Calculating Cut Events via Dancer Displacement

Owen Lu
Electroimpact Inc.
owenl@electroimpact.com

***Abstract*—The timing of material addition and cut events are central to the mechanical dancer displacement method to determine the placement of carbon fiber tow ends. Due to the consistent performance of the tensioning system, a characteristic signal in dancer displacement can be used to estimate the timing of a material cut event using high-speed encoder processing. The method developed improves upon old methods by combining control law equations with curve fitting to produce an estimate of the cut timing event.**

I. Introduction

The key idea of this algorithm is to use the form of the simplified control law equations to estimate the cut events. Notably, using simpler methods such as velocity thresholds has been shown to be difficult in the past due to the fact that the dancer velocity curves transition smoothly, thus using threshold techniques will systematically delay cut event time estimations. The problem of higher-order oscillation is also an important problem that adds error to the estimations.

The algorithm proposed uses the control law to estimate the tow velocity as it exits the tensioning system. The control law allows prediction of velocity purely based on control constants and the dancer displacement, velocity and acceleration. The estimate is then used to perform a curve fit assuming the characteristics of a cut event. It has been proposed that the cut event can be detected and modeled by dancer velocity dropping with a filtered ramp characteristic.

II. Velocity Approximation via Control Law

On the OH20 machine, a single control law is used to generate velocities according to dancer displacement.

$$C = \frac{K_e}{\tau s + 1} \tag{1}$$

Assuming that $\tau = \frac{1}{2K_e}$ the relationship between dancer displacement and tow velocity can be estimated with the equation below.

$$\frac{x_d}{v} = \frac{\left(\frac{1}{2}s + K_e\right)}{s + K_e{}^2} \tag{2}$$

Converting to discrete-time to estimate v, we get the below equations.

$$a_k = 2(x_k'' + 2K_e x_k' + K_e^2 x_k - K_e v_k) \tag{3}$$

$$v_{k+1} = v_k + a_k \Delta t \tag{4}$$

III. Ramp Velocity Model

Using the estimate of the tow velocity we can observe the point where there is a rapid change in the estimated velocity. It is near this point where the cut event has happened.

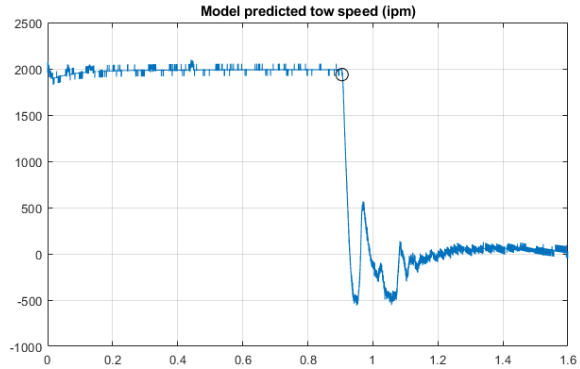


Figure 1 – Velocity estimation using control law

Zooming closer the cut point it is clear that the tow velocity has ramp characteristics but lacks the sharp discontinuous derivatives that theoretical ramp will have. This is attributed to the clamping action on the tow end before a cut action. Thus, an equation for a filtered ramp is formulated and fit to the graph. Notice, that the full-time scale is roughly 30ms showing that the process of tow deceleration happens very quickly.

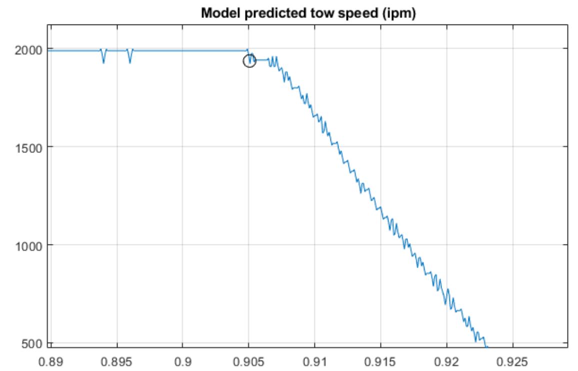


Figure 2 – Plot of the tow speed near the estimated cut event; time of cut event shown by the black circle

The equation describing the behavior near the cut event in the time domain is given below. Empirically, this is approximately valid for 40ms centered around the cut event.

$$f\ t\ = v_0 + a_0 \left(\tau \left(e^{-\frac{t-t_0}{\tau}} - 1 \right) +\ t - t_0\ \right) \cdot u\ t - t_0 \tag{5}$$

Where
$u\ t$ is the step function
t_0 is the time of the cut event
v_0 is the tow velocity during the cut
a_0 is the tow cut acceleration
τ is the filter constant

IV. ALGORITHM

The problem of finding the cut event time is to curve fit the signal near the cut event point to compute t_0. Note that all values (t_0, v_0, a_0, τ) are simultaneously solved. Since estimations of initial constants to be fed into Gauss-Newton (GN) methods are known to a relatively high degree of accuracy, only a few iterations are used. Furthermore, the small window of only 40ms means that N, the number of points, is less than 100 which can be solved extremely quickly.

The general definitions and method to use the GN

$$S\ \beta\ = \sum_{i=1}^{N} r_i^2 \tag{6}$$

$$\beta_0 = [t_0 \quad v_0 \quad a_0 \quad \tau]^T \tag{7}$$

$$r_i = v_i - f\ t_i, \beta \tag{8}$$

$$J_{ij} = \frac{\partial r_i}{\partial \beta_j} \tag{9}$$

$$\beta_{k+1}\ = \beta_k -\ \boldsymbol{J}^T \boldsymbol{J}\ ^{-1} \boldsymbol{J}^T \boldsymbol{r}\ \beta_k \tag{10}$$

Where
$S(\beta)$ is the sum of squared residuals as a function of curve fit parameters β
$\boldsymbol{r}$ is the $N \times 1$ column vector of residuals
$\boldsymbol{J}$ is the $N \times 4$ matrix of residual derivatives

Empirically, 3 iterations is sufficient to gain good estimates of t_0.

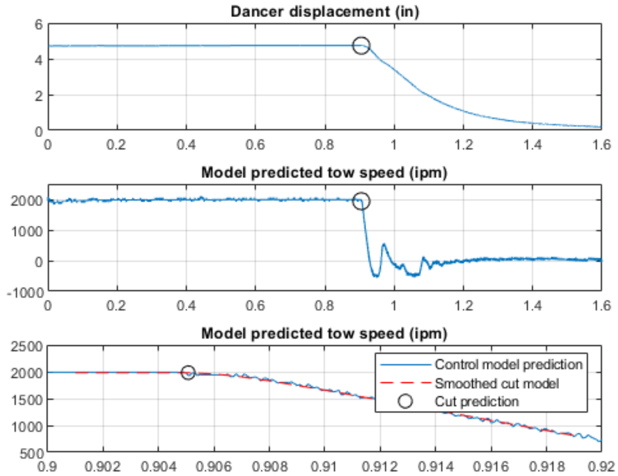
V. PROCEDURE

Using the algorithm described can be implemented using the following procedure.

1. For all dancer lanes
2. Initialize tow velocity
3. Calculate the acceleration and update velocity using Eq. 3 and Eq. 4
4. On cut event trim the signal to approximately 20-30ms around the estimated cut event
5. Using initial estimate vector β_0 compute $f\ t$ in Eq. 5
6. Calculate all residuals and form vector $\boldsymbol{r}$
7. Compute the Jacobian $\boldsymbol{J}$
8. Update β estimate with Eq. 10
9. Repeat steps 6-8 with updated β values
10. Take t_0 as the cut event time

VI. RESULTS

On a simple test running at 2000ipm on a flat tool, the algorithm works very well identifying the cut event with agreeance with what a person might do by eye. A rough comparison would show that the difference is approximately less than 0.3ms. Currently, since using the vision system is the only way to verify the new RIPIT accuracy, it has yet to be shown that this algorithm works well in practice.



It should be noted that it is assumed that the dancer has reached its approximate steady-state displacement. Thus, this method is unlikely to work during minimum piece events. However, minimum, piece events are generally run slow, drastically reducing displacement uncertainty due to timing estimation errors.