Tunable Tension for Gesture Animation

Michael Neff
mpneff@ucdavis.edu
University of California, Davis
Davis, California, USA

ABSTRACT

Variation in muscular tension has important expressive impacts on agent motion; however, it is difficult to tune simulations to achieve particular effects. With a focus on gesture animation, we introduce mass trackers, a lightweight approach that employs proportional derivative control to track point masses that define the position of each wrist. The restriction to point masses allows the derivation of response functions that support straightforward tuning of system behavior. Using the point mass as an end-effector for an inverse kinematics rig allows easy control of both loose and high tension arm motion. Examples illustrate the expressive variation that can be achieved with this tension modulation. Two perceptual studies confirm that these changes impact the overall level of tension perceived in the motion of a gesturing character and further explore the parameter space. Practical guidelines on tuning are discussed.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

character animation, gesture, muscle tension

ACM Reference Format:
Michael Neff. 2022. Tunable Tension for Gesture Animation. In *ACM International Conference on Intelligent Virtual Agents (IVA '22), September 6–9, 2022, Faro, Portugal.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3514197.3549631

1 INTRODUCTION

Intelligent virtual agents use their bodies, including their gestures, to communicate. This is particularly true when considering communication of emotion, personality and other social cues. Variation in muscular tension is a key contributor to this expressive communication [6, 12, 26, 27, 36]. There is an ebb and flow between tension and relaxation that builds variation and interest in movement. A rise in tension can serve to accent a movement [26], providing emphasis or conveying an emotion; consider the taught energy of anger or the limpness of dejection. Controlling such tension variation provides a powerful tool that allows virtual agents to adjust their expression. The approach presented here has been used to adjust the impression of both personality [37] and emotion [10].

IVA '22, September 6–9, 2022, Faro, Portugal

ACM ISBN 978-1-4503-9248-8/22/09.
https://doi.org/10.1145/3514197.3549631

Figure 1: Frames from a tense (left) and loose (right) animation.

Physical simulation provides a method for directly controlling the tension of a character by adjusting the activation of virtual "muscles". The primary challenge of the approach is that it is difficult to tune for particular expressive effects. A secondary challenge is that such approaches can be difficult to implement and integrate into character pipelines.

This paper presents a simplified approach to physical simulation that allows for the derivation of a straightforward and intuitive tension parameterization that is directly related to the expressive animation goals. The approach can be easily integrated into existing animation systems and applied to motion capture, data from machine learning or with keyframing approaches.

The approach uses a spring and damper, which is also known as a proportional-derivative (PD) controller, to attach a point-mass to the input location of the wrist (taken from the motion data). Depending on the tuning of the controller and the character's motion, there will be some error between the input wrist location and the location of the point mass. The arm is animated with an inverse kinematics rig such that the wrist follows the location of the point mass, rather than the original location of the wrist. Adjusting the gains on the spring and damper allow various physical effects, most importantly varying how tightly the motion follows the trajectory and adding secondary oscillatory effects to the motion. The approach is particularly appropriate for the animation of manual gestures. While good gesture animation involves the entire body, tension variation, and in particular the oscillatory aspects attended to here, largely involve movements of the arms, which can alternate from swinging freely to being tightly controlled.

As a first-order approximation of muscle, proportional-derivative (PD) control has long been a popular actuator in physics-based animation (e.g. [20]). The most common approaches apply a PD actuator at each degree of freedom in every joint in order to provide the torques that drive the character. As well as being computationally simple, the main advantage of focusing on point-based control instead, as is done here, is that it allows a derivation of the response function. This supports a more intuitive parameterization of the controller with two terms, steady state error and a quantity called zeta, that make it straightforward to generate desired expressive

effects. This is the main contribution of the work and described in Sec. 3. In addition, the paper reports on two user studies. One validates that this modulation of arm motion is sufficient on its own to alter the perceived tension in gesture animation. The second explores the impact of the steady state error and zeta parameters more deeply, providing useful guidance to agent designers.

2 BACKGROUND

Early work demonstrated the expressive importance of tension variation for computer animation, and proposed an antagonistic formulation that supported motion shaping, but provided limited guidance on how to tune controllers for specific effects [32]. Allen et al. [3–5] proposed novel methods for accurately tuning PD control for animation, but focused on critically damped motion. We focus on using simple physical simulation to control the oscillations present in underdamped motion, as this regime is particularly expressively rich. Other work has focused on using tension control to modulate reactions, for example to being hit or falling [41–43]. Still other work has focused on using physical parameters in an optimization framework to adjust style [31].

Wiggly splines share a similar goal to this work in allowing animators to directly control oscillatory behavior [21]. An advantage of this proposed work is that it can directly inherit variation from the already existing dynamics of the character's motion. Tension-Continuity-Bias (TC) splines [24] implement a different notion of tension: how tightly a trajectory moves from point to point. They do not directly support oscillation and the locations for the splines knots must be identified, which is not straightforward when working with a continuous motion representation, such as motion capture data.

A different body of work has advanced increasingly realistic biomechanical models for animation(e.g. [28–30]). While such approaches are impressive and may well eventually support the most realistic character animation models, they are challenging to tune and do not offer easy animator control, as targeted here.

Gesture animation has a long research history (e.g. [9]). Early work often focused on providing expressive control of motion, e.g. [11, 13, 18, 19], some using TCB splines [18, 19]. Other work applied physical simulation to the gesture space [33, 39], but these works did not provide an easy way to tune the simulation for expressive effects. Much recent work has focused on applying machine learning techniques to try to synthesize gesture sequences (e.g. [14, 16, 25]). While some of this work has specifically targeted gesture style [1, 2], control remains limited and is difficult to fine tune the expressive qualities of an embodied agent. The technique provided here could be layered on top of machine learning approaches to provide additional control. It can also be used to smooth output.

3 TENSION CONTROL

People change their muscular tension as their emotions change, generating significant and salient variation in their motion. As discussed above, we will develop a technique called "mass trackers" to provide an effective, computationally lightweight method to capture this variation for gesture animation. The approach attaches a mass to the input wrist position using a proportional derivative controller. The location of the mass, rather than the original wrist point, is used as the IK end-effector constraint for the corresponding arm. Forward simulating the movement of the mass as the target wrist moves generates various momentum effects that can be controlled by adjusting the gain and damping of the controller. Below we derive guidance on how to tune these parameters for specific effects.

Ignoring gravity, the PD controller acting on the mass is defined by Eq. 1:

$$F = k_s(x_d - x) - k_d\dot{x} = m\ddot{x} \tag{1}$$

where k_s is the proportional gain, k_d is the damping gain, x is the mass position and x_d is the desired position (input wrist position in the animation sequence). x_d is also called the "set point".

There are three basic response realms for a PD controller. If it is overdamped, it will fail to reach the set point. If it is critically damped, it will reach the set point as quickly as possible without overshoot. Underdamped will reach the set point more quickly, but oscillate around it before settling. For expressive control, we are generally most interested in underdamped motion as the nuance of the oscillations is what we wish to capture. Key character animation parameters are the number of oscillations, the frequency of the oscillations, the amount of overshoot and the overall smoothness of the motion. It is desirable to have more direct control of these parameters.

Equation 1 defines a prototypical second order system. In control theory, it is common to analyze the behavior of systems by looking at their response to a test function (e.g. [17]). Since this is a tracking controller, a ramp function $r(t) = vt$ is an appropriate test function, where v is a real constant that in this case corresponds to the velocity of the input. By the method of Laplace transforms, the response of the system due to a ramp function is:

$$x(t) = v\left[t - \frac{2\zeta}{\omega_n} + \frac{1}{\omega_n\sqrt{1-\zeta^2}}e^{-\zeta\omega_n t}\sin(\omega_n\sqrt{1-\zeta^2}t + \theta)\right] \tag{2}$$

where the *natural frequency* $\omega_n = \sqrt{\frac{k_s}{m}}$, the *damping ratio* $\zeta = \frac{k_d}{2m\omega_n}$ and $\theta = \cos^{-1}(2\zeta^2 - 1)$, $\zeta < 1$. $\zeta = 1$ provides critical damping (fastest rise time without overshoot). By subtracting out the input (vt), the error ϕ is given as:

$$\phi(t) = v\left[\frac{2\zeta}{\omega_n} + \frac{1}{\omega_n\sqrt{1-\zeta^2}}e^{-\zeta\omega_n t}\sin(\omega_n\sqrt{1-\zeta^2}t + \theta)\right] \tag{3}$$

A number of useful relations can be derived from this. The error will have peaks when $\frac{d\phi}{dt} = 0$. The times of the peaks are:

$$t_{peak} = \frac{n\pi - \frac{1}{2}\theta}{\omega_n\sqrt{1-\zeta^2}} \text{ for } n = 1, 2, \ldots, \tag{4}$$

so the peaks are a constant time apart for a particular set of parameters. The maximum overshoot occurs at the first peak and is given by:

$$\phi_{max} = v\left[\frac{2\zeta}{\omega_n} + \frac{1}{\omega_n\sqrt{1-\zeta^2}}e^{\frac{-\zeta}{\sqrt{1-\zeta^2}}(\pi - \frac{1}{2}\theta)}\sin(\pi + \frac{1}{2}\theta)\right] \tag{5}$$

It depends on the input velocity, ζ and ω_n, which in turn depends on the stiffness of the tracker. The maximum overshoot as function of ζ and stiffness is plotted in Figure 2.

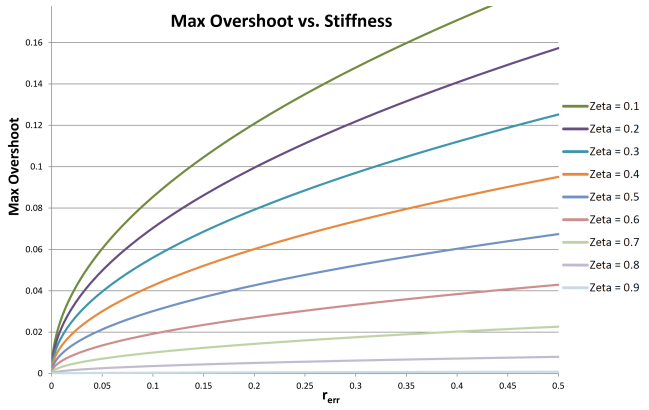

Figure 2: Maximum overshoot as a function of stiffness and zeta.

For animation, the number of oscillations in the error response has a strong visual impact, and this can be estimated for the ramp test function. The *settling time* is defined as the time at which the transient error response is within some threshold of the final value. If c_ts is the percent transient error at settling time (often 0.05 or 5%), the following relation envelopes the response:

$$\frac{-2v\zeta}{\omega_n} + \frac{v}{\omega_n\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t} = -(1+c_{ts})\frac{2v\zeta}{\omega_n} \tag{6}$$

This yields the settling time:

$$t_{settle} \approx \frac{1}{\zeta\omega_n} ln(2c_{ts}\omega_n\sqrt{1-\zeta^2}) \tag{7}$$

Equating t_{settle} and t_{peak} yields the following relation for the number of oscillations before settling, which depends solely on ζ:

$$n \approx \frac{-1}{\pi}\left[\frac{\sqrt{1-\zeta^2}}{\zeta} \ln\left(2c_{ts}\zeta\sqrt{1-\zeta^2} + \frac{1}{2}\theta\right)\right] \tag{8}$$

This is shown in Figure 3. In practice, we suggest $\zeta \geq 0.3$, with 0.2 or 0.15 sometimes being useful in high tension cases. We find that too much oscillation can look unnatural. See also Figure 5.

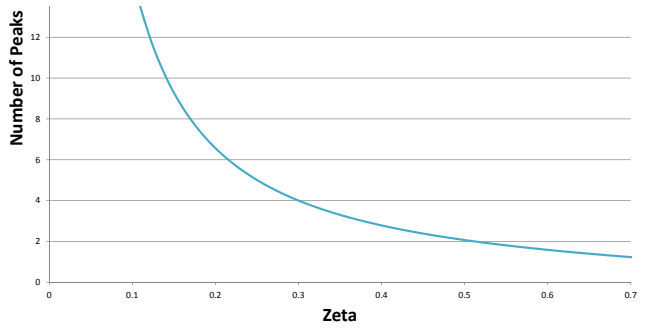

Figure 3: Approximate number of peaks (oscillations) vs. ζ.

Taken together, these relations give guidance on the features we have found most salient for animation: the amount of overshoot, the number of oscillations, the period of these oscillations and the lag, as summarized in Table 1. The steady state error is $\phi_{ss} = \frac{-2v\zeta}{\omega_n}$. If we assume a constant input velocity v, this yields $t_{lag} = \frac{-2\zeta}{\omega_n}$, which can be applied as an offset to the start time of the input to approximately maintain the desired timing.

When the input stops moving and the transient effects have dissipated, there will be a vertical rest error, r_{err}, due to gravity:

$$k_s(x_{yd} - x_y) = mg \implies r_{err} = x_{yd} - x_y = \frac{mg}{k_s} \tag{9}$$

This is also called the steadystate error. To avoid this error, we simply add r_{err} to the vertical component of the input desired position of the wrist. We also find it convenient and more intuitive to parameterize stiffness in terms of r_{err} instead of k_s. This gives a natural frequency $\omega_n = \sqrt{\frac{g}{r_{err}}}$. In practice, we find $r_{err} \in [30cm..1mm]$ yields a range from a very loose to a very stiff character.

To provide more intuitive control. we parameterize the mass tracker by (r_{err}, ζ)[1] instead of (k_s, k_d). The impact of changing stiffness (r_{err}) is illustrated in Figure 4. Notice the impact on both the amount of overshoot and the period of the oscillations. The impact of changing ζ is shown in Figure 5. This also impacts the amount of overshoot, but most significantly, controls the number of oscillations. Too many oscillations will look unnatural on a character, but a small number (often $0.3 \leq \zeta \leq 0.5$) can add useful nuance to an animation.

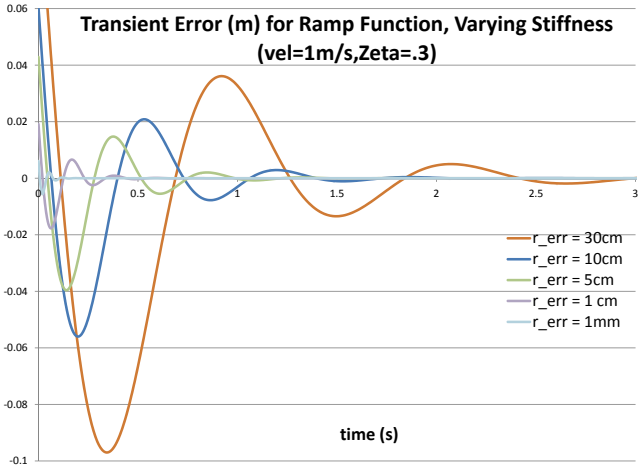

Figure 4: Tracking error as a function of variation in stiffness

4 MASS TRACKER IMPLEMENTATION

To implement mass trackers, the parameters of the mass and PD controller must be defined. We set $m = 0.4kg$ as a rough approximation of the weight of a hand. The gains of the PD controller are calculated from the user input (r_{err}, ζ). From Eq. 9,

$$k_s = \frac{mg}{r_{err}}. \tag{10}$$

The damping gain is simply ζ times the critical damping, or:

$$k_d = 2\zeta\sqrt{mk_s}. \tag{11}$$

[1]The letter ζ is sometimes written out as zeta in the paper.

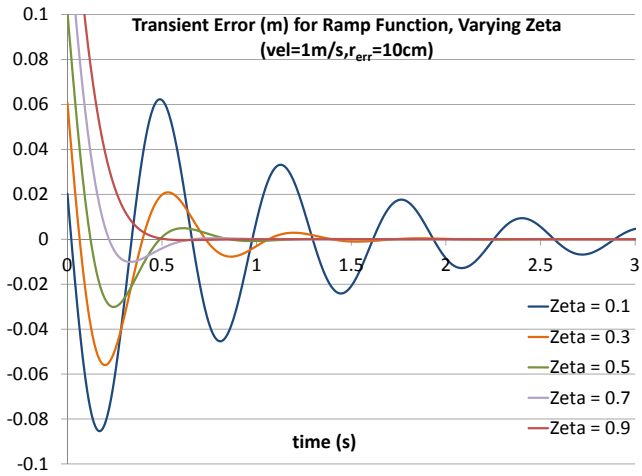

Figure 5: Transient error for different values of ζ.

Table 1: Parameter changes to impact the behavior of the tracking controller.

Desired Change	ζ	**S**tiffness	**I**nput Velocity
Increase Overshoot	decrease	decrease	increase
Increase Oscillation "Frequency"	decrease (limited)	increase	
Increase Number Oscillations	decrease	no effect	no effect

The wrist positions are calculated in the character's chest frame, which allows the gestures to naturally move with the character if body motion is edited. It is of course possible to use other frames, depending on the needs of the application. The vertical y set point has r_{err} added to it in order to compensate for the steady state error due to gravity. This ensures that the wrist position in the output animation is the same as that in the input when the wrist comes to rest and the mass has settled. Our prototype implementation employs Euler integration to update the mass trackers. This has worked fine in practice, but it would be possible to replace this with a more efficient integration technique if desired.

It is straightforward to integrate this technique into an existing animation pipeline. Animation systems output a new skeleton poses at each frame. The point masses should be initialized to the location of the wrists in the first frame of animation. Then the following steps should be added to the frame update:

(1) Take the existing output pose and express the wrist positions in the chest frame.
(2) Update point mass positions: Use the current wrist positions and specified controller parameters as input and forward integrate the acceleration from Eq. 1 to obtain the new point mass position (add the force from gravity, mg, to Eq. 1).
(3) Use IK to position the wrists at the locations of the masses.
(4) Use this new pose as the output pose of the animation system.

We found a simple IK chain that goes from the shoulder to the wrist is sufficient and set the swivel angle to match the input data. There are freely available IK routines (e.g. [38]).

5 RESULTS

The accompanying video (https://youtu.be/6WbdI9ZPHjY) contains numerous examples of gesture sequences created using the techniques described above, demonstrating both loose and tense motion (e.g.Fig. 1). In order to generate a gesture animation, it is necessary to first have an initial specification of the trajectory of the wrists. This will in turn be refined by the mass trackers to produce the final animation. This trajectory can be obtained either through keyframing, using motion capture data or taking the output of a machine learning algorithm. For the demonstrations shown here, we selected two motion capture clips from a publicly available dataset[2] [15] in order to provide the base motion.

When using keyframes as the motion representation, fewer keys can be used as simulation can fill in some of the dynamic subtleties of the motion. Motion capture presents a different challenge as the dynamics of the original motion are baked into the recorded data. It may be obvious that a motion can be made more loose – reducing the stiffness of the tracker will cause it to track the original motion less closely, leading to smoother and more free looking motion – but it is less obvious that a motion can be made to look more stiff as a stiff tracker will try to closely replicate the original trajectory. To see what is possible, it is necessary to understand a little of the grammar of gesture. Gestures consist of *phases* [22]. The *stroke* is the main meaning carrying portion of the gesture and generally the portion that we wish to most maintain in the final animation due to its relation to semantics. On either side of strokes, there may be *holds*, moments of stillness. *Retractions* bring the hands to rest and *preparations* connect between strokes or rest poses and strokes.

If an originally loose motion is recorded, the loose dynamics are baked into the data, including oscillations. If a stiff tracker reproduces these oscillations, the motion can look relaxed even though the simulated tension is high. However, the soft oscillatory behavior associated with such loose dynamics is largely contained in the hold phase following a stroke. If this short portion of the motion is simply replaced with a constant hold – the hand is held at the end of the stroke, something that does occur in real gesture – the dynamics of the mass tracker can then add subtle variation in this portion of the motion. This could be high frequency vibration to create a more stiff motion or a loose continuation and oscillation to create more relaxed motion. These small dynamic nuances communicate important expressive information about the state of the character. Additional edits can also be layered on top of tension changes, such as changing stroke speed which will increase overshoot.

In many gesture applications, the phase information will be present (e.g. it is included in the Behavior Markup Language [40]). It can be annotated if not present. This is a relatively small amount of work compared to say trying to hand animate physically plausible oscillations, a task we have found challenging. In applications where it is not desirable to provide this annotation, the approach can still be used. In these cases, it will generally still perform well for low tension settings which tend to soften any oscillations in the input

[2] https://trinityspeechgesture.scss.tcd.ie/

motion. For increased tension, it may not perform as well since the tracking can replicate oscillations from lower tension input motion and the stop in the motion provided by the static hold is helpful for producing high tension oscillations.

Many of the examples in the high tension animations use a very small ζ, as low as 0.15, for illustrative purposes to make the edit more easily noticed. In practice, such high oscillation is most effective if reserved for extreme situations where there is a moment of great tension, such as in heightened anger. The most effective animations will fluctuate between different levels of tension and relaxation to create texture and reflect changing mood and emphasis. As illustrated in the video, any level of oscillation is possible by adjusting zeta, including completely removing the oscillation.

For gesture, the most effortful portion of the motion is the stroke phase [23]. In practice, we therefore generally use higher stiffness for this phase and lower stiffness for retractions. During retractions, people are generally relaxing as they move to a rest pose and added pendular swing of the arms as they come to a character's side is often visually desirable. A good starting point for a fairly stiff character is to use steady state error and zeta values of (5mm, .3) for the stroke to provide high frequency vibrations and (1cm, .4) for controlled retractions. For a more loose character, (10cm, .5) can be used for strokes with limited oscillation and (15cm, .4) for loose retractions with more arm swing. Again, the richest animation will vary tension for different gestures throughout a sequence.

An added benefit of mass trackers is that they can inherit motion from the body since they are not rigidly attached to it. Using loose tension for relaxed characters can add pleasing natural arm swing when a character turns or shifts weight.

6 EVALUATION

Two different evaluations were conducted. The first establishes that the changes enacted by the mass trackers are indeed perceived as changes in tension. The second explores the parameter space and naturalness of the resulting motion to provide guidance on using mass trackers in practical character work.

6.1 Validation of Perceived Tension Changes

An initial study aimed to establish that the changes enacted by the mass trackers are perceived as changes in tension, as desired.

6.1.1 Stimulus. Five short gestural animations were selected from the clips in the video (Sec. 5). In all cases, the original input motion was optical motion capture of a single speaker. All clips were between 8 and 13 seconds and contained several gestures. New versions of each clip were generated using high and low tension settings. Similar settings were used for all clips. For high tension in clip one, error and zeta values of (.3cm, .15) were used and (.5cm, .2) were used for the rest of the high tension clips. Clip two used low tension settings of (15cm,.5) and the remaining clips used (30cm, .4). These ten clips were used in the study and each included the original audio.

6.1.2 Method. An online survey was created using Amazon Mechanical Turk. Participants rated each of the ten animations for how much tension they perceive in the motion on an 11-point scale,

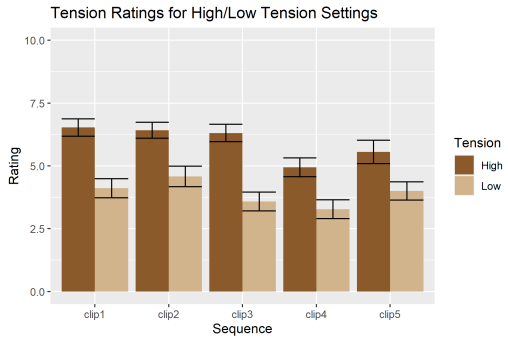

Figure 6: Tension ratings for each clip generated with high and low tension settings.

where 0 corresponds to "Very Loose" and 10 "Very Tense". Participants were shown an instruction screen, an example of a range of animations and then each of the ten animations, one-by-one. Animations played once, full screen, in random order and could not be scrubbed. The answer entry appeared after each video completed.

A total of 36 participants took part in the study. They were required to be "Master" workers (a Mechanical Turk designation based on past work), have an approval rating above 97% on previous work, live in the United States and use a desktop computer. Mean age was 41.9 (SD 9.9, max 71, min 24). Participants were paid two USD and the study had a median duration of 5.2 minutes.

6.1.3 Results. Survey results are shown in Figure 6. For every clip, there is a clear difference in the perceived tension between the version synthesized with high tension parameters and the one with low.

Statistical analysis was performed by fitting a linear mixed-effects model to the data using the lmer() function in R (similar to an ANOVA) [7, 8], with tension ratings as the dependent variable, Tension settings and Sequence as the fixed effects and participant ID as a random effect. The significance of main effects and interactions was calculated with Wald tests (Anova). Post-hoc tests were performed using estimated marginal means ([34, 35]) to conduct pairwise comparisons using the Tukey method for correction.

Both the tension settings ($\chi^2(1) = 93.27$, $p < 2.2e-16$) and motion clip ($\chi^2(4) = 21.01$, $p = 0.00032$) have significant main effects on the ratings, but there is no significant interaction ($\chi^2(4) = 4.59$, $p = 0.33$). The low tension settings produced motion that was seen as significantly more loose than the high tension settings for every clip. This confirms that the model supports effective modification of perceived motion tension.

6.2 Exploration of Parameter Space

The second study was designed to understand the impact of parameter choice on achieving particular tension effects, in particular, the impact of zeta. This was guided by several hypotheses based on experience with the approach:

- H1: The perception of increased tension relies on oscillation, so will only be present for small zeta.
- H2: Very low zeta may appear less natural with high tension.

- H3: Too much oscillation in loose motion can appear unnatural, so higher zeta is preferred for loose motion.
- H4: The technique is not overly sensitive to parameter selection, and in particular, some range of steady state errors will be acceptable for a desired effect.

6.2.1 Stimuli. Given the large set of parameter levels for each clip, it is necessary to keep the number of clips small in order to not fatigue participants with the total combinations. We chose two base clips. The first was clip three in the initial study above. The second was clip two from the first study, but in order to more fully explore the gesture space, the velocity of every gesture in this clip was doubled. This created a higher tempo clip that would also react differently to tracking, providing two different base points.

For high tension, a base error of 5mm was selected and for low tension, 15cm. To explore the impact of zeta, each of these clips was generated with zeta values of .15, .2, .3, .45 and .6. To explore the impact of error changes, two additional error levels were generated at .3 zeta for low and high tension: 3mm and 1cm for high tension and 30 and 50cm for low tension. In addition, the kinematic (no mass trackers) input clips were included. This produced a total of 30 clips: 2 base motions x 2 tension levels (high and low) x (5 zeta levels + 2 additional error levels) + 2 kinematic clips.

6.2.2 Method. The study was deployed on Mechanical Turk, following the same general method as Experiment 1 (Sec. 6.1.2). After each clip, participants rated the perceived tension as before, and also their agreement to the statement "The motion in the clip appears natural." on an eleven point scale from 0 (Strongly Disagree) to 10 (Strongly Agree). Since there could be a relationship between the audio and the tension of the gesture which might bias naturalness ratings, the audio was not included in this experiment. Again, 36 people partook in the experiment, with mean age 44.0 (SD 10.6, min 29, max 69). They had to meet the same qualifications as study 1. They were paid 5 USD for the study with a median duration of 12.9 minutes.

We felt it was important for each participant to see the full range of high and low tension clips to avoid any tendency to artificially inflate the range of ratings as might occur if they only saw high tension clips, for instance. There is no reason, though, to expect the parameters to have the same effect at high and low tension nor across the two base clips, so we used a set of eight planned comparisons in order to test the hypotheses: all zeta levels + kinematic for each tension level and each clip (4 groups of 6 stimuli) and the three different error levels for zeta=.3 + kinematic, again for each tension level and clip (4 groups of 4 stimuli). These comparisons were done for both tension and naturalness ratings.

6.2.3 Results and Discussion. As in Study 1, a linear mixed effects model was fit to the data for every planned comparison. The statistical results are listed in Table 2. Figure 7 shows the ratings for high tension motion at different levels of zeta. For the fast clip, tension is perceived as significantly higher at zeta = .15 (p=0.0044) and zeta =.2 (p=0.0078). For the normal speed clip, the zeta=.15 (p=.0010) is seen as significantly more tense. This provides evidence confirming H1, the additional oscillation provided by low zeta appears important for increasing the perception of tension. It can also be observed that the overall tension level appears higher in the fast clip. This could be related to people correlating speed and tension, the greater overshoot or perhaps to the particular gesture forms.

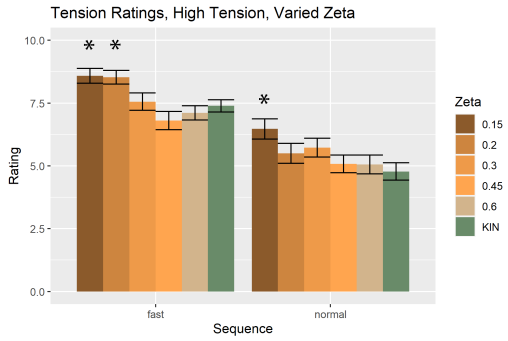

Figure 7: Tension ratings for high tension (5mm error) with varied zeta. Significant differences from the kinematic baseline are marked with a *.

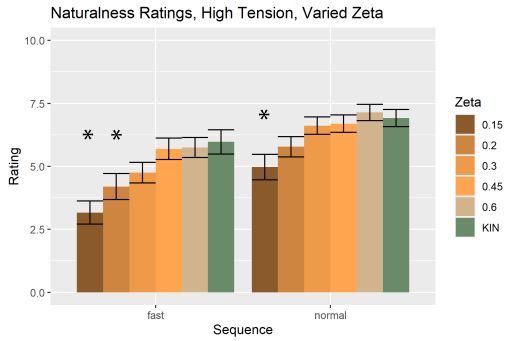

Figure 8: Naturalness ratings for high tension (5mm error) with varied zeta. Significant differences from the kinematic baseline are marked with a *.

Figure 8 shows the naturalness ratings for the high tension clips as zeta is varied. There is a significant decline in naturalness for zeta of .15 (p<.0001) and .2 (p=0.0011) for the fast clip and zeta of .15 (<.0001) for the normal clip, the same levels that lead to a significant increase in perceived tension. This drop was anticipated at the .15 level, but not at .2, partially confirming Hypothesis 2. It is worth remembering that people are judging these clips with no context and this type of tension increase would correspond to fairly extreme emotions, e.g. "shaking with anger", so what might seem unnatural without context might be believable with the correct context cues. This should be confirmed and people may want to reserve this heightened tension for high impact situations.

Tension ratings for the low tension parameters and varied zeta are shown in Figure 9. For both the fast and the normal clip, all the zeta levels produce motion that is perceived as significantly looser than the kinematic motion. In addition, there were no significant differences between the different levels of zeta.

Naturalness ratings for the low tension parameters are shown in Figure 10. There were no significance differences for the fast motion, but zeta of 0.15 and 0.2 were rated less natural for the normal speed

Tension	Rating	Parameter	Fast Clip	Normal Clip
high	tension	zeta	$\chi^2(5) = 52.6, p < .0001$	$\chi^2(5) = 22.05, p = .0005$
high	naturalness	zeta	$\chi^2(5) = 64.0, p < .0001$	$\chi^2(5) = 43.6, p < .0001$
low	tension	zeta	$\chi^2(5) = 83.7, p < .0001$	$\chi^2(5) = 23.8, p = 0002$
low	naturalness	zeta	$\chi^2(5) = 18.5, p = .0024$	$\chi^2(5) = 86.1, p < .0001$
high	tension	error	$\chi^2(3) = 3.1, p = .38$	$\chi^2(3) = 10.6, p < .014$
high	naturalness	error	$\chi^2(3) = 13.3, p = .0040$	$\chi^2(3) = 9.19, p < .027$
low	tension	error	$\chi^2(3) = 80.7, p < .0001$	$\chi^2(3) = 28.8, p < .0001$
low	naturalness	error	$\chi^2(3) = 0.164, p = .98$	$\chi^2(3) = 9.29, p < .026$

Table 2: Significance of main effects, Exp. 2.

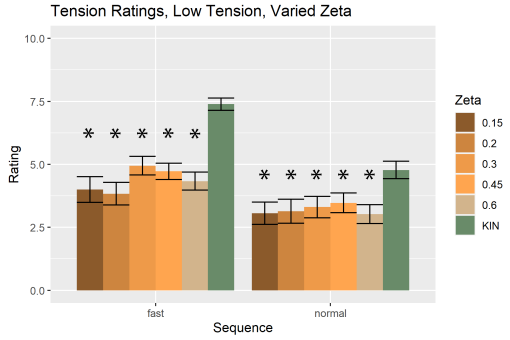

Figure 9: Tension ratings for low tension (15cm error) with varied zeta. Significant differences from the kinematic baseline are marked with a *.

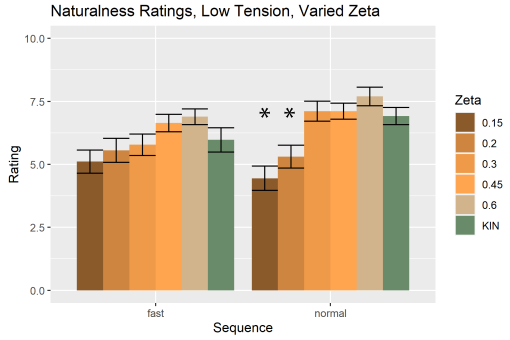

Figure 10: Naturalness ratings for high tension (15cm error) with varied zeta. Significant differences from the kinematic baseline are marked with a *.

motion. This provides partial support for Hypothesis 3, confirming it for the normal clip, but disconfirming it for the fast clip. Unlike with high tension, the low tension effect can be achieved across the range of tested zeta values. There is therefore no reason to use the very low zeta values for low tension and it is recommended to stay with zeta above .3 for this application (.4 and .5 were used in the low tension example clips in Sec. 5).

The error (stiffness) variants were decided a priori, and the selected zeta of .3 is above what was found in the zeta comparisons to trigger an impression of high stiffness. It is not surprising, therefore, that most clips were not seen as significantly more tense than kinematic. Perceived tension for both clips increased as the error increased (from 3mm, to 5mm, to 1cm), although only the 1cm normal clip was seen as significantly more tense than kinematic (mean perceived tension of 5.89 for 1cm compared to 4.78 for kinematic, p=.0010). This illustrates another method of increasing perceived tension: lowering stiffness, while still keeping it relatively high, will increase the size of the overshoot, making the oscillations more visually salient. There were no significant differences in naturalness between the stiffness levels.

For the low stiffness error variations, all error levels (15, 30 and 50cm) appeared significantly looser than the kinematic motion. There were no significant differences in perceived tension for the tracked variants (there was a tendency for 15cm to be perceived as more tense than 30cm (p=0.072) and 50cm (p=0.054)). For no clip did the naturalness level vary significantly from the kinematic clip, although for the normal base motion, 50cm error was perceived as significantly less natural than 15cm error (7.11 to 5.86, p=0.031). This is a very loose controller, so it is surprising that it performed this well.

For low tension in particular, similar perceptual results can be obtained for a range of parameters, which offers partial support for H4, that the approach is not overly sensitive to parameter choice. High tension is more sensitive, however. Two constellations have been identified that increase perceived tension: high stiffness with very low zeta (e.g. 5mm and .15) or slightly lower stiffness with higher zeta (e.g. 1cm and .3). The first produces higher frequency oscillations and the second larger oscillations. Other constellations may be possible, but it appears clear visual manifestation of oscillations are key in increasing the perceived tension.

It should be noted that all the test scenarios focused on tense or loose motion. We anticipate that the intermediate range of tension levels can be invoked by using intermediate parameters (e.g. errors between 1 and 15cm).

7 DISCUSSION AND CONCLUSION

Mass trackers provide agent designers with an effective additional control to edit the style of gesture animation. The derivation presented here provides clear guidance on how to tune the controllers to achieve desired effects. The impact of both r_{err} and ζ are much easier to understand than k_s and k_d. This dramatically reduces the amount of tune and test iteration required with many previous simulation approaches. The studies provide further guidance on appropriate parameters to use in practice. In addition, mass trackers are computationally light weight and can be used to smooth motion if needed. The approach does not require implementing

a full skeletal simulation or determining moments of inertia for the model, etc. There is no need to solve the balance problem. The only added computation is to update two point masses and solve a simple IK problem.

The technique has limitations. It is only a partial model, focusing on control of the position of the wrists. While effective in practice, there are important cases that this does not handle. For example, rapid radial rotation of the forearm will not be impacted by loosening tracking. In some cases, it might also be useful to add dynamic oscillations of the wrist for floppy movements. While the technique has been used in practice for several years (e.g. [10, 37]), only a small number of examples clips were tested in the studies discussed here. Finally, while the torso tends to exhibit little oscillatory movement during gesture, a model that reflected the impact of tension on posture would be a useful addition.

ACKNOWLEDGMENTS

Thanks to Harrison Jesse Smith and Gabriel Castillo for using this technique in their work, Ylva Ferstl for providing the motion capture data and the anonymous reviewers for strengthening the work. Financial support from the National Science Foundation on grant IIS 2232066 is gratefully acknowledged.

REFERENCES

[1] Chaitanya Ahuja, Dong Won Lee, Yukiko I. Nakano, and Louis-Philippe Morency. 2020. Style Transfer for Co-Speech Gesture Animation: A Multi-Speaker Conditional-Mixture Approach. https://arxiv.org/abs/2007.12553

[2] Simon Alexanderson, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. 2020. Style-Controllable Speech-Driven Gesture Synthesis Using Normalising Flows. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 487–496.

[3] Brian Allen, Derek Chu, Ari Shapiro, and Petros Faloutsos. 2007. On the beat!: timing and tension for dynamic characters. In *Symposium on Computer Animation*. San Diego, CA, USA, 239–247.

[4] B. Allen, M. Neff, and P. Faloutsos. 2010. Pose control in dynamic conditions. *Motion in Games* (2010), 48–58.

[5] B.F. Allen, M. Neff, and P. Faloutsos. 2011. Analytic proportional-derivative control for precise and compliant motion. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 6039–6044.

[6] Eugenio Barba. 1991. Dilated Body. In *A Dictionary of Theatre Anthropology: The Secret Art of The Performer*, Eugenio Barba and Nicola Savarese (Eds.). Routledge, London.

[7] Douglas Bates et al. 2005. Fitting linear mixed models in R. *R news* 5, 1 (2005), 27–30.

[8] Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. 2014. Fitting linear mixed-effects models using lme4. *arXiv preprint arXiv:1406.5823* (2014).

[9] Justine Cassell, Catherine Pelachaud, Norman Badler, Mark Steedman, Brett Achorn, Tripp Bechet, Brett Douville, Scott Prevost, and Matthew Stone. 1994. Animated Conversation: Rule-Based Generation of Facial Expression Gesture and Spoken Intonation for Multiple Conversational Agents. *Proceedings of SIGGRAPH 94* (1994), 413–420.

[10] Gabriel Castillo and Michael Neff. 2019. What do we express without knowing? Emotion in Gesture. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. 702–710.

[11] Diane M. Chi, Monica Costa, Liwei Zhao, and Norman I. Badler. 2000. The EMOTE Model for Effort and Shape. In *Proc. SIGGRAPH 2000*. 173–182.

[12] Jean Dorcy. 1961. *The Mime*. Robert Speller and Sons, Publishers, Inc. Translated by Robert Speeler, Jr. and Pierre de Fontnouvelle.

[13] Funda Durupinar, Mubbasir Kapadia, Susan Deutsch, Michael Neff, and Norman I Badler. 2016. Perform: Perceptual approach for adding ocean personality to human motion using laban movement analysis. *ACM Transactions on Graphics (TOG)* 36, 1 (2016), 1–16.

[14] Ylva Ferstl, Michael Neff, and Rachel McDonnell. 2020. Adversarial gesture generation with realistic gesture phasing. *Computers & Graphics* (2020).

[15] Ylva Ferstl, Michael Neff, and Rachel McDonnell. 2021. ExpressGesture: Expressive gesture generation from speech through database matching. *Computer Animation and Virtual Worlds* (2021), e2016.

[16] Shiry Ginosar, Amir Bar, Gefen Kohavi, Caroline Chan, Andrew Owens, and Jitendra Malik. 2019. Learning individual styles of conversational gesture. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3497–3506.

[17] Farid Golnaraghi and Benjamin C. Kuo. 2010. *Automatic Control Systems*. John Wiley & Sons.

[18] B. Hartmann, M. Mancini, and C. Pelachaud. 2002. Formational parameters and adaptive prototype installation for MPEG-4 compliant gesture synthesis. In *Proc. Computer Animation 2002*. 111–119.

[19] B. Hartmann, M. Mancini, and C. Pelachaud. 2006. Implementing expressive gesture synthesis for embodied conversational agents. In *Proc. Gesture Workshop 2005 (LNAI, Vol. 3881)*. Springer, Berlin; Heidelberg, 45–55.

[20] Jessica K. Hodgins, Wayne L. Wooten, David C. Brogan, and James F. O'Brien. 1995. Animating Human Athletics. In *Proc. SIGGRAPH 1995*. 71–78.

[21] Michael Kass and John Anderson. 2008. Animating oscillatory motion with overlap: wiggly splines. In *ACM SIGGRAPH 2008 papers*. 1–8.

[22] Adam Kendon. 1972. Some relationships between body motion and speech. *Studies in dyadic communication* 7, 177 (1972), 90.

[23] S. KITA, I. VAN GIJN, and H. VAN DER HULST. 1998. Movement phase in signs and co-speech gestures, and their transcriptions by human coders. In *Proceedings of the International Gesture Workshop on Gesture and Sign Language in Human-Computer Interaction*. Springer-Verlag, 23–35.

[24] Doris H. U. Kochanek and Richard H. Bartels. 1984. Interpolating Splines with Local Tension, Continuity, and Bias Control. *Computer Graphics (Proceedings of SIGGRAPH 84)* 18, 3 (1984), 33–41.

[25] Taras Kucherenko, Patrik Jonell, Sanne van Waveren, Gustav Eje Henter, Simon Alexandersson, Iolanda Leite, and Hedvig Kjellström. 2020. Gesticulator: A framework for semantically-aware speech-driven gesture generation. In *Proceedings of the 2020 International Conference on Multimodal Interaction*. 242–250.

[26] Rudolf Laban. 1988. *The Mastery of Movement* (fourth ed.). Northcote House, London. Revised by Lisa Ullman.

[27] Joan Lawson. 1957. *Mime: The Theory and Practice of Expressive Gesture With a Description of its Historical Development*. Sir Isaac Pitma and Sons Ltd., London. Drawings by Peter Revitt.

[28] Seunghwan Lee, Moonseok Park, Kyoungmin Lee, and Jehee Lee. 2019. Scalable muscle-actuated human simulation and control. *ACM Transactions On Graphics (TOG)* 38, 4 (2019), 1–13.

[29] Sung-Hee Lee, Eftychios Sifakis, and Demetri Terzopoulos. 2009. Comprehensive biomechanical modeling and simulation of the upper body. *ACM Transactions on Graphics (TOG)* 28, 4 (2009), 1–17.

[30] Yoonsang Lee, Moon Seok Park, Taesoo Kwon, and Jehee Lee. 2014. Locomotion control for many-muscle humanoids. *ACM Transactions on Graphics (TOG)* 33, 6 (2014), 1–11.

[31] C. Karen Liu, Aaron Hertzmann, and Zoran Popović. 2005. Learning physics-based motion style with nonlinear inverse optimization. *ACM Transactions on Graphics* 24, 3 (Aug. 2005), 1071–1081.

[32] Michael Neff and Eugene Fiume. 2002. Modeling Tension and Relaxation for Computer Animation. In *Proc. ACM SIGGRAPH Symposium on Computer Animation 2002*. 81–88.

[33] Michael Neff and Eugene Fiume. 2005. AER: Aesthetic Exploration and Refinement for Expressive Character Animation. In *Proc. ACM SIGGRAPH / Eurographics Symposium on Computer Animation 2005*. 161–170.

[34] L Russell, S Henrik, L Jonathon, B Paul, and H Maxime. 2018. Estimated marginal means, aka least-squares means. *The American Statistician* 34 (2018), 216–221.

[35] Shayle R Searle, F Michael Speed, and George A Milliken. 1980. Population marginal means in the linear model: an alternative to least squares means. *The American Statistician* 34, 4 (1980), 216–221.

[36] Ted Shawn. 1963. *Every Little Movement: A Book about Francois Delsarte* (second revised ed.). Dance Horizons, Inc., New York.

[37] Harrison Jesse Smith and Michael Neff. 2017. Understanding the Impact of Animated Gesture Performance on Personality Perceptions. *ACM Trans. Graph.* 36, 4, Article 49 (July 2017), 12 pages. https://doi.org/10.1145/3072959.3073697

[38] D. Tolani, A. Goswami, and N.I. Badler. 2000. Real-time inverse kinematics techniques for anthropomorphic limbs. *Graphical models* 62, 5 (2000), 353–388.

[39] Herwin van Welbergen, Dennis Reidsma, Zsófia M Ruttkay, and Job Zwiers. 2009. Elckerlyc. *Journal on Multimodal User Interfaces* 3, 4 (2009), 271–284.

[40] H. Vilhjalmsson, N. Cantelmo, J. Cassell, N. E. Chafai, M. Kipp, S. Kopp, M. Mancini, S. Marsella, A. Marshall, C. Pelachaud, et al. 2007. The behavior markup language: Recent developments and challenges. In *Intelligent Virtual Agents*. Springer, 99–111.

[41] Victor B. Zordan and Jessica K. Hodgins. 1999. Tracking and Modifying Upper-Body Human Motion Data with Dynamic Simulation. *Computer Animation and Simulation '99* (1999).

[42] Victor B. Zordan and Jessica K. Hodgins. 2002. Motion Capture-Driven Simulations that Hit and React. In *Proc. ACM SIGGRAPH Symposium on Computer Animation 2002*. 89–96.

[43] Victor Brian Zordan, Anna Majkowska, Bill Chiu, and Matthew Fast. 2005. Dynamic response for motion capture animation. *ACM Transactions on Graphics* 24, 3 (Aug. 2005), 697–701.