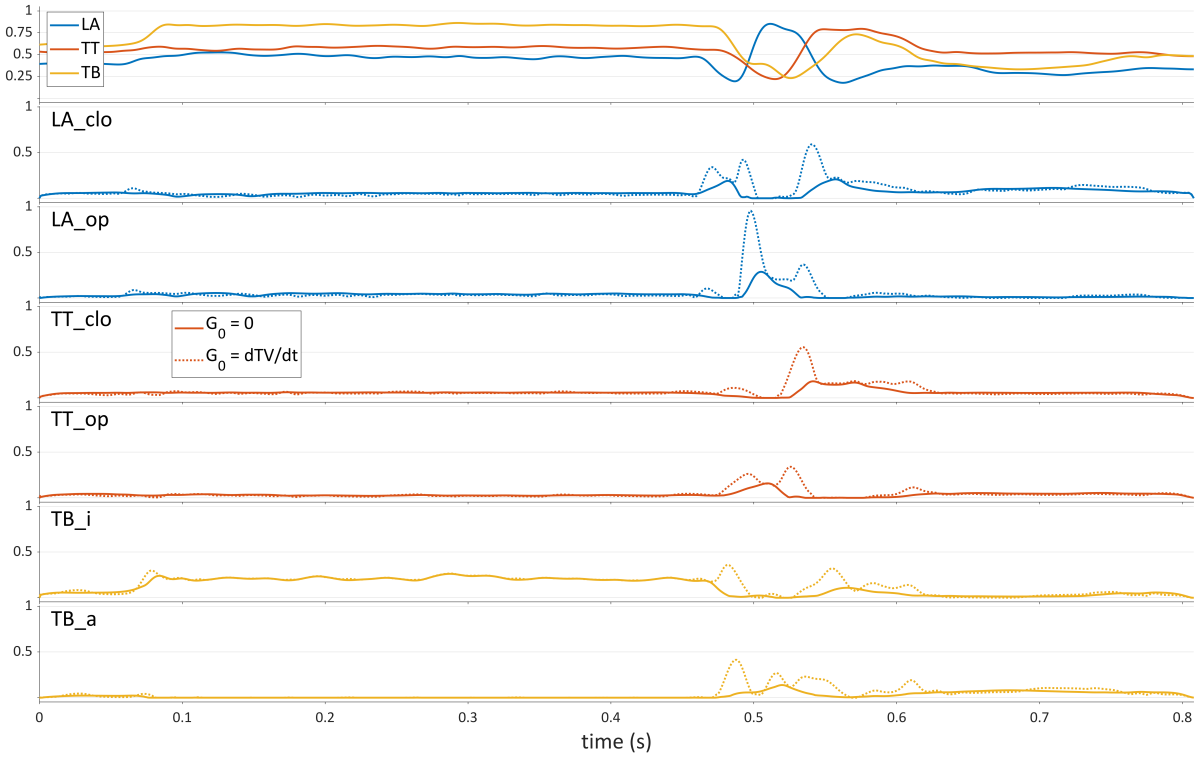

Fig. 24. Comparison of optimized gestural activation functions obtained from zero- and gradient-based initialization, with same token of /pat/ from Fig. 22. Top panel: empirical tract variables. Bottom panels: gestural activation functions.

Gestural activation functions obtained from zero-initialization: general properties

For reasons that we elaborate in the Discussion section, we focus subsequent analyses on gestural activation functions obtained from zero-initialization. There are several aspects of these functions that we examine: small persistent levels of activation pre- and post-response, variation in pre- and post-response activation levels, activation associated with passive consequences of movement, activation peaks with quasi-Gaussian profiles, phase separation of activation peaks for antagonistic gestures, and apparent hypo- and hyper-activation of consonantal closure gestures (i.e. |LA clo| and |TT clo|) when the same gesture occurs in both the onset and coda position of the response (i.e. for /pap/ and /tat/ forms).

Examination of average activation trajectories shows that gestures tend to have small, persisent levels of activation before and after the response. Fig. 25 illustrate this by showing the mean activation tractories for each participant for |LA clo| and |LA op| gestures in /pa/, /ta/, and /Øa/ responses. Not surprisingly, during the response (circa time 0, which is the vowel onset), there is an activation peak for |LA clo| for /pa/ responses but not for /ta/ and /Øa/ responses. Before the response, both |LA clo| and |LA op| have small but non-zero activation levels of about 0.05, with some variation across participants. These presumably reflect the fact that the labial posture adopted during the pre-response /i/ is not equivalent to the LA neutral attractor target of 0.5, and hence some LA clo/op activation is necessary to adjust the posture.

There is a greater amount of post-response |LA clo| activation than pre-response, and more across-participant variation. This pattern reflects inter-participant differences in the post-response labial posture: some participants tend to adopt a closed or nearly closed labial posture at the end of each trial.

For instance, P02 shows similar levels of activation pre- and post-response. Contrast this with P01, where |LA clo| is highly active after the response. Similar patterns are observed for TT gestures (not shown).

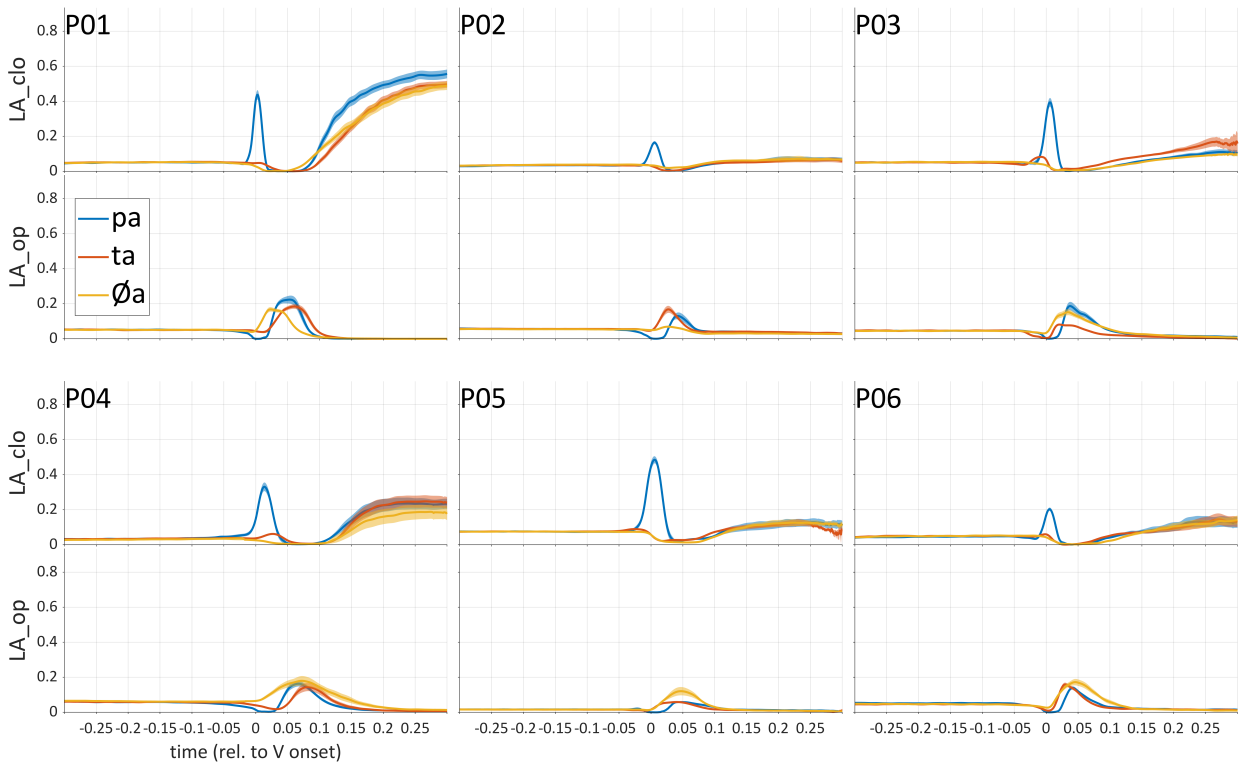

Fig. 25. Average optimized gestural activation functions for |LA clo| and |LA op| gestures for /pa/, /ta/, and /a/ responses in the basic model. Lines show means for each participant/response category and 95% confidence intervals.

In interpreting the activation patterns it is important to remember that the model does not capture biomechanical interactions between tract variables. Hence one possible hypothesis is that the labial posture during the pre-response vowel /i/ is a consequence of tongue-jaw-lip coupling: the TB [i] gesture drives TBCD (tongue body constriction degree) toward a target state, and this is accomplished by movements of the jaw and tongue body; because the lips are coupled to the jaw, there is a passive effect on the jaw. Hence the subthreshold activation patterns observed pre- and post-response may reflect passive biomechanical effects rather than active gestural influences.

The phenomenon of activation associated with passive effects may also be evident in |LA op| activation trajectories during the period of time associated with the vowel of the target response (beginning around time 0.05 s in Fig. 25). For all of the response categories shown in the figure, there is |LA op| activation during this period. This is not unexpected, given that the posture for /a/ involves a relatively open vocal tract, including a wide lip aperture. The lip aperture state during the vowel may be a passive biomechanical effect: the jaw is lowered to lower the tongue body (and/or retract the tongue root), and the lower lip passively lowers because of its connection to the jaw. On the other hand, lip aperture could also be an actively controlled state variable, with an |LA op| gesture being coupled to the |TB [a]| gesture.

A closer view of the |LA clo| and |LA op| average gestural activation functions during the response onset (Fig. 26) shows that the average profile of the |LA clo| activation peak associated with the bilaibal closure is quasi-Gaussian. There are notable deviations from this profile that are attributable to variation

in the pre- and post-peak activation levels. The quasi-Gaussian profile is also observed for most of the participants for the bilabial closure in coda (Fig. 27). The exception is P04, who exhibits a sigmoidal profile due to production of unreleased codas, i.e. [pap˺].

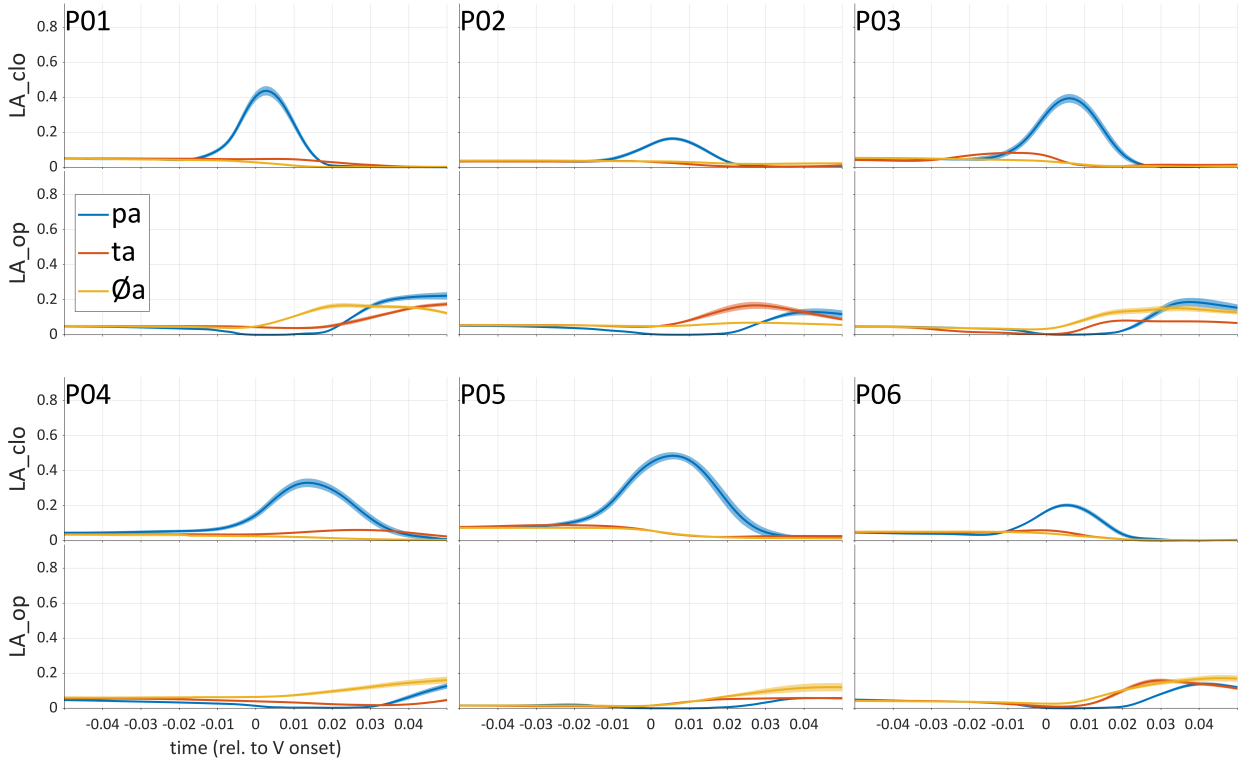

Fig. 26. Response-onset view of optimized gestural activation functions for |LA clo| and |LA op| gestures for /pa/, /ta/, and /a/ responses in the basic model. Lines show means for each participant and response category, along with 95% confidence intervals.

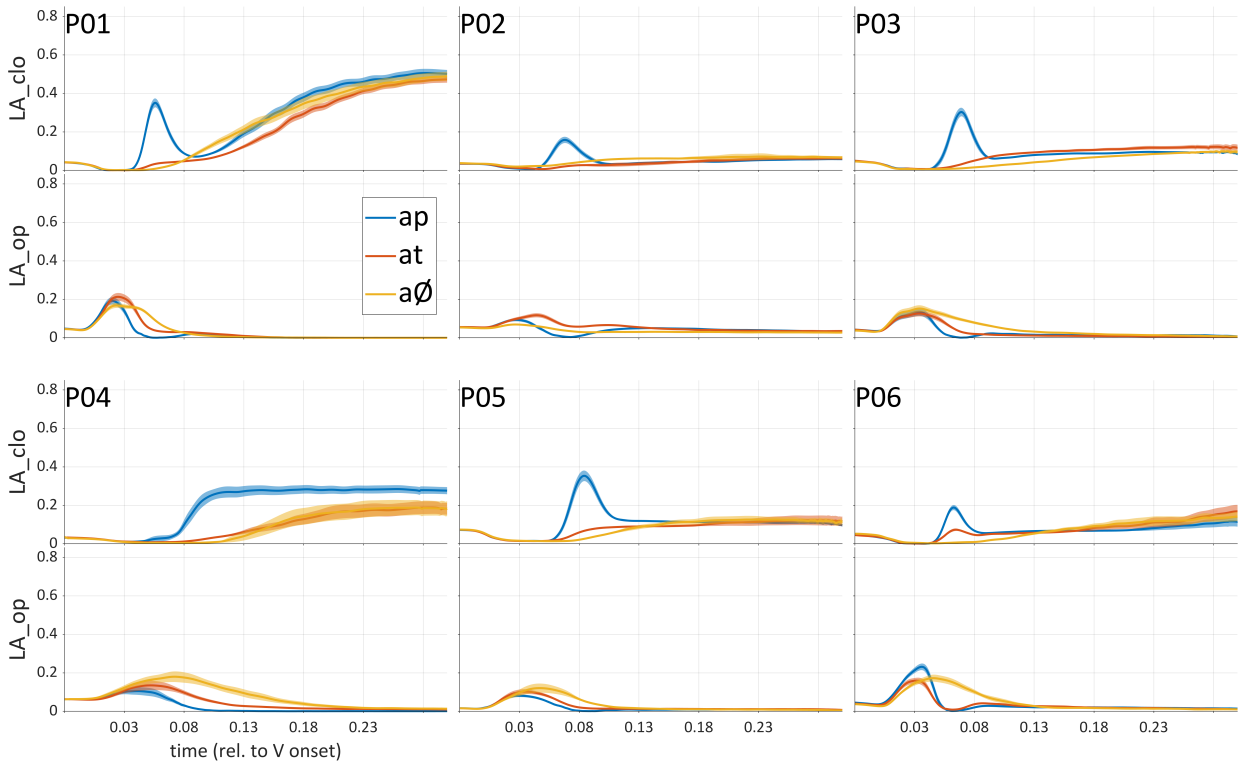

Fig. 27. Gestural activation functions for |LA clo| and |LA op| gestures for /ap/, /at/, and /a/ responses in the basic model. Lines show means for each participant/response category and 95% confidence intervals.

As mentioned previously, antagonistic clo/op activation peaks are complementary for zero-initialization. Another way to visualize this is via the phase portraits in Fig. 28, which plot |LA clo| vs |LA op| gestural activation on the time interval [-0.050, 0.050] for /pa/, /ta/, and /a/ responses. The markers in each panel indicate the beginning of the time interval (circles), time 0 (squares), and the end of the time interval (diamonds). Line colors also grow darker over time in each portrait. To interpret the phase portraits, note that values below the diagonal line in each panel corresponds to times when |LA clo| activation is higher than |LA rel| activation, and vice versa for values above the diagonal. If the activation peaks were perfectly out-of-phase, the phase portraits would be L-shaped: |LA op| activation would remain at zero while |LA clo| activation makes a horizontal excursion, and then |LA clo| activation would remain at zero while |LA op| makes a vertical exclusion. The deviations from perfect phase separation are quite small, and thus the patterns may be consistent with analyses in which closure and release gestures are controlled via anti-phase coordination of coupled oscillators (Nam, 2007; Tilsen, 2017, 2020).

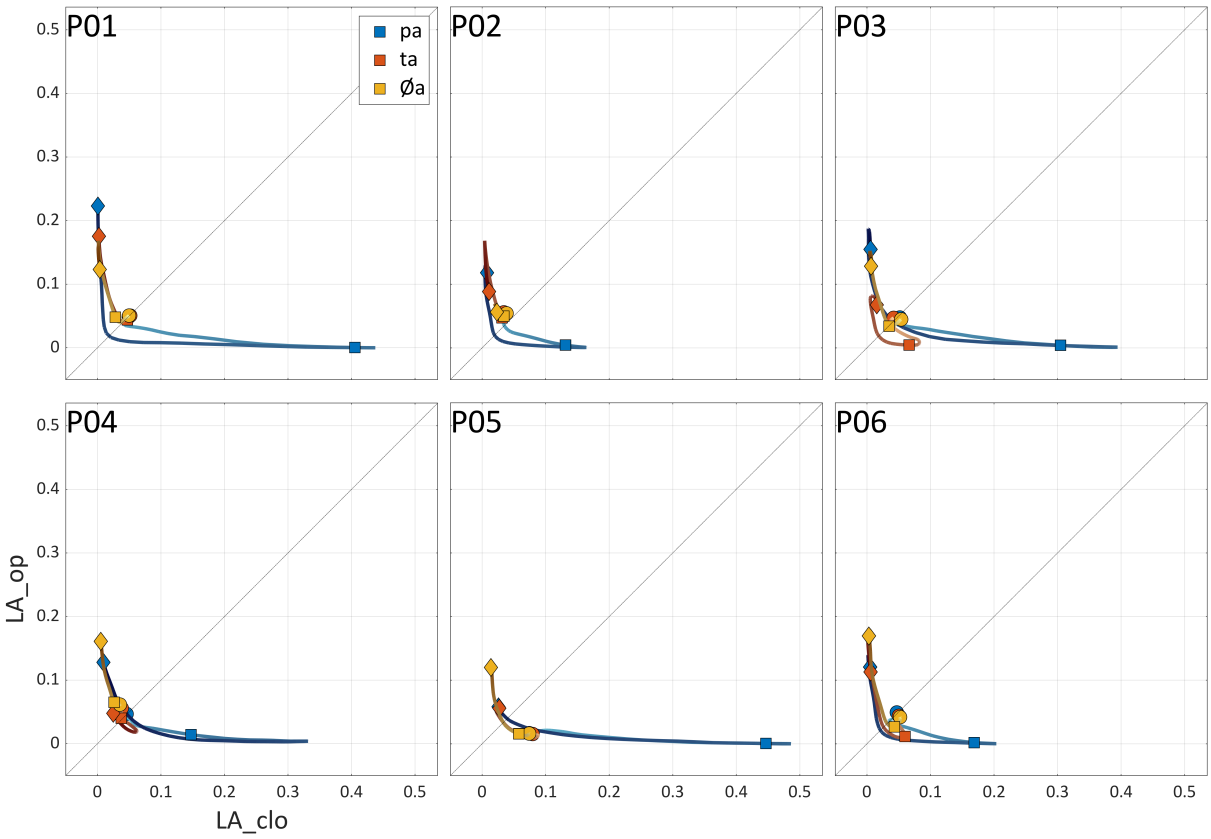

Fig. 28. Phase plots of |LA clo| vs |LA op| mean gestural activation on the time interval [-0.050, 0.050] for /pa/, /ta/, and /a/ responses in the basic model. Markers indicate beginnings (circles), time 0 (squares), and ends (diamonds) of trajectories. Line colors grow darker over time.

Another interesting pattern involves variation in the peaks of gestural activation functions that are conditioned by onset-coda gestural identity: activation peaks for codas may be hyper- or hypo-active relative to onset peaks, depending on the participant. These pattern are illustrated in Fig. 29, which shows gestural activation functions of |LA clo| and |LA op| for /pap/, /tap/, and /ap/ responses. For P02, P03, and P05, the activation peak associated with the coda bilabial closure is lower in /pap/ (blue lines) than in /tap/ (red line). This can described as a hypo-activation of the |LA clo| gesture. In contrast, for P01 and P02, the same peak is higher in /pap/ than in /tap/; this can be described as hyper activation. The interpretation of this variation is complicated by the fact that the relative activation peaks between /ap/ and /pap/ differ as well.

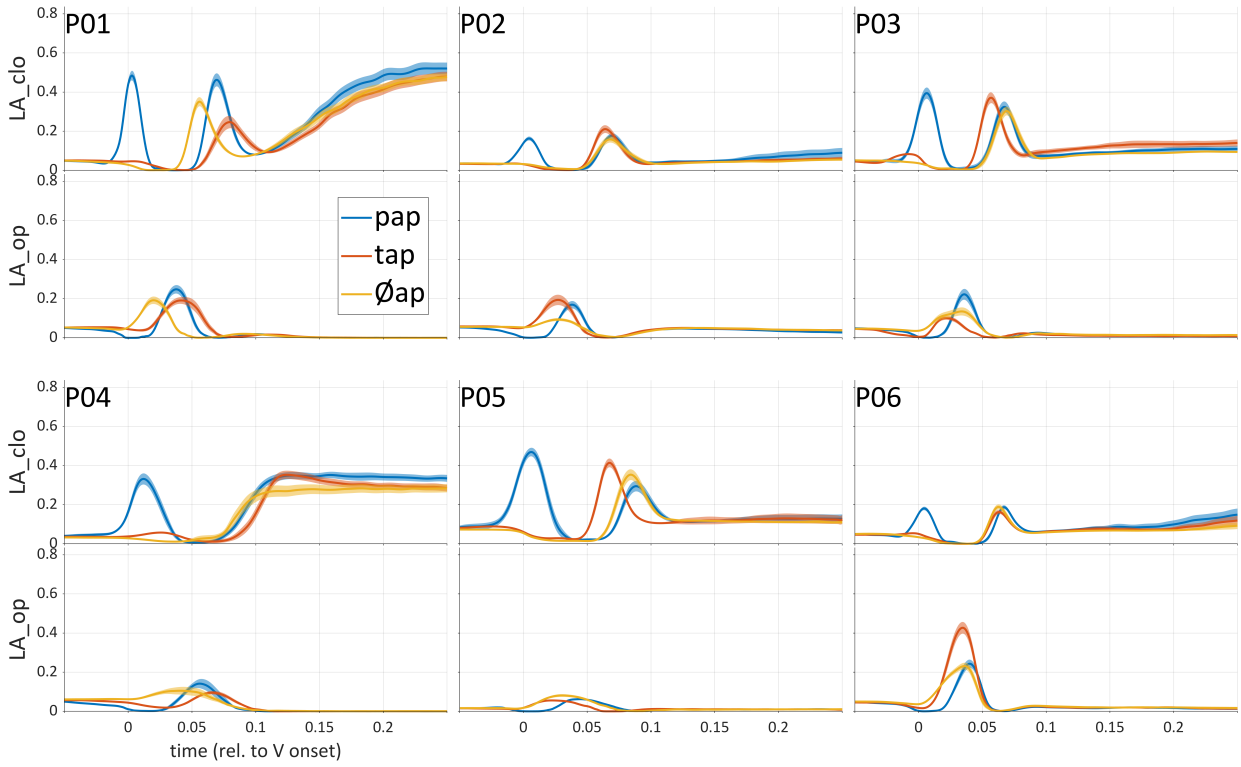

Fig. 29. Average gestural activation functions for |LA clo| and |LA op| gestures for /pap/, /tap/, and /ap/ responses. Lines show means for each participant/response category, and filled regions show 95% confidence intervals.

The causes of these differences in maximal activation warrant further investigation. One possibility is that onset-vowel coarticulation affects the vocalic posture (see below), which thereby induces changes in the gestural activation that is necessary to accomplish a coda constriction. Another possibility is that the activation amplitudes interact with the target and/or stiffness parameters. Indeed, the recentered across-subject parameter values for zero-initialization in Fig. 17 showed that the stiffnesses of |LA clo| were higher for /pap/ responses than /tap/ responses. On the other hand, it is not clear why amplitudes would interact with stiffness in this way. A third possibility, and perhaps the most interesting one, is that mechanisms which are not incorporated in the model cause a gesture that is repeated in a word form to have diminished activation. This could arise from habituation-like attenuation of spiking rate in the premotor neural population that encodes a gesture.

As with TT and LA gestures, |TB [i]| and |TB [a]| gestural activation functions exhibit complementary activation and passive effects; however, they lack quasi-Gaussian activation profiles. Fig. 30 shows average gestural activation functions for these gestures for all target response categories. The fact that the activation profiles are not quasi-Gaussian may be related to the fact that vowel segments appear not to be comprised of constriction and release gestures.

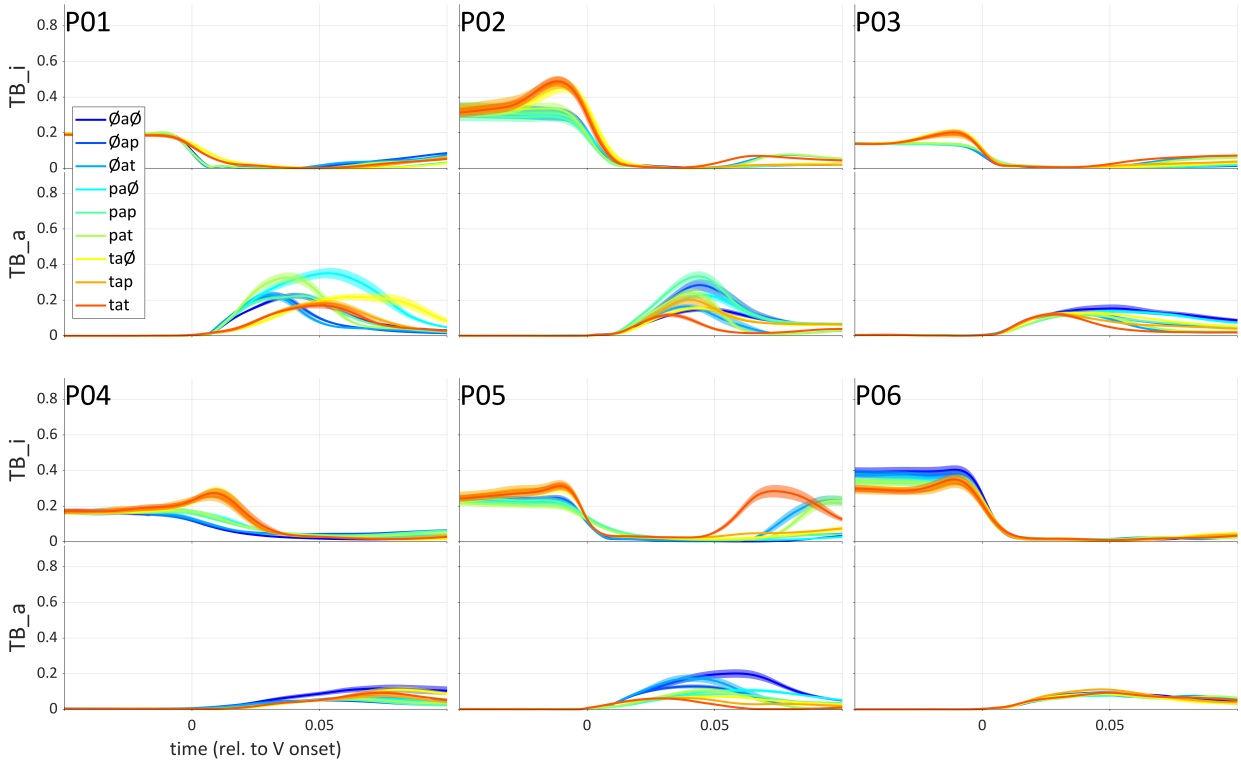

Fig. 30. Gestural activation functions for |TB [i]| and |TB [a]| gestures for all responses. Lines show means for each participant/response category and 95% confidence intervals.

Regarding passive effects in TB gestural activations, observe that for some of the participants there are higher levels of |TB [i]| activation at response onset for targets which include a |TT clo| onset gesture (i.e. /tap/, /tat/, and /ta/). This extra |TB [i]| activation may be attributable to jaw raising in support of the |TT clo| gesture. Of course, a similar effect could also be expected for /p/ onsets, but the relative sizes of these effects may differ and vary across participants. A related case of passive effect involves |TB [a]| activation, which during production of the vowel tends to exhibit lower maximal activation when there is a |TT clo| gesture in its environment. Presumably jaw raising for |TT clo| is responsible for this effect as well.

Discussion and conclusion

Overall, the analyses show that the optimized parameters and activation functions depend strongly on initialization. This dependency was observed despite the fact that both zero- and gradient-based initializations resulted in high-quality fits of empirical tract variables and had highly correlated loss across trials. Consider that the initialization space is very high-dimensional: the basic models include dimensions not only for all gestural target and stiffness parameters, but also for the value for each gestural system at each time step, of which there were about 800 for each trial. Thus there are around (6 x 2) + (6 x 800) = 4812 parameters in each optimization, most of which are dimensions for gestural activation. The high correlation of loss between strategies indicates that even though the two initialization strategies begin in very different locations in the activation function subspace, both are able to converge to different regions of the parameter space which result in similar-quality fits. This tell us that the network model in its current form cannot be used to identify gestural parameters and activation functions that are independent of initialization, or which necessarily reflect a global optimum.

Thus we should ask: are there reasons to prefer one initialization strategy over the other? Relatedly, are there general desirable properties of the network outputs which can be used to guide the preference? Earlier we noted that in typical deep learning contexts, parameters are often initialized to random values. Although this approach has not been tested systematically with the current network and datasets, it seems unlikely to provide useful results, for a couple reasons. First, regarding target parameters, there are desirable relations between the targets of antagonistic gestures, such that closure (clo) and release (op) targets are on opposite ends of a tract variable coordinate. Allowing random initialization to subvert these relations could lead to exchanges in the "meaning" (i.e. functional results) of clo and op gestures, and might fail under gradient-based initialization because the activation functions are designed with these differences in mind.

Second, the gestural activation functions should be obey a mild smoothness constraint, such that there do not exist very abrupt changes in their values. Both initialization schemes impose this constraint on the initial activation functions. In the case of the gradient-based scheme, smoothness is imposed because tract variable derivatives are effectively smooth (in a relative sense), both before and after the rectification. In both cases, the optimized activation functions are also mildly smooth, even though the gradient-based initialization results is somewhat less smooth functions. It is not clear exactly why this happens: there is nothing explicit in the optimization algorithm that imposes this constraint. It may be a consequence of the fact that empirical tract variables change smoothly and the model requires smooth inputs to generate smooth outputs (again, our sense of *smooth* here is relative).

Although both initialization strategies provide good fits and relatively smooth activation functions, they differ in a several important ways. First, zero-initialization is simpler than gradient-based-initialization, because it does not make use of the empirical tract variable data. Second, zero-initialization results in more consistency in parameter values across trials. It is not clear why this is the case. It could be that because gradient-based initialization begins with activation values that are closer to the optimized ones, there is—in the course of the optimization—relatively less movement in activation dimensions and relatively more in the target and stiffness parameter dimensions. In contrast, zero-initialization may be dominated by movement in activation dimensions for a greater proportion of optimization iterations, because the initial error gradients in these dimensions will be larger.

Third, the optimized activation states of antagonistic gestures are differently related depending on the initialization. Gradient-based initialization resulted in co-activation of antagonists. This is an interesting pattern because it is analogous to muscular control, where movements accomplished by an agonist may involve slight activation of an antagonist. The balance between the effects of opposing systems may result in overall greater stability. Zero-initialization, on the other hand, was associated with complementary activation of antagonists. This may be desirable because it effectively reduces the entropy of the gestural system, by introducing a negative correlation between antagonists that is stronger than the positive correlation observed under gradient-based initialization. For related reasons, zero-initialization also requires less total gestural activation—if some form of activation-based energy conservation principle applies to the control system, this would be an advantage.

On the basis of the above differences, I am inclined to prefer the zero-initialization strategy. However, caution is warranted in developing a strong preference, because the effects of relative learning rates on the optimization results are not yet understood. Specifically, we used a learning rate for gestural activation that was 3 orders of magnitude greater the learning rates for target and stiffness parameters. This large ratio was not systematically derived, although pilot optimizations indicated that lowering the ratio of learning rates too much often resulted in instability of loss over iterations and convergence failure.

The main advantage of the RNN approach over standard methods of analyzing articulatory time series is that it does not require us to impose any temporal delimitation on articulatory gestures. It is fully consistent with the systems-conception of articulatory gesture and provides a new version of an old

theoretical entity—the gestural activation function—which can be analyzed in novel ways. However, as should be clear from the above exposition, the potential utility of this entity is compromised in a couple of ways: namely, the results of optimizations are non-unique (they depend on initialization), and the network neglects to model articulators, thereby conflating passive and active gestural effects.

The latter issue may be resolved by incorporating model articulator position and velocity nodes into the network, along with parameters that represent the weights of the pseudo-inverse Jacobian matrices in the SM89 model. These parameters describe the relative influences of tract variable changes on changes in articulator positions/velocities and should be learnable via the backpropagation methods employed here. The loss would then be calculated directly based on the articulators, i.e. the horizontal and vertical positions of sensors. Such an approach would allow for implicit learning of parameters which are otherwise quite difficult to estimate (Lammert et al., 2010). It is also possible that the output space could be a real-time MRI image, where the network learns to generate pixel intensity maps from gestural activation functions.

The non-uniqueness problem is more challenging to address. To some extent, it seems that certain choices must be made (and hopefully motivated) based on desirable properties of the input and static parameters. For example, consistency in gestural target parameters across response categories may be desirable because it amounts a more parsimonious account of the long-term memories associated with gestures (and indeed, many descriptions of the AP/TD framework adopt a hypothesis of gestural target/stiffness invariance). On the other hand, there may be a trade-off between less variation in target/stiffness parameters and more variation in gestural activation functions, which raises the question: what causes variation in gestural activation? Here we need new, creative ideas for mechanisms of causation in the next highest level of the never-ending hierarchy of causality. Some of my previous work has attempted to incorporate competitive queuing dynamics (Bullock, 2004; Bullock & Rhodes, 2002; Grossberg, 1978, 1987) as a level of control/organization above the gestural level (see Tilsen, 2013, 2016, 2018, 2019b).

The non-uniqueness issue also relates to another topic of investigation, which is the effects of relative learning rates and parameter constraint on optimization results. It is not clear to what extent the optimizations obtained here depend on the large difference in learning rates for gestural activation vs. target/stiffness parameters. This issue might be productively addressed by using algorithms with adaptive dynamic learning rates, which are common in deep learning contexts.

The field model in its current form provided lower-quality fits than the basic model. The reason for this is that there were no free parameters to scale the areas of the activation fields that were used to calculate dynamic stiffness. This problem may be readily addressed by incorporating one optimizable parameter, or tract-variable specific parameters.

Finally, the task of developing methods for reducing the dimensionality of the gestural activation function remains to be undertaken. It is evident from inspection of activation functions that they will lend readily to extremum/threshold-based methods that are typically applied directly to tract variables. By applying these methods instead to activation functions, it is easier to keep in mind that the resulting "activation intervals" are based on arbitrary decisions about when a gestural system is sufficiently active to warrant the postulation of a bounded interval. More to the point, if the model can be successfully extended along the lines proposed above—i.e. to generate articulator trajectories—and if constraints on parameters can be reasonably justified, then the activation functions learned by the model can be viewed as theoretical entities which are themselves interesting objects of analysis.

References

Bullock, D. (2004). Adaptive neural models of queuing and timing in fluent action. *Trends in Cognitive Sciences*, *8*(9), 426–433.

Bullock, D., & Rhodes, B. (2002). Competitive queuing for planning and serial performance. *CAS/CNS Technical Report Series*, *3*(003), 1–9.

Chen, T. Q., Rubanova, Y., Bettencourt, J., & Duvenaud, D. K. (2018). Neural ordinary differential equations. *Advances in Neural Information Processing Systems*, 6571–6583.

Dupont, E., Doucet, A., & Teh, Y. W. (2019). Augmented neural odes. *Advances in Neural Information Processing Systems*, 3134–3144.

Grossberg, S. (1978). A theory of human memory: Self-organization and performance of sensory-motor codes, maps, and plans. *Progress in Theoretical Biology*, *5*, 233–374.

Grossberg, S. (1987). The adaptive self-organization of serial order in behavior: Speech, language, and motor control. *Advances in Psychology*, *43*, 313–400.

Lammert, A. C., Goldstein, L., & Iskarous, K. (2010). Locally-weighted regression for estimating the forward kinematics of a geometric vocal tract model. *Eleventh Annual Conference of the International Speech Communication Association*.

Nam, H. (2007). Syllable-level intergestural timing model: Split-gesture dynamics focusing on positional asymmetry and moraic structure. In *In J. Cole & J. I. Hualde (Eds.), Laboratory phonology* (Vol. 9, pp. 483–506). Walter de Gruyter.

Saltzman, E., & Munhall, K. (1989). A dynamical approach to gestural patterning in speech production. *Ecological Psychology*, *1*(4), 333–382.

Tilsen, S. (2013). A Dynamical Model of Hierarchical Selection and Coordination in Speech Planning. *PloS One*, *8*(4), e62800.

Tilsen, S. (2016). Selection and coordination: The articulatory basis for the emergence of phonological structure. *Journal of Phonetics*, *55*, 53–77.

Tilsen, S. (2017). Exertive modulation of speech and articulatory phasing. *Journal of Phonetics*, *64*, 34–50.

Tilsen, S. (2018). *Three mechanisms for modeling articulation: Selection, coordination, and intention* (Cornell Working Papers in Phonetics and Phonology 2018).

Tilsen, S. (2019a). Motoric mechanisms for the emergence of non-local phonological patterns. *Frontiers in Psychology*, *10*, 2143.

Tilsen, S. (2019b). *Syntax with oscillators and energy levels*. Language Science Press.

Tilsen, S. (2020). *Detecting anticipatory information in speech with signal chopping*.

Weinan, E. (2017). A proposal on machine learning via dynamical systems. *Communications in Mathematics and Statistics*, *5*(1), 1–11.

Appendix: backpropagation

Basic model backpropagation

The derivative of the loss, $L_i(t)$ for TV *i* with respect to the output $X_i(t)$, is simply the error, i.e. the difference between the model output and target output, $Y_i(t)$, (Eq. 27). In the current design, this is the only contribution to the loss that comes from the state of the network at time *t*. Other contributions to $L_i(t)$ come from the position and velocity at the previous time step, $X_i(t-1)$ and $V_i(t-1)$. This is where backpropagation through time becomes necessary: because the current error is determined by the current position of a TV system, and in turn the current position of a TV system is determined by its previous state (position and velocity), the error at time t must be attributed in part to $X_i(t-1)$ and $V_i(t-1)$.

$$\frac{\partial L_i(t)}{\partial X_i(t)} = E_i(t) = X_i(t) - Y_i(t) \quad \text{(Eq. 27)}$$

The portion of the error associated with the current position, which is due to the previous position, is:

$$\frac{\partial X_i(t)}{\partial X_i(t-1)} = 1 \quad \text{(Eq. 28)}$$

And the portion of the error associated with the current position due to the previous velocity is:

$$\frac{\partial X_i(t)}{\partial V_i(t-1)} = \Delta t \quad \text{(Eq. 29)}$$

The error associated with the velocity $V_i(t)$ is attributable to the previous position and velocity; thus backpropagation through time is also required. The relevant partial derivatives are shown in (Eqs. 30 and 31).

$$\frac{\partial V_i(t)}{\partial X_i(t-1)} = -\Delta t K_i(t) \quad \text{(Eq. 30)}$$

$$\frac{\partial V_i(t)}{\partial V_i(t-1)} = 1 - 2\Delta t\sqrt{K_i(t)} \quad \text{(Eq. 31)}$$

The error in the current velocity also depends on the error in the current dynamic stiffness and target, so we use the partial derivatives in (Eq. 32, 33) to backpropagate error to the stiffness and target nodes:

$$\frac{\partial V_i(t)}{\partial K_i(t)} = -\Delta t[K_i(t)^{(-1/2)}V_i(t-1) + X_i(t-1) - T_i(t)] \quad \text{(Eq. 32)}$$

$$\frac{\partial V_i(t)}{\partial T_i(t)} = \Delta t K_i(t) \quad \text{(Eq. 33)}$$

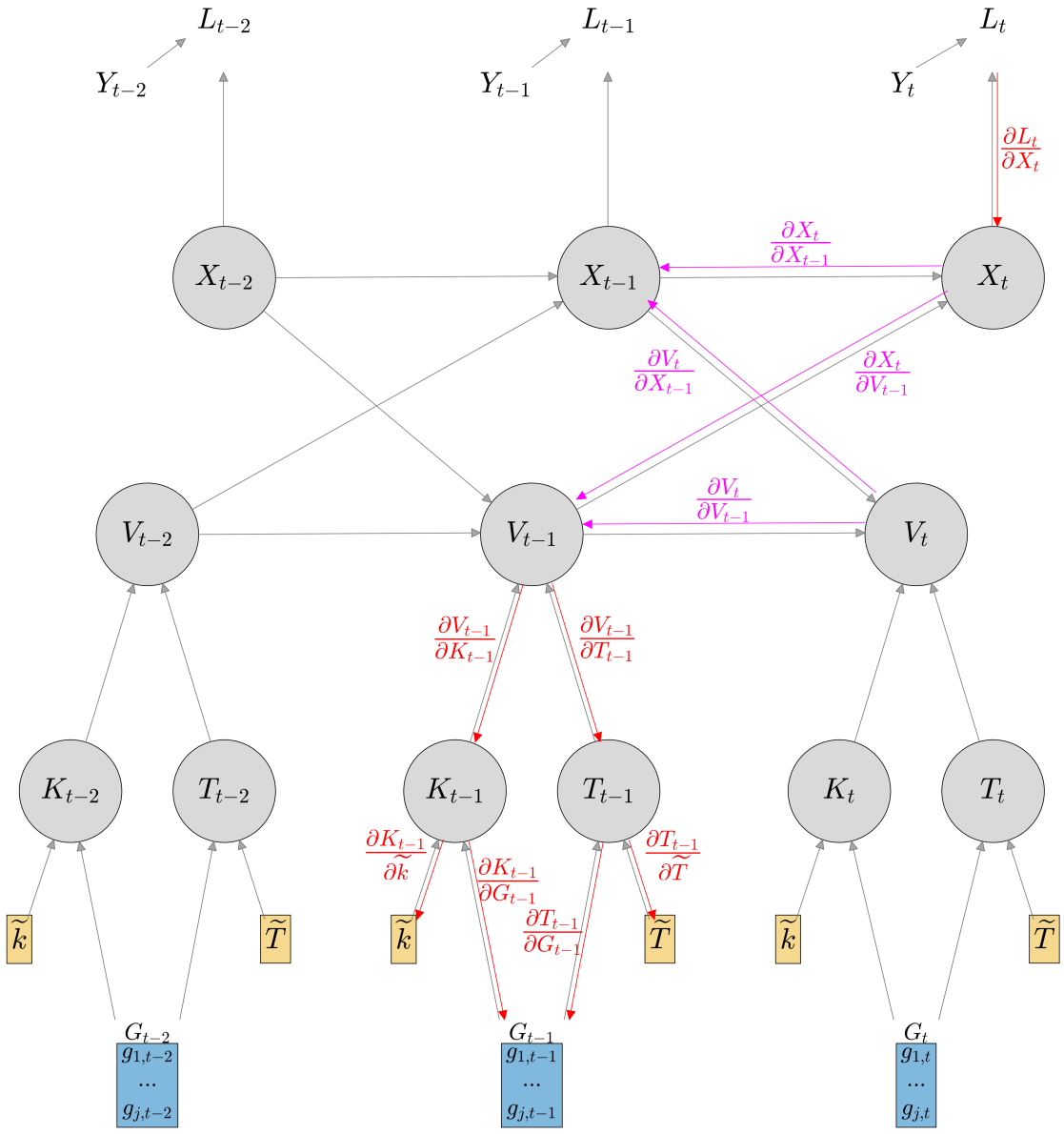

Fig. 31. Illustration of error flow in basic model.

Finally, we can backpropagate error to the parameters and activation functions. Part of the error in $K_i(t)$ and $T_i(t)$ is attributable to the gestural parameters. For each gesture *j*, the partial derivatives which modulate error flow from the dynamic parameters to the gesture parameters are shown in (Eq. 34, 35). Note that W_{ij} is the gesture-to-TV interaction matrix.

$$\frac{\partial K_i(t)}{\partial \tilde{k}_j} = \frac{W_{ij} G_j(t)}{\sum_j W_{ij}\, G_j(t)} \qquad \text{(Eq. 34)}$$

$$\frac{\partial T_i(t)}{\partial \tilde{T}_j} = \frac{W_{ij} G_j(t)}{\sum_j W_{ij}\, G_j(t)} \qquad \text{(Eq. 35)}$$

The other part of the error associated with the dynamic parameters is attributable to gestural activation, because gestural activation is the weighting term in the forward equations (Eq. 17, 18). The partial derivatives which regulate this flow are shown in (Eq. 36, 37).

$$\frac{\partial \mathrm{K_i(t)}}{\partial \mathrm{G_j(t)}} = \frac{\mathrm{W_{ij}\tilde{k}_j \sum_j W_{ij}\, G_j(t) - W_{ij} \sum_j W_{ij}\, G_j(t)\tilde{k}_j}}{\left[\sum_j \mathrm{W_{ij}\, G_j(t)}\right]^2} \quad \text{(Eq. 36)}$$

$$\frac{\partial \mathrm{T_i(t)}}{\partial \mathrm{G_j(t)}} = \frac{\mathrm{W_{ij}\tilde{T}_j \sum_j W_{ij}\, G_j(t) - W_{ij} \sum_j W_{ij}\, G_j(t)\tilde{T}_j}}{\left[\sum_j \mathrm{W_{ij}\, G_j(t)}\right]^2} \quad \text{(Eq. 37)}$$

Field model backpropagation

The partial derivatives used for backpropagation of error to position, velocity, stiffness, and target nodes are the same as those in the basic model. For the backpropagation of error to the intentional planning fields, there are two sources of error: from the dynamic stiffness, and from the dynamic target. The relevant partial derivatives are shown in (Eq. 38, 39).

$$\frac{\partial K_i(t)}{\partial F_{ik}(t)} = \frac{1}{\Delta t} \quad \text{(Eq. 38)}$$

$$\frac{\partial T_i(t)}{\partial F_{ik}(t)} = \frac{\tau_k \sum_{k=1}^{n} F_{ik}(t) - \sum_{k=1}^{n} F_{ik}(t)\tau_k}{[\sum_{k=1}^{n} F_{ik}(t)]^2} \quad \text{(Eq. 39)}$$

The intentional planning fields depend on the excitatory and inhibitory components of the field, and so error is propagated to these components using the partial derivatives in (Eq. 40, 41).

$$\frac{\partial F_{ik}(t)}{\partial F_{ik}^{+}(t)} = \begin{matrix} 1, & F_{ik}(t) > 0 \\ 0, & F_{ik}(t) = 0 \end{matrix} \quad \text{(Eq. 40)}$$

$$\frac{\partial F_{ik}(t)}{\partial F_{ik}^{-}(t)} = \begin{matrix} -1, & F_{ik}(t) > 0 \\ 0, & F_{ik}(t) = 0 \end{matrix} \quad \text{(Eq. 41)}$$

Both excitatory and inhibitory components depend on the gestural activation. (Eqs. 42 and 43) describe how error is propagated from the components to gestural activation.

$$\frac{\partial F_{ik}^{+}(t)}{\partial G_j(t)} = W_{ij}\mathcal{N}\left(\tau_k, \tilde{T}_j^{+}\right) \quad \text{(Eq. 42)}$$

$$\frac{\partial F_{ik}^{-}(t)}{\partial G_j(t)} = W_{ij}\mathcal{N}\left(\tau_k, \tilde{T}_j^{-}\right) \quad \text{(Eq. 43)}$$

The excitatory and inhibitory components also depend on the central values of the Gaussian force distributions associated with gestures, as shown in (Eq. 44, 45).

$$\frac{\partial F_{ik}^{+}(t)}{\partial \tilde{T}_j^{+}} = \frac{\left(\tau_k - \tilde{T}_j^{+}\right)}{\sigma}\mathcal{N}\left(\tau_k, \tilde{T}_j^{+}\right)W_{ij}G_j(t) \quad \text{(Eq. 44)}$$

$$\frac{\partial F_{ik}^{-}(t)}{\partial \tilde{T}_j^{-}} = \frac{\left(\tau_k - \tilde{T}_j^{-}\right)}{\sigma}\mathcal{N}\left(\tau_k, \tilde{T}_j^{-}\right)W_{ij}G_j(t) \quad \text{(Eq. 45)}$$

Matlab implementation and gradient checking

Here Matlab code for the basic model is provided. This code can be used to implement the forward and backward passes of the network. Gradient checking can be implemented for various parameters by comparing a gradient obtained via backpropagation to a numeric gradient obtained by imposing small perturbations of ±ε to a single parameter. For example, to check that $\frac{\partial L}{\partial G_j(t)}$, the gradient of the loss with respect to change in gesture *j* at time *t*, is being calculated correctly in the backward pass, calculate L_1 as the loss from a forward pass with $G(t,j)_1 = G(t,j) - \varepsilon$, and L_2 as the loss from a forward pass with $G(t,j)_2 = G(t,j) + \varepsilon$, and calculate the numeric gradient $\frac{\partial \mathrm{L}}{\partial G_j(t)} = \frac{L_2 - L_1}{2\epsilon}$. Then compare the numeric gradient to the value of dG(t,j) from the backward pass. These values should differ by only a small amount that is close to machine precision.

```
%{
variables:
dt:         time step;
Ntv:        number of tract variables;
Ng:         number of gestures + number of neutral attractors
W:          logical matrix with map of tract variable (rows) to gestures (columns)
E(t,i):     error at time t for TV i
X(t,i):     position at time t for TV i
V(t,i):     velocity at time t for TV i
K(t,i):     stiffness at time t for TV i
T(t,i):     target at time t for TV i
dX(t,i):    gradient of loss wrt. position for TV i at time t
dV(t,i):    gradient of loss wrt. velocity for TV i at time t
G(t,j):     gestural activation at time t for gesture j
Kg(j):      gestural and neutral attractor stiffnesses
Tg(j):      gestural and neutral attractor targets
%}

%forward pass
for t=2:Nt

    %dynamic stiffness and target:
    K(t,:) = W*(G.*Kg)' ./ (W*G');
    T(t,:) = W*(G.*Tg)' ./ (W*G');

    %velocity:
    V(t,:) = V(t-1,:) + dt*(-2*sqrt(K(t,:)) .* ...
                V(t-1,:) - K(t,:).* (X(t-1,:) - T(t,:)));

    %position:
    X(t,:) = X(t-1,:) + dt*V(t-1,:);
end

%error and loss
E = X-Y;
L = sum((1/2)*E.^2);

%error associated with position and velocity at final time step is 0:
dX(Nt,:) = 0;
dV(Nt,:) = 0;
```

```
%backward pass
for t=Nt:-1:2

    %gradient of loss wrt. position is current error plus error backpropagated through time from
    the following timestep:
    dX(t,:) = dX(t,:) + E(t,:);

    %gradient of current loss wrt. to current velocity is 0:
    dL_dV = 0;

    %gradient of total loss wrt. current velocity includes the error backpropagated through time
    from the following timestep:
    dV(t,:) = dV(t,:) + dL_dV;

    %the following quantities are for calculating the backpropagation of error through time:
    %gradients of position error relative to previous position and velocity:
    dX_dX_prev = 1;
    dX_dV_prev = dt;

    %gradient of velocity error relative to previous position and velocity:
    dV_dX_prev = -dt * K(t,:);
    dV_dV_prev = 1 - 2*dt * (K(t,:).^(1/2));

    %pass error backward in time:
    dX(t-1,:) = dX(t,:) .* dX_dX_prev + dV(t,:) .* dV_dX_prev;
    dV(t-1,:) = dV(t,:) .* dV_dV_prev + dX(t,:) .* dX_dV_prev;

    %gradient of velocity error relative to stiffness and target
    dV_dK = -dt * (K(t,:).^(-1/2)) .* V(t-1,:) - dt*X(t-1,:) + dt*T(t,:);
    dV_dT = dt * K(t,:);

    %denominator and numerators for gradients (see Eqs. 13, 14)
    denom = W * G(t,:)';
    numer_dK = W * (G(t,:) .* Kg)';
    numer_dT = W * (G(t,:) .* Tg)';

    %loop over tract variables and gestures
    for i=1:Ntv
        for j=1:Ng

            %gradients of dynamic stiffness/target for TV i relative to gesture j:
            dK_dG(i,j)= (denom(i)*W(i,j)*Kg(j) - numer_dK(i)*W(i,j)) * (1/denom(i))^2;
            dT_dG(i,j)= (denom(i)*W(i,j)*Tg(j) - numer_dT(i)*W(i,j)) * (1/denom(i))^2;
        end
    end

    %loop over tract variables and gestures
    for i=1:Ntv
        for j=1:Ng

            %error backpropagated to gestural activation (accumulate over tract variables)
            dG(t,j) = dG(t,j) + dV(t,i) * (dV_dK(i)*dK_dG(i,j) + dV_dT(i)*dT_dG(i,j));
        end
    end

    %loop over tract variables and gestures
    for i=1:Ntv
        for j=1:Ng

            %gradient of dynamic stiffness/target relative to gestural parameter
            dK_dKg(i,j) = W(i,j)*G(t,j)/denom(i);
            dT_dTg(i,j) = dK_dKg(i,j);

            %accumulate error in parameters:
            dKg(j) = dKg(j) + dV(t,i) * dV_dK(i) * dK_dKg(i,j);
            dTg(j) = dTg(j) + dV(t,i) * dV_dT(i) * dT_dTg(i,j);
        end
    end
end
```