The International Journal of Advanced Manufacturing Technology (2021) 116:417–438
https://doi.org/10.1007/s00170-021-07211-2

ORIGINAL ARTICLE

Accurate prediction of machining feedrate and cycle times considering interpolator dynamics

Rob Ward[1] · Burak Sencer[2] · Bryn Jones[3] · Erdem Ozturk[4]

Received: 3 February 2021 / Accepted: 3 May 2021 / Published online: 18 June 2021

Abstract

This paper presents an accurate machining feedrate prediction technique by modelling the trajectory generation behaviour of modern CNC machine tools. Typically, CAM systems simulate machines' motion based on the commanded feedrate and the path geometry. Such approach does not consider the feed planning and interpolation strategy of the machine's numerical control (NC) system. In this study, trajectory generation behaviour of the NC system is modelled and accurate cycle time prediction for complex machining toolpaths is realised. NC system's linear interpolation dynamics and commanded axis kinematic profiles are predicted by using finite impulse response (FIR)–based low-pass filters. The corner blending behaviour during non-stop interpolation of linear segments is modelled, and for the first time, the minimum cornering feedrate that satisfies both the tolerance and machining constraints has been calculated analytically for 3-axis toolpaths of any geometry. The proposed method is applied to 4 different case studies including complex machining toolpaths. Experimental validations show actual cycle times can be estimated with $>90\%$ accuracy, greatly outperforming CAM-based predictions. It is expected that the proposed approach will help improve the accuracy of virtual machining models and support businesses' decision-making when costing machining processes.

Keywords Interpolation · Feedrate prediction · FIR filters · Cycle time estimation · Milling

1 Introduction

With the introduction of concepts like virtual manufacturing [1] and digital twins [2], building process models and predicting actual machining process conditions in the computer environment has become paramount in attaining higher productivity and throughput in today's manufacturing. For example, accurate machining cycle time prediction is vital for industry during the quotation process to ensure achievable and profitable contracts. The prediction models and generation of accurate digital twins are a collective modeling effort which requires both detailed modelling of the process and the dynamic machine behaviour. Considering the machining processes, current literature provides accurate models to predict milling process physics [3–5]. Nevertheless, when applied in practice, these models show large discrepancies from the actual process behaviour.

One reason can be identified as the influence of the machine tool drive dynamics. In particular, the behaviour of the numerical control (NC) plays a key role. Trajectory generation (interpolation) algorithms embedded in the NC system control the feedrate profile, which is a key input for machining process models. For example, contouring (positioning) errors alter tool engagements [6] which lead to inaccurate force predictions [7]. Thus, in order to accurately develop realistic digital twins for machining processes, the feedrate profile generated by the NC system of a machine tool must be accurately predicted. This paper deals with modelling and prediction of interpolator dynamics of modern NC systems to accurately estimate machining cycle times and cutting forces along complex parts.

In this paper, the term "cycle time" is used to measure the "machining cycle time" which refers to the overall feed motion duration to travel along a machining part program. Once a part program (G-code) is deployed a CNC machine tool, the NC unit parses the part program and interpolates the tool motion between successive cutter

✉ Rob Ward
r.ward@amrc.co.uk

Extended author information available on the last page of the article.

locations (CL). Most modern CAM systems provide toolpaths in terms of discrete CL data and rely on linear interpolation algorithms that run in the NC units. With the introduction of cheap memory modules, long part programs do not pose a limit, and even basic circular paths are programmed with series of short linear segments [8, 9]. Therefore, modern NC systems are equipped with propriety algorithms that interpolate these lengthy series of short CL blocks smoothly. These algorithms are called *Look-ahead* or *Compressor* functions and are capable of generating a non-stop motion with a time optimal feedrate profile [10] that respects kinematic limits of the machine [12, 13]. Prediction of a machine's actual feedrate profile requires detailed modelling of the NC system's real-time interpolation behaviour. This includes the motion transition between CL blocks, for example a typical feedrate profile for continuous motion is shown in Fig. 1. During the initial linear motion from zero to commanded feedrate, the performance and behaviour of the machine tool is dependent upon the acceleration and jerk constraints alone. However, as the tool approaches the end of the first CL line (corner transition 1 in Fig. 1) to change the feed direction, the tool decelerates to a minimum cornering feedrate before accelerating again to the commanded feedrate. The reduction in feedrate in the vicinity of CL line junction point is due to both the machine tool satisfying the tool centre point (TCP) error tolerance constraints throughout the cornering transition and the machine tool kinematic constraints [19]. The TCP error can be seen at corner transition 2 where the TCP is at maximum displacement between the CL line and the TCP position. The TCP error constraint imposed upon the toolpath limits the maximum feedrate during cornering transitions and this significantly affects the overall machining cycle time.

Most NC systems utilise jerk-limited trajectory generation to smoothly alter feedrate and interpolate along CL lines [12, 14]. The generated feedrate profile is defined in the form of a cubic polynomial [8]. Axis acceleration limits are imposed based on the torque/power capacity of the drives, and the jerk limits are set to limit unwanted vibrations during rapid feed motion [15]. This general jerk-limited feedrate profile is well-known, and acceleration and jerk limits of the machine can be read from the NC system. Therefore, the use of jerk-limited trajectory as a template allows prediction of feedrate kinematics of modern NC systems and it can correctly predict point-to-point (P2P) trajectories. During P2P interpolation, the tool accelerates from a full stop to the set feedrate and decelerates again for a full stop at the end of the CL line. Once the acceleration and jerk limits are known, the feedrate profile can be generated

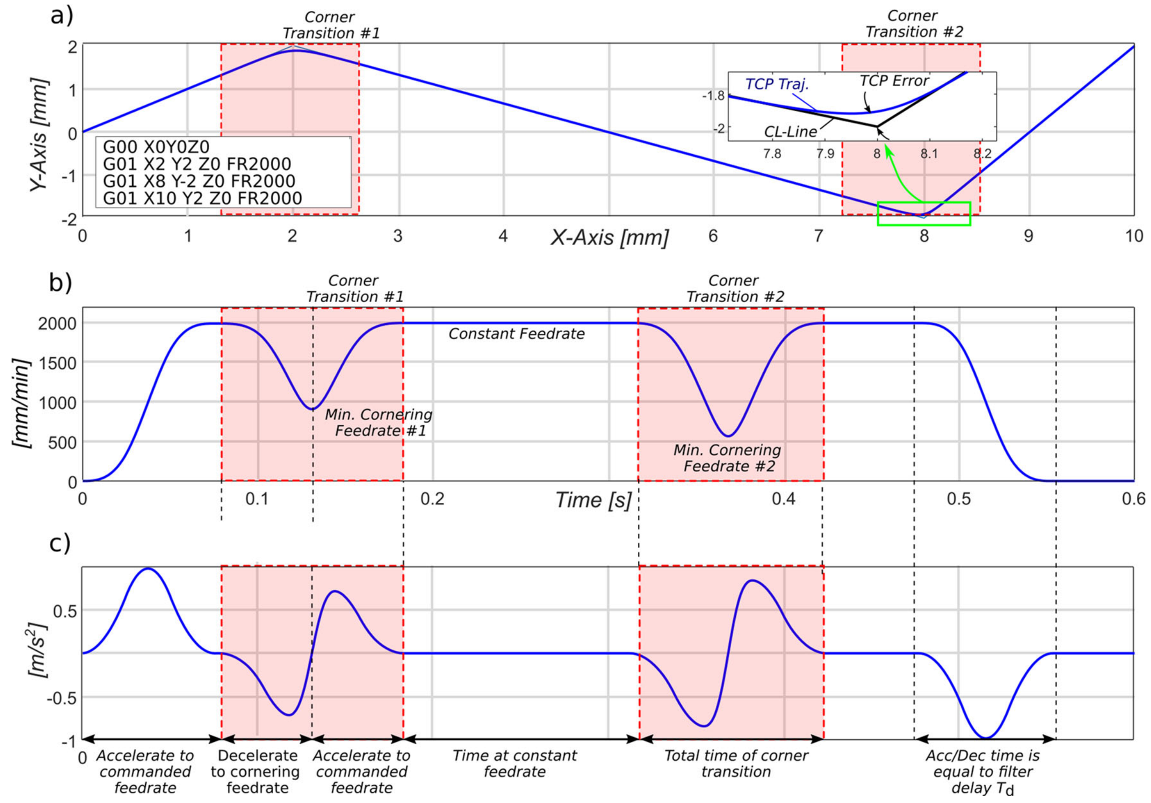


Fig. 1 Typical kinematic profiles of an NC program

to predict cycle times. Past research considered modelling of NC behaviour of 3 and 5 axis machine tools for both P2P and contouring paths [11, 16, 17].

Predicting feedrate profiles along short segmented complex toolpaths for high-speed machining (HSM) is a challenging task. This is due to the fact that look-ahead modules of NC systems alter jerk limits on the fly as it blends series of CL lines to generate a non-stop smooth continuous feed motion. Here, modelling the path blending behaviour is crucial. NC systems blend linear CL lines together smoothly whilst applying geometric blending error and kinematic limit control. Machine tool literature reports that circular arcs [8] and cubic [8] or quintic splines [12] can be used for such geometric path blending. There are also methods based on filtering where the discrete toolpath is blended based on low-pass filtering. Finite impulse response (FIR) filters are used for such purpose [18]. Such filtering-based techniques are more computationally efficient and greatly favoured for real-time interpolation on NC systems. For instance, Heidenhain [19, 20], Mitsubishi [21] and more recently Siemens [22] NC systems utilise FIR and IIR (infinite impulse response) filters for look-ahead and non-stop smooth interpolation. Typically, users enter a blending tolerance which confines the path blending (contour) errors. Based on the blending tolerance, the NC system approximates the given discrete CL lines and plans the fastest motion with its kinematic limits. Therefore, accurate prediction of cycle times for conventional toolpaths requires modelling of NC system's non-stop interpolation behaviour along linear paths.

This paper models the non-stop interpolation behaviour of modern NC systems and predicts feedrate profiles along HSM toolpaths by considering the real-time path blending behaviour of NC systems. Section 2 briefly introduces the low-pass filtering-based real-time interpolation method, which is used as a template. It is then used to predict P2P and contouring motion of NC systems in subsequent Section 3. Illustrative examples and experimental validations are provided in each section. Finally, Section 4 provides realistic cycle time, feedrate profile and cutting force prediction for complex aerospace parts.

2 Low-pass filtering-based real-time interpolator dynamics

This section models real-time interpolation behaviour of an NC system to predict the feedrate profile and overall machining cycle time. Most conventional NC systems utilise IIR or FIR filtering–based techniques for computationally efficient real-time interpolation and feed profile planning. In this work, finite impulse response (FIR) filters are used to capture the NC system's behaviour. A simple 1st-order FIR filter can be expressed in the Laplace (s) domain by:

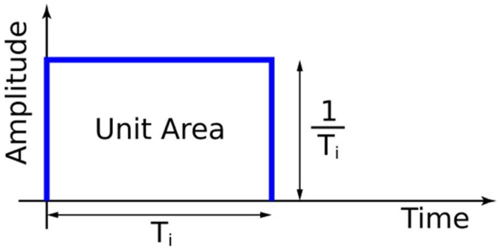


Fig. 2 Impulse response of a 1st-order FIR filter

$$M_i(s) = \frac{1}{T_i}\frac{1 - e^{-sT_i}}{s}, \quad i = 1 \ldots n, \tag{1}$$

where s is a complex number and T_i is the time constant of the i^{th} filter. The impulse response is depicted in Fig. 2. As seen in Eq. 1, the filter contains an integrator, which acts to smooth the input signal. These two features of 1st-order FIR filters are appealing from a NC system perspective, since G-codes (represented by rectangular velocity pulses) can be convolved through a series of such filters to generate smooth velocity profiles. Since the area underneath the rectangular impulse response is unitary, the area underneath the original input is not altered [18, 19, 21–24].

Figure 4 illustrates this filtering-based interpolation procedure. As shown, consider a G-code for a total displacement command of L at a feedrate of F. It is represented by a velocity pulse with an amplitude of F and duration of T_v; hence, $L = FT_v$. Subsequent convolution of the velocity pulse with the FIR filter yields the higher order velocity response. Using 2-FIR filters in series generates reference trajectories with piece-wise constant jerk profiles and using three FIR filters in series further smooths the reference velocity making them snap limited. Although jerk-limited trajectories are most common in high-speed machinery, snap-limited trajectories are tuned for ultra-precision machines [25] to further mitigate the effect of unwanted vibrations.

The duration of the original velocity pulse T_v and the time constants of the filters T_n determine the velocity and acceleration profiles, which can be derived analytically by evaluating the convolution integral between the input velocity pulse and the rectangular impulse response of the filter as follows:

$$\begin{aligned} v'(t) &= v(t) * m(t) \\ &= \frac{1}{T_1}\int_0^t \left([v(\tau) - v(\tau - T_v)]\,[u(t-\tau) - u(t - T_1 - \tau)]\right) d\tau \\ &= \frac{1}{T_1}\begin{bmatrix} \int_0^t v(\tau)u(t-\tau)d\tau - \int_0^t v(\tau)u(t - T_1 - \tau)\,d\tau \\ -\int_0^t v(\tau - T_v)\,u(t-\tau)d\tau + \int_0^t v(\tau - T_v)\,u(t - T_1 - \tau)\,d\tau \end{bmatrix} \end{aligned} \tag{2}$$

where $v(t)$, $v'(t)$ and $m(t)$ represent the velocity pulse, interpolated velocity signal and the impulse response of the FIR filter (Eq. 1) respectively. For multiple first-order FIR filtering with different time constants, a detailed analysis is given in [18]. This research introduces linear interpolation using FIR filters with identical time constants.

2.1 FIR interpolation with matching time constants

Typically, the time constants of FIR filters are selected to mitigate structural vibrations of the machine tool [27]. Matching the time constant with the vibration period of the lightly damped modes helps avoid exciting them during rapid acceleration. One method to specify the time constants is to set them equal, $T_1 = T_2$. In this special case, the FIR filter acts as a pure low-pass filter with a roll-over frequency of $\omega_c \approx \frac{2\pi}{T_1}$. Figure 3 shows the attenuation in the frequency response for multiple FIR filters with matching time constants. The time constant, when set low enough, helps prevent the excitation of any higher frequency vibrations during rapid accelerations. This simpler method compared to tuning individual filters provides a convenient method of vibration suppression during high feedrates.

For interpolation using 2-FIR filters with matching time constants, the transfer function of the resulting FIR filter is:

$$M_{2FIR}(s) = \left(\frac{1}{T_1}\frac{1-e^{-sT_1}}{s}\right)\left(\frac{1}{T_1}\frac{1-e^{-sT_1}}{s}\right) \tag{3}$$

and the resulting velocity profile when a rectangular feed pulse $v(t)$ is filtered $T_1 = T_2 < T_v$ becomes:

$$v'(t) = v(t) * m_{2FIR}(t) \tag{4}$$

$$v'(t) = \begin{cases} \frac{F}{2T_1^2}t^2 & 0 \le t < T_1 \\ \frac{F}{2T_1}\left(-t^2 + 4T_1 t - 2T_1^2\right) & T_1 \le t < 2T_1 \\ F & 2T_1 \le t < T_v \\ \frac{F}{2T_1}\left(-t^2 + 2T_v t - T_v^2 + 2T_1^2\right) & T_v \le t < T_v + T_1 \\ \frac{F}{2T_1^2}\left(t^2 - 2T_v t - 4T_1 t + (T_v + 2T_1)^2\right) & T_v + T_1 \le t < T_v + 2T_1 \end{cases} \tag{5}$$

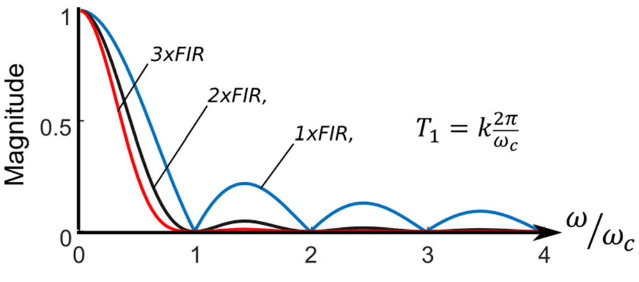


Fig. 3 Magnitude of the frequency response of multiple FIR filters

The corresponding acceleration and jerk responses can be derived from Eq. 5 as:

$$a'(t) = \begin{cases} \frac{F}{T_1^2}t & 0 \le t < T_1 \\ \frac{F}{T_1^2}(-t + 2T_1) & T_1 \le t < 2T_1 \\ 0 & 2T_1 \le t < T_v \\ \frac{F}{T_1^2}(-t + T_v) & T_v \le t < T_v + T_1 \\ \frac{F}{T_1^2}(t - T_v - 2T_1) & T_v + T_1 \le t < T_v + 2T_1 \end{cases} \tag{6}$$

$$j'(t) = \begin{cases} \frac{F}{T_1^2} & 0 \le t < T_1 \\ -\frac{F}{T_1^2} & T_1 \le t < 2T_1 \\ 0 & 2T_1 \le t < T_v \\ -\frac{F}{T_1^2} & T_v \le t < T_v + T_1 \\ \frac{F}{T_1^2} & T_v + T_1 \le t < T_v + 2T_1 \end{cases} \tag{7}$$

When a square velocity pulse of magnitude F and length T_v is convolved with a first-order FIR filter with time constant T_1, the result is a trapezoidal velocity profile with constant acceleration of magnitude F/T_1 (Fig. 4b). The total length of the kinematic profiles is extended by the filter time constant T_1 to $T_v + T_1$. When the trapezoidal velocity profile is convolved with a second first-order FIR filter with a matching time constant $T_1 = T_2$, the smoothness (order) of the velocity profile is increased. The continuity increases from C^1 to C^2, where C^n is the space of n$^{\text{th}}$ order continuously differentiable functions, as shown in Eq. 5 and Fig. 4c. However, using the matching time constant $T_1 = T_2$ results in five sections in the kinematic profile and not seven as for the case for two different time constants where $T_1 \neq T_2$. The resulting acceleration profile is triangular around T_1 and $T_v + T_1$ with peak magnitudes F/T_1 and lengths of $2T_1$; the now jerk-limited profile has peak magnitudes of F/T_1^2. The total length of the kinematic profiles is extended to $T_v + 2T_1$. The relationship between T_1 and T_v determines the kinematic constraints as for the different filter cases.

Convolving the velocity profile with a third first-order FIR filter with the same time constant $T_1 = T_2 = T_3$ results in a C^3 velocity profile, C^2 acceleration profile and C^1 jerk profile. The velocity, acceleration and jerk equations for the 3-FIR case are shown in Appendix A. The smooth acceleration profile has a peak magnitude of $3F/4T_1$ at times $1.5T_1$ and $T_v + 1.5T_1$ and the jerk profile has peak magnitudes of F/T_1^2. The overall length of the kinematic profiles has been extended from the original square velocity pulse length T_v to $T_v + 3T_1$. The total filter delay when using 3-FIR filters with matching time constants T_1 is therefore $3T_1$.

It can be shown that a high order FIR filter can be accurately modelled and implemented with using only 3 first order FIR filters. The benefit of using 3 or more first

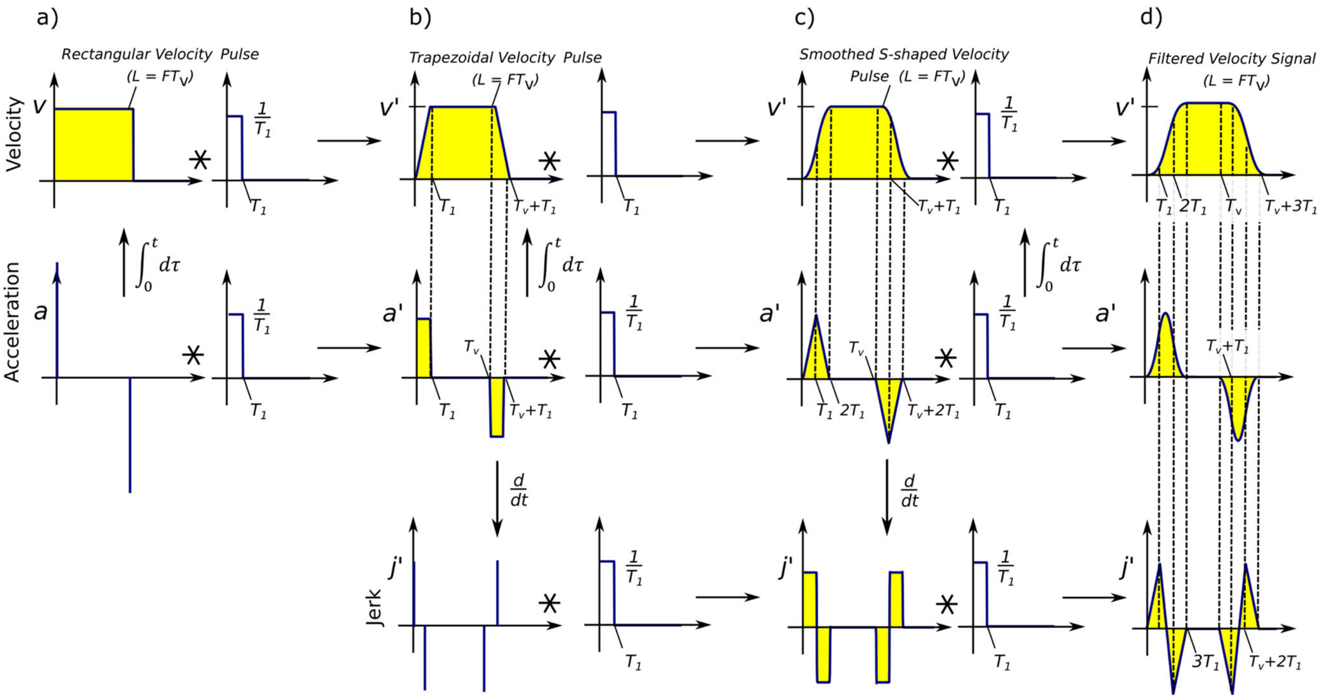


Fig. 4 Smooth trajectory generation by 3 first-order FIR filters with matching time constants

order FIR filters with the same time constant is that the filter response approaches that from a Gaussian filter. The Gaussian response has no overshoot whilst minimising the acceleration and deceleration time periods which makes it the ideal time domain filter for interpolating kinematic profiles [28]. The ability to approximate the Gaussian filter with 3 FIR filters with the same time constant simplifies the design and selection of the filter to a single design parameter T_1. For both the 2 and 3 FIR filter cases, T_1 can be analytically calculated from the maximum permissible jerk J_{max} using Eq. 6 and Eq. 32 respectively as follows:

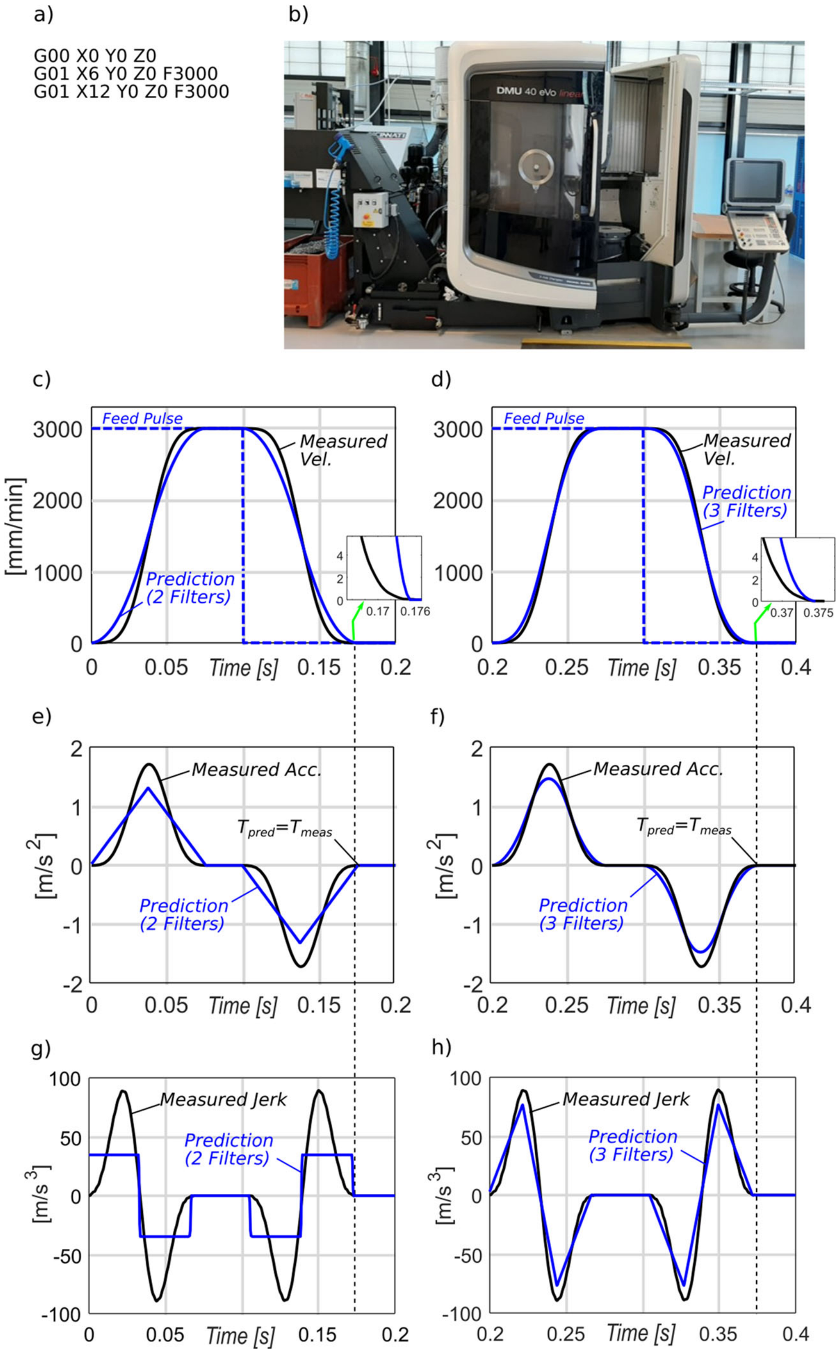
$$J_{max} = \frac{\Delta F}{T_1^2}, \longrightarrow T_1 = \sqrt{\frac{\Delta F}{J_{max}}} \tag{8}$$

2.2 Identification of real-time interpolator dynamics of an NC system

The previous section presented the filtering-based real-time trajectory generation. In this section, it is shown how the interpolator response of a machine tool can be modelled via the identification of the filter time constants. A case study was conducted on the DMG Mori eVo 40 machine tool shown in Fig. 5. The machine is commanded by a single G-code to move 6 mm at a speed of 3000 mm/min, and the interpolated reference motion profile is recorded on the NC system directly at a sampling time of $T_s = 0.003s$. Figure 5c to h show the recorded kinematic profiles. The machine is set to undergo a simple point-to-point (P2P) motion and therefore the tool comes to a full stop before moving to the next commanded position. As shown for the measured system, the NC system generates smooth velocity and acceleration profiles. The acceleration profile mimics a smooth "bell-shaped" profile. Overall, acceleration and deceleration duration are measured to be $T_{acc} = T_{dec} =$ 0.0765 s. The cruise velocity portion is roughly measured to be 0.023 s. In order to simulate the feed profile, a series of 2- and 3-FIR filters are used. For the 2-FIR case, the time constant is selected as $T_1 = \frac{T_{acc}}{2}$ and for the 3-FIR case, it is set to $T_1 = \frac{T_{acc}}{3}$. The predicted velocity, acceleration and jerk profiles for the 2-FIR case are shown in Fig. 5c, e and h respectively. The time of the measured displacement is equal to the time of the predicted displacement. The difference between the velocity profiles is due to the acceleration. The 2-FIR case exhibits the triangular acceleration profile compared to the smooth measured response. The maximum acceleration for the 2-FIR case is constrained and less than the measured response.

In order to compare the different filter cases, the machine is commanded to move along the same G-code, and the proposed interpolator model for the 3-FIR case is used. As shown in Fig. 5d, the velocity profiles for the 3-FIR case closely resemble the measured velocity profile and the total time of the measured displacement matches the total time for the simulated displacement. The

Fig. 5 Measured and predicted velocity, acceleration and jerk profiles for 2-FIR (c,e,g) and 3-FIR (d,f,h) P2P motion interpolation



```
G00 X0 Y0 Z0
G01 X6 Y0 Z0 F3000
G01 X12 Y0 Z0 F3000
```

simulated acceleration profile is smooth and the maximum acceleration is higher than for the 2-FIR case but still lower than the measured response. Increasing the order of the simulated system would allow the maximum acceleration to approach the measured response. In general, by increasing the order of the FIR filter, the predicted acceleration profile of the filtered pulse approaches the acceleration profile of the measured response and results in a simulated velocity profile which closely resembles the dynamics of the machine interpolator.

The filter delay is calculated from the jerk Eq. 8 and the duration of the acceleration phase in each case is equal to the total filter delay. The time constant (filter delay) can be analytically calculated from machine tools' specifications (J_{max}) and therefore kinematic profiles can be generated using FIR filters without the requirement for parameter identification through system testing.

In this section, it has been shown that the dynamics of an NC interpolator are increasingly well-approximated by the series combination of identical first-order FIR filters. In addition, the relationship between the parameters of these first-order filters and the resulting interpolator response has been derived.

2.3 Multi-axis P2P motion generation

FIR filtering–based interpolation of single axis motion was presented in the previous sections. Extending the method to P2P multi-axis linear motion this section describes the process to interpolate kinematic profiles between two points using high order FIR filters.

The start and end positions of a linear G01 command in 3 axes can be represented by $\mathbf{P}_\mathrm{s} = \left[P_{s,x}, P_{s,y}, P_{s,z}\right]^T$ and $\mathbf{P}_\mathrm{e} = \left[P_{e,x}, P_{e,y}, P_{e,z}\right]^T$, respectively, as shown in Fig. 6a.

The tool displacement L is calculated by taking the Euclidean norm of the vector between the two commanded positions, $L = \|\mathbf{P}_e - \mathbf{P}_s\|_2$. The velocity pulses of each axis $\left(v_x, v_y, v_z\right)$ are calculated by multiplying the feed pulse $v(t)$ by the unit velocity vector $\mathbf{u} = (\mathbf{P_e} - \mathbf{P_s})/\|\mathbf{P_e} - \mathbf{P_s}\|_2$.

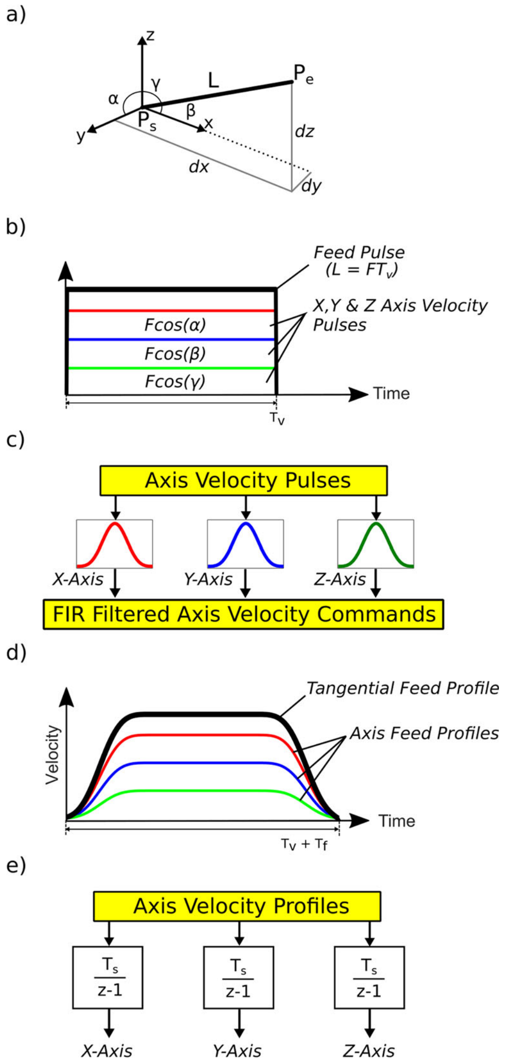


Fig. 6 Multi-axis interpolation based on high order FIR filtering

$$\frac{d\mathbf{P(t)}}{dt} = \dot{\mathbf{P}}\mathbf{(t)} = v(t)\mathbf{u} = \begin{bmatrix} v_x(t) \\ v_y(t) \\ v_z(t) \end{bmatrix} \tag{9}$$

where $\dot{\mathbf{P}}\mathbf{(t)}$ represents the first time derivative of the P2P displacement (Fig. 6b).

In order to generate (and interpolate) the reference velocity commands (v_x', v_y', v_z'), the individual axis velocity pulses (v_x, v_y, v_z) are convolved with the FIR filter (Fig. 6c and d):

$$\frac{d\mathbf{P}'(t)}{dt} = \dot{\mathbf{P}}'(t) = \begin{bmatrix} v_x'(t) \\ v_y'(t) \\ v_z'(t) \end{bmatrix} = \dot{\mathbf{P}}(t) * m(t) \tag{10}$$

Finally, the filtered position commands are generated by integrating the filtered axis velocity commands:

$$\mathbf{P}'(t) = \begin{bmatrix} p_x'(t) \\ p_y'(t) \\ p_z'(t) \end{bmatrix} = \int_0^t \begin{bmatrix} v_x'(t) \\ v_y'(t) \\ v_z'(t) \end{bmatrix} d\tau \tag{11}$$

3 Prediction of interpolator behaviour during non-stop motion

The previous section showed that P2P linear interpolation behaviour of an NC system can be modelled by velocity pulses low-pass filtered by a series of first-order FIR filters. The only required parameter to predict the machine's feed profile and accurately estimate the resulting cycle time is the time constant, i.e. total delay of the FIR filter. As shown, the filter time delay can be calculated from the maximum permissible jerk (Eq. 8) and commanded feedrate. This section focuses on accurate prediction of interpolator behaviour during non-stop contouring motion, which is the most commonly used interpolation technique for high-speed machining (HSM).

3.1 Modelling of non-stop (contouring) interpolation behaviour

Typical high-speed machining toolpaths found in die and mould manufacturing or in aerospace industry consist of series of short segmented toolpaths [29]. When interpolated in HSM mode, the NC interpolator does not undergo a full stop at the end of each CL line. Instead, the CL lines are blended together for a non-stop smooth motion interpolation where machining feedrate is reduced to a cornering speed V_c around junction points of the CL blocks (see Fig. 1). The prediction of V_c is crucial to accurately capture the actual feedrate profile and estimate the resultant cycle time. Several constraints affect the cornering speed

(V_c) and overall acceleration profile around the CL data points. Firstly, V_c is controlled by the blending (cornering) tolerance [16]. Typically, lower blending tolerance delivers more accurate motion but generates slower feed profiles. In contrary, a larger tolerance value allows faster speeds and shorter overall cycle time. The relationship between the blending tolerance and the feed drop around the corner must be captured. Secondly, the deceleration/acceleration profile and the transition duration from the programmed feedrate (F) to the cornering speed (V_c) are dictated by acceleration and jerk limits of the machine. Both of these key characteristics must be modelled to accurately predict the varying feedrate profile along HSM toolpaths.

In an effort to accurately model the interpolator behaviour, the feed pulse distribution shown in Fig. 7b is proposed in this manuscript. Notice that the feed pulse profile is different from the case used for the P2P motion. Feed pulses of each CL block are commanded back-to-back with no dwell time in between. In other words, they are constructed as a continuous pulse stream. The duration of the feed pulse is T_v. Notice that the feed pulse does not have a constant amplitude of F. Instead, around CL block junctions the feed command value is dropped down to F_c. Such small feed pulse is added to model the blending kinematics, commanding the feedrate to drop down to a cornering feed of F_c. The duration of the cornering feed pulse is set to T_b, which controls how long the deceleration and acceleration last around the blend.

When the feed pulse profile is interpolated with a FIR filter, the resulting velocity profiles are smooth velocity profiles that better approximate the actual velocity profiles of the machining interpolator. Figure 7a and Fig. 7c show the toolpath and the corresponding interpolated X-axis and Y-axis velocity profiles respectively. The total length of the velocity profiles is equal to the sum of the pulse lengths plus the filter delay T_d. Figure 7d shows the cornering feedrate V_c of the resultant velocity profile is equal to the commanded blending pulse feedrate F_c and this occurs at half the filter delay $T_d/2$ from the start of the Y-axis profile.

The cornering feedrate is controlled by setting the blending velocity pulse F_c equal to the desired cornering tangential velocity V_c and setting the acceleration and deceleration time for the interpolated feed profile equal to the time required to reduce from F to F_c. A scaling factor is applied to F to represent F_c as a function of commanded feedrate F:

$$F_c = F\alpha = V_c \tag{12}$$

where V_c is the resultant 3-axis TCP velocity defined as

$$V_c = \sqrt{v_x'^2 + v_y'^2 + v_z'^2} \tag{13}$$

and v_x', v_y' and v_z' represent the interpolated axis velocities at the minimum corning feedrate.

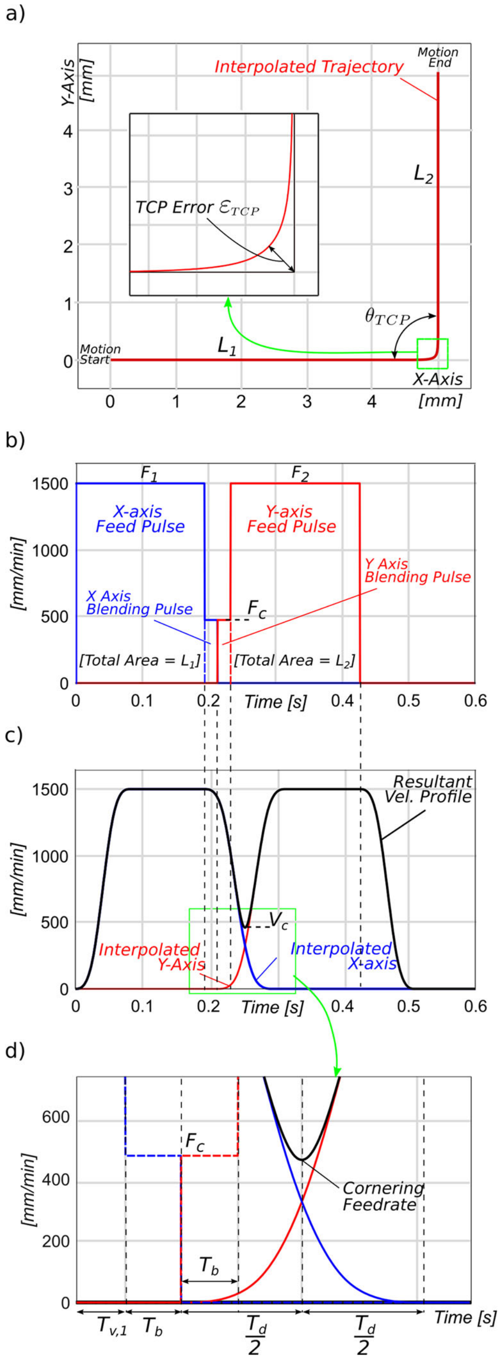


Fig. 7 FIR-based interpolation of a right angled toolpath with a constant feedrate

The total acceleration and deceleration time of the interpolated feed profile to reach $F\alpha$ from F is represented by T_b, it is a function of the filter delay T_d, and it can be calculated as:

$$T_b = \frac{1}{2}T_d\,(1 - \alpha) \tag{14}$$

The final task when modelling the pulse train is calculating the main velocity pulse lengths T_v. In Section 2, the length of the velocity pulse T_v was calculated from L/F; however, with the introduction of the blending pulses, T_v must be modified in order to preserve the total area of the pulses and hence the TCP displacement.

The commanded TCP displacement is calculated from the total area of the velocity pulse and the blending pulse, and this can be seen in Fig. 7b where the total area within the X-axis and Y-axis pulses is equal to L_1 and L_2 respectively.

For a single axis displacement L, the pulse areas comprise of the main pulse (calculated as FT_v) and the blending pulse (calculated as F_cT_b):

$$L = FT_v + F_cT_b \tag{15}$$

Rearranging Eq. 15 and incorporating Eq. 12 yields the modified value of T_v as:

$$T_v = \frac{L}{F} - \alpha T_b \tag{16}$$

Equation 16 holds for velocity commands with a single blending pulse; this is the case for the initial and final CL lines in a part program which start and end at zero feedrate (full stop). The remaining displacements in a part program are continuous and therefore the commands consist of a velocity pulse with a blending pulse either side as shown in Fig. 8. Accordingly, each cornering blend consists of two back-to-back blending pulses.

For the entire pulse train, each G01 command or CL line can be represented by an index k with k=1 corresponding to the initial command in the part program. The associated feedrate commands in the part program are hence denoted $F(k)$. Therefore, for the main commands in a part program, the modified value of T_v is calculated as:

$$T_v(k) = \frac{L(k)}{F(k)} - \alpha(k)T_b(k) - \alpha(k+1)T_b(k+1) \tag{17}$$

For constant feedrate, the adjoining blending pulses are symmetric. This leads to symmetrical interpolated velocity profiles and results in symmetrical displacement profiles, translating to the same toolpath trajectory for both forward and backward passes resulting in a more accurate finish.

3.2 Filtered signal generation

The composition of the velocity pulses and filtered kinematic profiles was shown in the previous section. In practice, the strategy for interpolation of multi-segmented NC toolpaths using high order FIR filtering, as shown in Fig. 8, is as follows:

1. Read NC code and parse commanded X,Y, Z positions and feedrate commands from individual G01 commands and extract defined tolerance setting.
2. Calculate toolpath geometry (cornering angles θ_{TCP}) and unit velocity vectors for each G01 command.
3. Calculate cornering feedrates where $V_c = F_c = F\alpha$ from the maximum permissible feedrate for the cornering angle and defined tolerance (Eq. 29 and Eq. 30) (demonstrated in Section 3.3).
4. Calculate pulse velocities F and velocity blending pulse widths T_b followed by modified velocity pulse widths T_v (Eq. 16 and Eq. 17).
5. Synchronise timed axis velocity pulses and generate unfiltered axis velocity signals (pulse train).
6. Define FIR filter time constants for the commanded feedrate from maximum permissible jerk (Eq. 8).
7. Using high order FIR filtering with matching time constants interpolate the axis velocity pulse signals to generate smooth kinematic profiles for each axis (Eq. 10).
8. Finally, integrate the filtered velocity signals to generate synchronised accurate position commands in the time domain (Eq. 11).

Sections 3.1 and 3.2 described the components of the velocity pulse train and application of FIR filtering for generation of kinematic profiles for non-stop high speed motion. The following sections will analytically demonstrate the relationship between the cornering speed V_c to the blending error and axis kinematic limits and ultimately demonstrate how F_c is selected to guarantee these constraints are satisfied.

3.3 Kinematic profiles for the 2-FIR filter case

The geometry of velocity blending pulses was presented and calculated in Section 3.1. The pulse signals are interpolated using FIR filters to generate kinematic profiles that control the cornering feedrate. This section analytically derives the equations for the kinematic profiles when using velocity blending pulses and FIR filtering–based interpolation to control the cornering feedrate. In doing so, the authors are able to analytically calculate the blending pulse feedrate command F_c which satisfies both TCP error and machine kinematic constraints during cornering transitions.

Using 2-FIR filters with matching time constants to interpolate a velocity pulse signal results in the kinematic profiles shown in Fig. 9. The profiles are split into 5 sections during acceleration/deceleration as shown in Fig. 9b for the Y-axis acceleration. The objective of the analytical expressions is to calculate the interpolated displacement at the point of maximum TCP error and the interpolated velocity at the minimum cornering feedrate. This occurs at half the total filter delay $T_d/2$ (see Fig. 9a). The total filter delay for the 2-FIR case is $T_d = 2T_1$, where T_1 is calculated from the maximum permissible jerk (Eq. 8), resulting in the maximum TCP error and minimum cornering feedrate occuring at T_1. Figure 9b shows T_1 is at the start of section 3;

therefore, only sections 1–3 of the kinematic profiles need considering. The analytical expressions for sections 1–3 of the displacement, velocity, acceleration and jerk profiles for the 2-FIR case are presented in Eqs. 18 to 21 respectively.

$$s'(t)=\begin{cases} \frac{1}{6}\frac{\alpha F}{T_1^2}t^3 & 0 \le t < T_b \\ \frac{1}{6}\frac{F}{T_1^2}\left(t^3+3T_b\left(\alpha-1\right)t^2+3T_b\left(1-\alpha\right)t+T_b^3\left(\alpha-1\right)\right) & T_b \le t < T_1 \\ \frac{1}{6}\frac{F}{T_1^2}((1-2\alpha)\,t^3+(3T_b\left(\alpha-1\right)+6T_1\alpha)\,t^2+(3T_b^2\left(1-\alpha\right)+\ldots & \\ \ldots-6T_1^2\alpha)t+T_b^3\left(\alpha-1\right)+2T_1^3\alpha) & T_1 \le t < T_b+T_1 \end{cases} \tag{18}$$

$$v'(t)=\begin{cases} \frac{1}{2}\frac{\alpha F}{T_1^2}t^2 & 0 \le t < T_b \\ \frac{1}{2}\frac{F}{T_1^2}\left(t^2+2T_b\left(\alpha-1\right)t-T_b^2\left(\alpha-1\right)\right) & T_b \le t < T_1 \\ \frac{F}{2T_1^2}((1-2\alpha)\,t^2+(T_1\alpha+2T_b\left(\alpha-1\right)+4T_1\alpha)\,t+T_1^2\alpha+\ldots & \\ \ldots+T_b^2\left(1-\alpha\right)-2T_1\alpha) & T_1 \le t < T_b+T_1 \end{cases} \tag{19}$$

$$a'(t)=\begin{cases} \frac{\alpha F}{T_1^2}t & 0 \le t < T_b \\ \frac{F}{T_1^2}\left(t+T_b\left(\alpha-1\right)\right) & T_b \le t < T_1 \\ \frac{F}{T_1^2}\left(\left(1-2\alpha\right)t+2T_1\alpha-T_b+T_b\alpha\right)+ & T_1 \le t < T_b+T_1 \end{cases} \tag{20}$$

$$j'(t)=\begin{cases} \frac{\alpha F}{T_1^2} & 0 \le t < T_b \\ \frac{F}{T_1^2} & T_b \le t < T_1 \\ \frac{F}{T_1^2}\left(1-2\alpha\right) & T_1 \le t < T_b+T_1 \end{cases} \tag{21}$$

The interpolated axis velocity at maximum TCP error (minimum cornering feedrate) occurs at $t = T_d/2 = T_1$; therefore, in the 2-FIR filter case, this results in the following expressions for interpolated velocity (Eq. 22) and displacement (Eq. 23):

$$v' = \frac{1}{2}\frac{F}{T_1^2}\left(T_1^2 - T_b^2\left(1-\alpha\right) + 2T_1T_b\left(\alpha-1\right)\right) \tag{22}$$

$$s' = \frac{1}{6}\frac{F}{T_1^2}\left(T_1^3 + 3T_1T_b^2\left(1-\alpha\right) + 3T_1^2T_b\left(\alpha-1\right) + T_b^3\left(\alpha-1\right)\right) \tag{23}$$

Using Eq. 14, the interpolated displacement (Eq. 22) and velocity (Eq. 23) can be expressed in terms of F and α:

$$v' = \frac{F}{2}\alpha\left(-\alpha^2+\alpha+1\right) \tag{24}$$

$$s' = \frac{F}{6}T_1\alpha\left(-\alpha^3+\alpha^2+1\right) \tag{25}$$

Figure 10 shows a cornering transition between two CL-lines or G01 commands. The maximum TCP contouring or corner blending error ε_{TCP} occurs in the centre of the cornering trajectory and is calculated by evaluating the interpolated axis displacements s' at $t = T_1$. The interpolated axis displacements are calculated from Eq. 25, and the vectors from the corner transition to these positions are represented by $\mathbf{l_1}$ and $\mathbf{l_2}$.

The contouring error ε_{TCP} (shown in Fig. 10) is calculated from the Euclidean distance between the vectors $\mathbf{l_1}$ and $\mathbf{l_2}$.

$$\varepsilon_{TCP} = \|\mathbf{l_2}-\mathbf{l_1}\| = \sqrt{l_1^2+l_2^2+2l_1l_2\cos\theta_{TCP}} \tag{26}$$

where θ_{TCP} represents the TCP cornering angle. Assuming constant feedrate in this example, $l_1 = l_2 = l_\varepsilon$, in which case, Eq. 26 simplifies to the following expression:

$$\varepsilon_{TCP}^2 \le 2l_\varepsilon^2\left(1+\cos\theta_{TCP}\right) \tag{27}$$

Inserting Eq. 14 and Eq. 23 into Eq. 26 enables the TCP corner blending error to be defined as:

$$\varepsilon_{TCP} = \frac{\sqrt{2}}{6}\sqrt{F^2\,{T_1}^2\,\alpha^2\,\left(\cos\theta_{TCP}+1\right)\left(-\alpha^3+\alpha^2+1\right)^2} \tag{28}$$

Using Eq. 28, the TCP error can be calculated for any toolpath geometry and commanded feedrate. The kinematic profiles for the 3-FIR case are shown in Fig. 17 in Appendix B and the derivation of TCP error for the 3-FIR case is included in Appendix C.

To ensure minimum cycle times, the actual feedrate must remain as close to the commanded feedrate as possible throughout the toolpath including cornering transitions. However, to satisfy both jerk and TCP error constraints, there is a maximum permissible cornering feedrate. Using Eqs. 28 and 42, it is possible to calculate the relationship

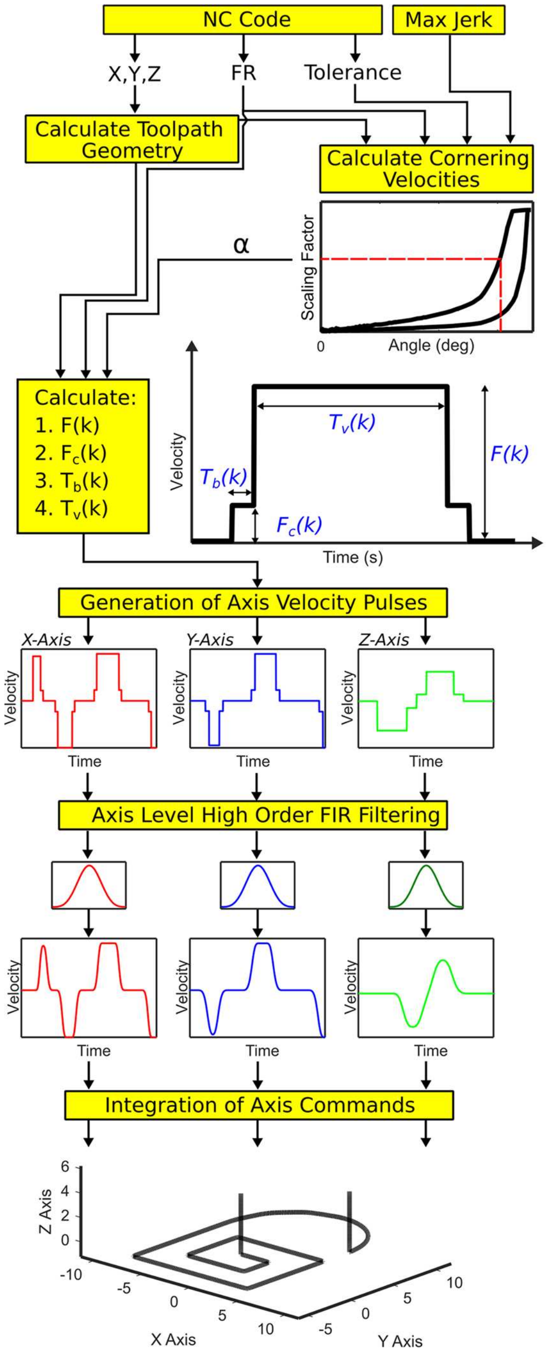


Fig. 8 Non-stop interpolation of kinematic profiles using high order FIR filtering

between TCP error, maximum permissible cornering feedrate and cornering angle for the 2-FIR and 3-FIR filter cases respectively.

Rearranging Eq. 28, the maximum permissible cornering feedrate for the 2-FIR filter case must satisfy:

$$FT_1\sqrt{\cos\left(\theta_{TCP}+1\right)}\left(\alpha^4-\alpha^3-\alpha\right)-3\sqrt{2}\varepsilon_{TCP}\leq 0 \tag{29}$$

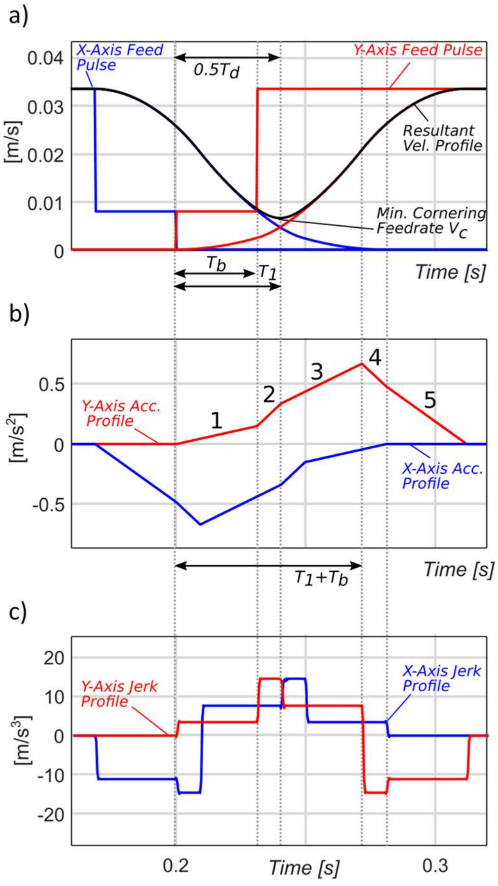


Fig. 9 Interpolated kinematic profiles of velocity blending pulses using 2-FIR filters

and for the 3-FIR filter using Eq. 42, the maximum permissible cornering feedrate must satisfy:

$$FT_1\sqrt{\cos\left(\theta_{TCP}+1\right)}\left(16\alpha^5+16\alpha^4-8\alpha^3-16\alpha^2-85\alpha-1\right)$$
$$-192\sqrt{2}\varepsilon_{TCP}\leq 0 \tag{30}$$

For a commanded feedrate F and range of cornering angles $\theta_{TCP}\in[0°,180°]$, Eqs. 29 and 30 are solved for solutions $0<\alpha\leq 1$ to calculate the limit to the feedrate scaling factor α. When multiplied by the

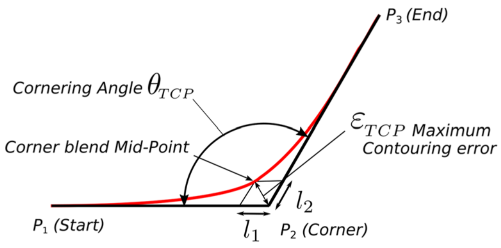

P_3 (End)
Cornering Angle θ_{TCP}
ε_{TCP} Maximum Contouring error
Corner blend Mid-Point
l_2
P_1 (Start)
l_1
P_2 (Corner)

Fig. 10 Toolpath showing contouring error and cornering angle between two consecutive G01 commands

commanded feedrate F, this represents the maximum permissible cornering feedrate that can be achieved whilst satisfying the kinematic and tolerance constraints. The blending pulse feedrate F_c is commanded to this limit value.

The reduction in cornering feedrate for both the 2-FIR and 3-FIR filter cases is shown in Fig. 11. Cornering feedrates selected below the curves will satisfy the TCP error constraints for the commanded feedrate and cornering angle. The figure shows the limits for 10μm and 50μm tolerance constraints. For the 50μm tolerance, the figure shows higher cornering feedrates can be achieved compared to the 10 μm case. It can also be recognised that higher feedrates can be achieved in the 3-FIR case. Therefore, there is an advantage of using 3-FIR filters to reduce the overall machining cycle time as the tool can remain at higher feedrates during cornering transitions than for the 2-FIR case. Despite the advantage of using a higher order filter, there remains a limit to the order of filters that can be used effectively for trajectory generation. As the order is increased, the filter time constant reduces. In the frequency domain, the notch (as shown in Fig. 3) will shift to higher frequencies. This will be constrained by the lowest structural mode of the machine tool.

This section has shown a method of using multiple first-order FIR filters with matching time constants to model continuous linear interpolation of velocity pulse signals. It has been shown that the cornering feedrate and TCP error can be controlled using velocity blending pulses. This method has been extended to predict feedrates and machining cycle time for toolpaths of any geometry and defined TCP tolerance. The following section demonstrates and validates the proposed method on industrial case studies.

4 Experimental validation

Machining experiments were conducted on a DMG Mori Universal eVo 40 5-axis machining centre with a Heidenhain TNC640 controller. Two short toolpaths were used for pocketing operations and a single long aerospace part program is evaluated in the cycle time prediction. The section concludes with validation of the feedrate prediction method integrated with a virtual machining model.

4.1 Case studies on pocketing toolpaths

The first two case studies, as shown in Fig. 12, consist of a contour and a trochoidal pocketing toolpath. These toolpaths are generated by CAM software [30], and the part programs are deployed to the machine directly with no modification. Table 1 shows the cutting conditions. As noted, 2 different feedrates 1000 and 3000 mm/min are used. The most important setting is contour error tolerance for HSM. Two different contouring tolerances, 10 and 50 μm, are used. Table 1 summarises the cycle time results. All simulated trajectories in the case studies were modelled using the method described in Section 3.2 and 3-FIR filters.

4.1.1 Machining cycle time estimation

The predicted machining cycle times are compared with the measured CNC and CAD/CAM calculated machining cycle times. The results are presented in Table 1 and Fig. 13a. For all cases, the predicted machining cycle times are accurate to within 3% of the measured cycle time with the exception of the trochoidal pocket (1000 mm/min, 10μm case) which is 5.52%. These compare favourably to the CAD/CAM calculated cycle times which has an error range from 0.22 to 54.99%. The significant result is the trochoidal pocket (3000 mm/min and 10μm case). The proposed method is able to accurately predict the increase in machining cycle time from 14.40 to 30.99 s when tightening the tolerance from 50 to 10μm, which is within 2.72% of the measured cycle time. This is compared to an error of 54.99% for the CAD/CAM calculated cycle time.

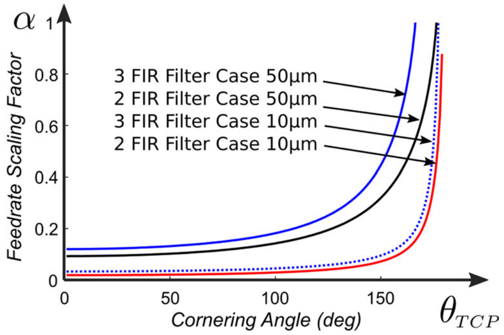


Fig. 11 Minimum cornering feedrate and cornering angle curves shown for 10μm and 50μm tolerance settings at 2000 mm/min for both 2 and 3 first-order FIR cases

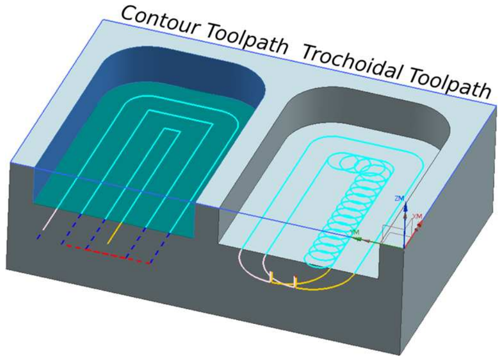


Fig. 12 Contour (left) and trochoidal (right) pocketing toolpaths designed in Siemens NX CAM

Table 1 Machining cycle times for contour and trochoidal pocket at different machining parameters

Case study	Feedrate (mm/min)	Tolerance (microns)	Measured time (s)	CAD/CAM time (s)	FIR prediction time (s)	CAD/CAM prediction error (%)	FIR prediction error (%)
Contour pocket	1000	10	27.07	26.56	27.32	-1.88	0.92
Contour pocket	1000	50	26.82	26.56	26.95	-0.97	0.48
Contour pocket	3000	10	10.37	8.85	10.46	-14.66	0.87
Contour pocket	3000	50	9.6	8.85	9.73	-7.81	1.35
Trochoidal pocket	1000	10	42.38	40.69	44.75	-3.98	5.59
Trochoidal pocket	1000	50	40.78	40.69	41.30	-0.22	1.28
Trochoidal pocket	3000	10	30.13	13.56	30.99	-54.99	2.86
Trochoidal pocket	3000	50	14.04	13.56	14.40	-3.41	2.56

4.1.2 Prediction of feedrate profile

To demonstrate the performance of the feedrate prediction method, a number of toolpath features were selected. The predicted, CAD/CAM calculated and measured CNC tangential velocities at these particular features were recorded and are presented in Tables 2 and 3. The contour pocket features consist of (1) a long G01 segment, (2) a sharp corner and (3) a rounded corner consisting of small G01 segments. The trochoidal pocket features consist of (1) the stepover segment and (2) the main arc. Depending on the tolerance and the commanded feedrate, large differences in tangential velocity can exist between the stepover segment and the main arc of a trochoidal toolpath which in turn results in a large cyclical variation of cutting forces. It is for this reason they are included in this study. The features described above are shown on the toolpaths in Fig. 13 and the corresponding position with respect to displacement and tangential velocities are demonstrated directly beneath. Overall, the prediction error ranges from 0.1 to 10.3% compared with CAD/CAM calculated error range of 0.22–2555%, where the error is calculated as a percentage difference from, and with respect to, the measured tangential velocity. The performance of the proposed feedrate prediction method at each feature is described below:

Long G01 segment The prediction error range is between 0.1 and 0.13% compared to the CAD/CAM calculated error range of 0.27–0.3%. The high accuracy is to be expected as no feedrate limiting features are present in the segment. The differences in measured velocity compared to the idealised CAD/CAM values are due to interpolator rounding during trajectory generation.

Sharp corner The prediction error range is between 7 and 20% compared to the CAD/CAM calculated error range of 107–2555%. The fundamental difference is due to the CAD/CAM calculation not taking into account the cornering kinematic constraints due to tolerance and thus not predicting the reduction in feedrate during the cornering segment. This holds true for all of the features demonstrated except the long G01 segment. For the 10μm tolerance cases, the tool comes to an almost complete stop — 4% and 11% of the commanded feedrate for the 3000 mm/min and 1000 mm/min cases, respectively; the presented method predicts these reductions.

Rounded corner The prediction error range is between 0.3 and 5.4% compared to the CAD/CAM calculated error range of 0.2 and 168%. The significant result is the 3000 mm/min and 10μm case (Fig. 13c) where the CAD/CAM calculation does not account for the reduction in velocity due to the tolerance requirement. The CAD/CAM calculated error is 168% compared to the measured value and the prediction error is within 2.4%.

Trochoid stepover The prediction error range is between 0.1 and 10% compared to the CAD/CAM prediction error range of 0.2–0.5%. The CAD/CAM calculation does not predict any differences along the trochoidal toolpath between the stepover and the main arc. This can be seen in Fig. 13d for the 3000 mm/min 10μm case. The blue line shows the CAD/CAM prediction but the actual kinematic profile is very different. The stepover results in tangential velocities close to the commanded feedrate as the cornering angles between the segments are less acute than for the rest of the main arc.

Trochoid main arc The prediction error range is between 0.6 and 7% compared to the CAD/CAM calculated error range of 0.2–208%. The reduction in tangential velocity around the main arc is due to the cornering angles between the segments. The influence of the toolpath tolerance on the

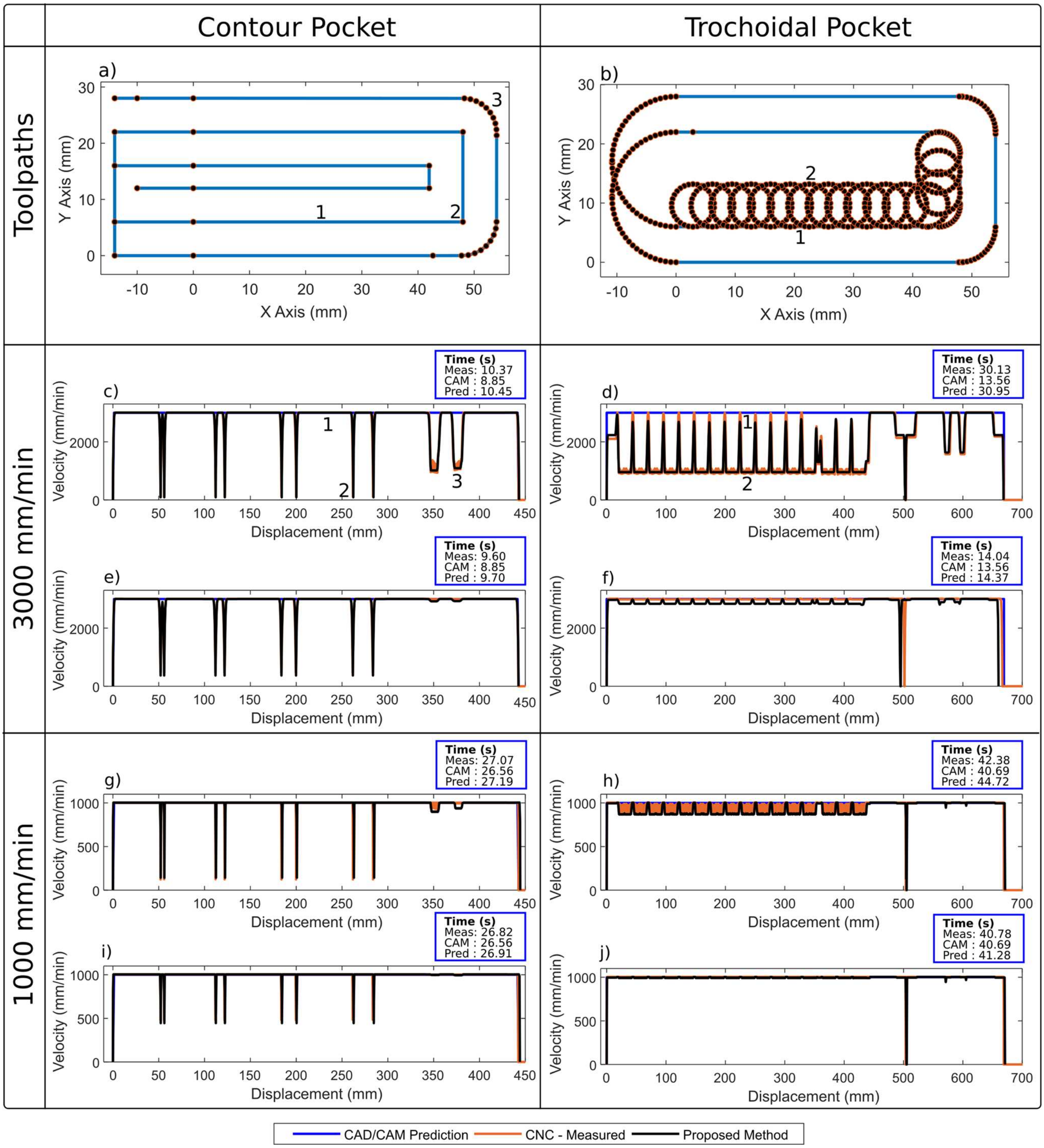


Fig. 13 Pocketing case studies — predicted, measured and CAD/CAM tangential velocities

cornering tangential velocity can be seen in Fig. 13d and Fig. 13f. The increase in tolerance from 50 to 10μm results in more than a 65% reduction in tangential velocity around the main arcs of the trochoids. The prediction method accurately predicts the feedrate within 1.5% of tangential velocity measured at the main arc. Taking this result one step further, this demonstrates that a feedrate driven cutting force model when incorporating the prediction method will

Table 2 Contour pocket case study: tangential velocity prediction and performance

Case study	Feedrate (mm/min)	Tolerance (microns)	Analysis point	Tangential velocities (mm/min)				
				Measured	CAD/CAM	Proposed prediction	CAD/CAM error	Proposed prediction error
Contour	1000	10	Point 1	1003	1000	1004	-0.30%	+0.10%
Contour	1000	50	Straight	1003	1000	1004	-0.30%	+0.10%
Contour	3000	10	G01	3008	3000	3005	-0.27%	-0.10%
Contour	3000	50		3008	3000	3004	-0.27%	-0.13%
Contour	1000	10	Point 2	124	1000	143	+706%	+15%
Contour	1000	50	Sharp	482	1000	446	+107%	-7%
Contour	3000	10	Corner	113	3000	93	+2555%	-17%
Contour	3000	50		465	3000	370	+545%	-20%
Contour	1000	10	Point 3	994	1000	940	+0.6%	-5.4%
Contour	1000	50	Rounded	998	1000	995	+0.2%	-0.3%
Contour	3000	10	Corner	1120	3000	1093	+168%	-2.4%
Contour	3000	50		2996	3000	2929	+0.13%	-2.23%

be able to predict the cyclical cutting forces due to the 65% variation in magnitude of feedrate fluctuations around the trochoidal toolpath.

4.2 Case study 3 — aerostructure toolpath

An industrial toolpath was chosen to validate the method against a representative aerostructure part. The part program consists of three toolpaths — roughing, finishing #1 floors and finishing #2 walls as shown in Fig. 14. The part programs were run at three tolerance settings, 10μm, 20μm and 50μm to demonstrate the significant impact tolerance has on machining cycle times and therefore on feedrate and cycle time prediction. Table 4 compares the predicted machining cycle times with both the measured cycle times and the predicted times from a commercial CAD/CAM software package for each individual toolpath.

The overall machining cycle times, calculated by summing the cycle times for the 3 sections of the part program, are shown in Table 5. The CAD/CAM prediction error ranges from 62.41% under prediction for the 10μm case to 36.42% under prediction for the 50μm case. The

Table 3 Trochoidal pocket case study: tangential velocity prediction and performance

Case study pocket	Feedrate (mm/min)	Tolerance (microns)	Analysis point	Tangential velocities (mm/min)				
				Measured	CAD/CAM	Proposed prediction	CAD/CAM error	Proposed prediction error
Trochoidal	1000	10	Point 1	1005	1000	1003	-0.50%	0.20%
Trochoidal	1000	50	Trochoid	1005	1000	1004	-0.50%	0.10%
Trochoidal	3000	10	Stepover	2994	3000	2687	+0.20%	10.3%
Trochoidal	3000	50		3014	3000	2985	-0.46%	0.96%
Trochoidal	1000	10	Point 2	938	1000	872	+6.61%	7%
Trochoidal	1000	50	Trochoid	1002	1000	994	-0.20%	0.60%
Trochoidal	3000	10	Main Arc	973	3000	958	+208%	1.5%
Trochoidal	3000	50		2986	3000	2838	+0.47%	4.96%

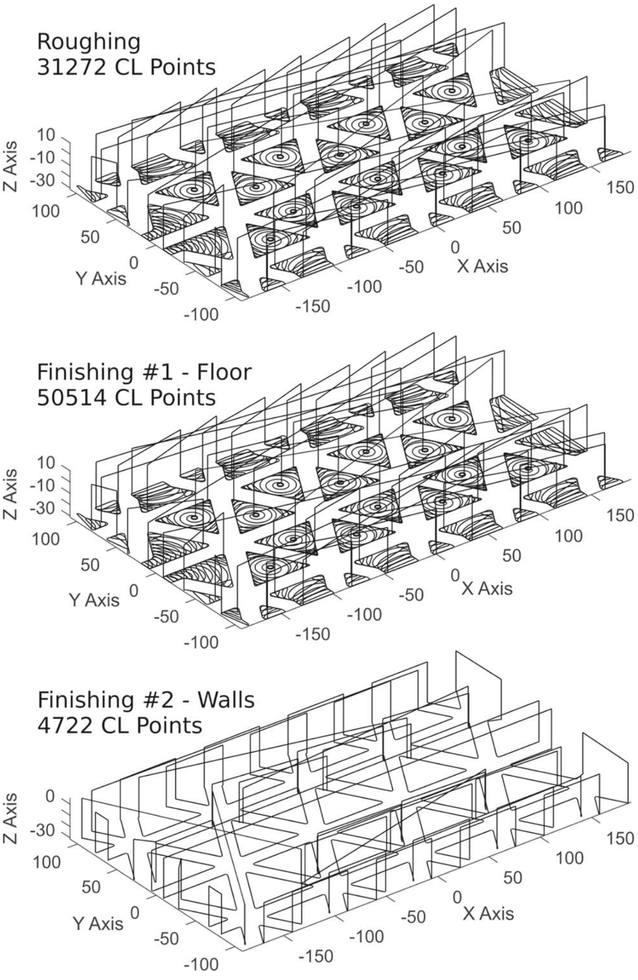


Fig. 14 Aerostructure toolpaths (shown in order of operation)

actual CAD/CAM predicted times do not change as the software does not account for tolerance; the calculation is based upon distance travelled along the toolpath and ideal feedrate. Therefore, as the tolerance is relaxed, the measured cycle time approaches the CAD/CAM case and their prediction becomes more accurate.

The prediction error from the proposed method (as shown in Table 5) ranges from 3.50% over prediction for the 10μm case to 4.69% for the 50μm case. The 20μm case has a prediction error of 5.34% under the measured cycle time which is approximately 10% of the CAD/CAM error (51.49%) for that particular case. The aerostructure case study validates the model for predicting both feedrate and machining cycle times for varying tolerance settings on very complex industrial toolpaths far outperforming the CAM software.

4.3 Case study 4 — accurate cutting force prediction using predicted feedrates

Lastly, the importance of accurate feedrate prediction for virtual machining models is demonstrated. This is realised by estimating cutting forces along the complex trochoidal toolpath shown in Fig. 15. Predicting the cutting forces, considering the complex tool engagements on this toolpath, is realised by adapting the cutting force prediction model presented in [4] with the proposed feedrate prediction method. Readers should refer to [2] and [4] for details of the cutting force model.

To validate the feedrate prediction method with a cutting force model, machining trials were conducted on the 5-axis DMG Mori eVo 40 machining centre fitted with a Heidenhain TNC640 controller. The toolpath, shown in Fig. 15, was designed using NX CAM as a trochoidal pocketing operation. A 40mm × 60mm × 10mm open sided pocket was selected as the test feature as shown in Fig. 16. A 2-fluted 12-mm solid carbide end mill with a HSK-63A tool holder was used. The workpieces were 236mm × 30mm × 6mm aluminium 7075, each held using a Geradi compact grip vice mounted to the dynamometer. A Kistler 9139AA dynamometer and

Table 4 Machining cycle time comparison for aerostructure part case study

Section	Feedrate (mm/min)	Tolerance (microns)	Measured time (s)	CAD/CAM time (s)	Proposed prediction time (s)	CAD/CAM error (%)	Proposed prediction error (%)
Roughing	8000	10	1017.40	332	1032.80	-67.37	1.51
Finish floor	8000	10	543.75	213	606.75	-60.83	11.59
Finish walls	8000	10	133.31	92	114.30	-30.99	-14.26
Roughing	8000	20	752.40	332	711.05	-55.87	-5.50
Finish floor	8000	20	435.01	213	420.85	-51.04	-3.26
Finish walls	8000	20	125.61	92	111.01	-26.76	-11.62
Roughing	8000	50	551.52	332	478.66	-39.80	-13.21
Finish floor	8000	50	334.37	213	368.28	-36.30	10.14
Finish walls	8000	50	115.92	92	107.90	-20.63	-6.92

Table 5 Total machining cycle times and errors for measured, predicted and CAD/CAM

Case	Feedrate (mm/min)	Tolerance (microns)	Total machining cycle times			Errors (%)	
			Measured (s)	CAD/CAM (s)	Proposed prediction (s)	CAD/CAM	Proposed prediction
1	8000	10	1694.46	637	1753.85	-62.41	3.50
2	8000	20	1313.02	637	1242.91	-51.49	-5.34
3	8000	50	1001.81	637	954.84	-36.42	-4.69

a National Instruments USB-6343 multi-channel DAQ was used to acquire cutting force data at 10kHz. The machining centre was connected to a local area network via a RJ45 network cable such that the machine controller data was accessed by two methods. The first using a pre-defined MTConnect datastream through a TCPIP connection at 20Hz and the second using an LSV2 protocol direct to the controller through a TCPIP connection at 111Hz.

The predicted cutting forces during the trochoidal section are shown in Fig. 15. The peak predicted cutting force for the standard feedrate model is 673N compared to 380N for the filtered feedrate model, from the peak measured cutting forces this gives prediction errors of 96.2% and 10.8% respectively. In the cornering section of the toolpath, the peak predicted cutting force for the standard feedrate model is 821N compared to 656N for the filtered feedrate model, from the peak measured cutting forces this gives prediction errors of 37.3% and 9.7% respectively. The validation trials show that the inclusion of an accurate feedrate profile in the cutting force model enables a more accurate prediction of cutting forces for complex toolpaths.

5 Conclusions

A novel method of accurately modelling the trajectory generation of NC systems has been proposed. The main conclusions from this research are as follows:

1. An accurate method of feedrate prediction using FIR-based linear interpolation was introduced.
2. The method was demonstrated along short segmented complex toolpaths.
3. The linear interpolation dynamics and commanded axis kinematic profiles of NC systems were predicted using both 2 and 3 first-order finite impulse response filters with the same time constant.
4. The corner blending behaviour during non-stop interpolation of linear segments was modelled by introducing velocity blending pulses.
5. For the first time, the minimum cornering feedrate that satisfies both the tolerance and machining constraints has been calculated analytically for toolpaths of any geometry.
6. The reduction in machining cycle time by using 3 FIR filters compared to 2 FIR filters was proven analytically.

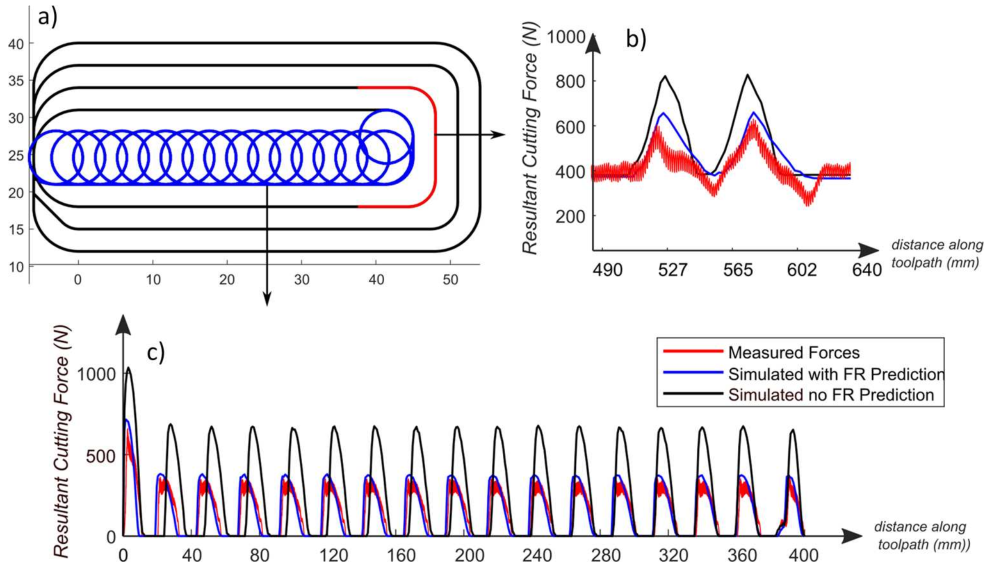


Fig. 15 Simulated and measured resultant cutting forces for a trochoidal pocket

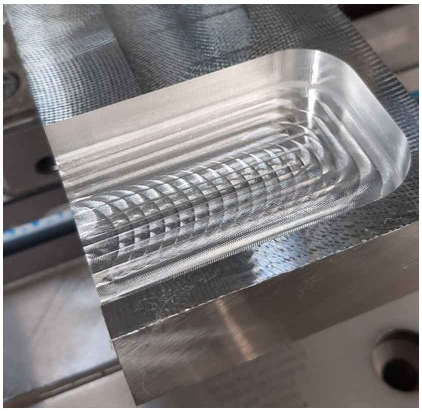


Fig. 16 Machined AL7075 pocket using a trochoidal toolpath

7. The feedrate prediction method was validated experimentally against four different case studies demonstrating industrial 3-axis machining toolpaths.
8. The proposed method demonstrated cycle times can be estimated with >90% accuracy, greatly outperforming CAM-based predictions.
9. The predicted feedrate method was incorporated into a cutting force model, demonstrating an increase in cutting force accuracy for a complex toolpath, and validated experimentally.

Further work will integrate the methods into virtual machining and digital-twin models and extend the method to 5-axis machining. In addition to experimental trials, virtual NC kernels and NC emulators will be used to compare offline prediction methods.

Appendix A: Kinematic equations for 3-FIR case

$$v'(t)=\begin{cases}\frac{F}{6T_1^3}t^3 & 0\le t<T_1\\ \frac{F}{2T_1^3}\left(-\frac{2}{3}t^3+3T_1t^2-3T_1^2t+T_1^3\right) & T_1\le t<2T_1\\ \frac{F}{2T_1^3}\left(\frac{1}{3}t^3-3T_1t^2+9T_1^2t-7T_1^3\right) & 2T_1\le t<3T_1\\ F & 3T_1\le t<T_v\\ \frac{F}{2T_1^3}\left(-\frac{1}{3}t^3+T_vt^2-T_vt+\frac{1}{3}T_v^3+2T_1^3\right) & T_v\le t<T_v+T_1\\ \frac{F}{2T_1^3}\left(\frac{2}{3}t^3-2T_vt^2-3T_1t^2+3T_1^2t+2T_v^2t+6T_vT_1t-\frac{2}{3}T_v^3-3T_v^2T_1-3T_vT_1^2+T_1^3\right) & T_v+T_1\le t<T_v+2T_1\\ \frac{F}{2T_1^3}\left(-\frac{1}{3}t^3+T_vt^2+3T_1t^2-T_v^2t-9T_1^2t-6T_vT_1t+\frac{1}{3}\left(T_v+3T_1\right)^3\right) & T_v+2T_1\le t<T_v+3T_1\end{cases} \tag{31}$$

$$a'(t)=\begin{cases}\frac{F}{2T_1^3}t^2 & 0\le t<T_1\\ \frac{F}{T_1^3}\left(-t^2+3T_1t-\frac{3}{2}T_1^2\right) & T_1\le t<2T_1\\ \frac{F}{T_1^3}\left(\frac{1}{2}t^2-3T_1t+\frac{9}{2}T_1^2\right) & 2T_1\le t<3T_1\\ 0 & 3T_1\le t<T_v\\ \frac{F}{T_1^3}\left(-\frac{1}{2}t^2+T_vt-\frac{1}{2}T_v^2\right) & T_v\le t<T_v+T_1\\ \frac{F}{T_1^3}\left(t^2-2T_vt-3T_1t+\frac{3}{2}T_1^2+T_v^2+3T_vT_1\right) & T_v+T_1\le t<T_v+2T_1\\ \frac{F}{2T_1^3}\left(-t^2+2T_vt+6T_1t-T_v^2-9T_1^2-6T_vT_1\right) & T_v+2T_1\le t<T_v+3T_1\end{cases} \tag{32}$$

$$j'(t)=\begin{cases}\frac{F}{T_1^3}t & 0\le t<T_1\\ \frac{2F}{T_1^3}\left(-t+\frac{3}{2}T_1\right) & T_1\le t<2T_1\\ \frac{F}{T_1^3}\left(t-3T_1\right) & 2T_1\le t<3T_1\\ 0 & 3T_1\le t<T_v\\ \frac{F}{T_1^3}\left(-t+T_v\right) & T_v\le t<T_v+T_1\\ \frac{2F}{T_1^3}\left(t-T_v-\frac{3}{2}T_1\right) & T_v+T_1\le t<T_v+2T_1\\ \frac{F}{T_1^3}\left(-t+T_v+3T_1\right) & T_v+2T_1\le t<T_v+3T_1\end{cases} \tag{33}$$

Appendix B: Kinematic profiles for 3-FIR case

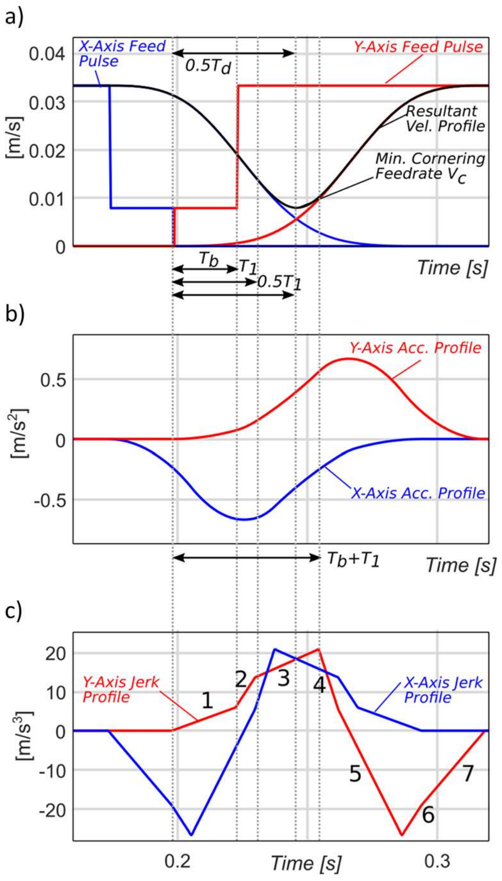


Fig. 17 Velocity, acceleration and jerk profiles generated by blended velocity pulses interpolated 3-FIR filters

Appendix C: Kinematic equations for the 3-FIR filter case (blending pulses)

$$s'(t) = \begin{cases} \frac{\alpha F}{24T_1^3}t^4 & 0 \le t < T_b \\ \frac{2\alpha F}{T_1^3}\left(-\frac{1}{24}t^4 + \frac{1}{4}T_1t^3 - \frac{3}{8}T_1^2t^2\right) + \frac{\alpha F}{2}t - \frac{\alpha F T_1}{8}^2 & T_b \le t < T \\ \frac{F\,(1-3\alpha)}{24T_1^3}\left(-t^4 + 4T_bt^3 - 6T_b^2t^2 + 4T_b^3t - T_b^4\right) + \frac{\alpha F}{24T_1^3}(-8b_1t^3 + 18T_1^2 + 12T_b^2 + \ldots & \\ \ldots - 36T_1T_b + 24T_b)t^2 + \left(8T_b^2 - 12T_1^3\right)t + 3T_1^4 - 18T_b^4 + 24T_1T_b^3 - 16Tb^3b_1 & T_1 \le t < T_1 + T_b \end{cases} \tag{34}$$

$$v'(t) = \begin{cases} \frac{\alpha F}{6T_1^3}t^3 & 0 \le t < T_b \\ \frac{2\alpha F}{T_1^3}\left(-\frac{1}{6}t^3 + \frac{3}{4}T_1t^2 - \frac{3}{4}T_1^2t\right) + \frac{\alpha F}{2} & T_b \le t < T_1 \\ \frac{F(1-3\alpha)}{6T_1^3}\left(t^3 - 3T_bt^2 + 3T_b^2t + T_b^3\right) + \frac{\alpha F}{T_1^3}\left(\frac{1}{2}T_1^3 - \frac{1}{3}T_b^3 - (\frac{3}{2}T_1^2 - 3T_1T_b + T_b^2\right)t + \ldots & \\ \ldots + b_1t^2 - 2T_bb_1t) & T_1 \le t < T_1 + T_b \end{cases} \tag{35}$$

$$a'(t) = \begin{cases} \frac{\alpha F}{T_1^3} t & 0 \leq t < T_b \\ \frac{2\alpha F}{T_1^3} \left(-\frac{1}{2} t^2 + \frac{3}{2} T_1 t - \frac{3}{4} T_1^2 \right) & T_b \leq t < T_1 \\ \frac{2\alpha F}{T_1^3} \left(\frac{3}{2} T_1^2 - \frac{3}{2} T_1 T_b + \frac{1}{2} T_b^2 - T_b b_1 + b_1 \right) - \frac{F(1-3\alpha)}{T_1^3} \left(\frac{1}{2} T_b^2 - \left(T_b t - \frac{1}{2} t^2 \right) \right) & T_1 \leq t < T_1 + T_b \end{cases} \tag{36}$$

$$j'(t) = \begin{cases} \frac{\alpha F}{T_1^3} t & 0 \leq t < T_b \\ \frac{\alpha F}{T_1^3} \left(-t + \frac{3}{2} T_1 \right) & T_b \leq t < T_1 \\ \frac{F(1-3\alpha)}{T_1^3} (t - T_b) + \frac{2\alpha F}{T_1^3} \left(\frac{3}{2} T_1 - T_b \right) & T_1 \leq t < T_1 + T_b \end{cases} \tag{37}$$

where $b_1 = \frac{3}{2} T_1 - T_b$.

The maximum TCP error occurs at $t = \frac{T_d}{2} = \frac{3}{2} T_1$ for the 3 first-order FIR filter cases; the interpolated axis velocity and displacement are defined as Eqs. 38 and 39 respectively:

$$v' = \frac{F}{48\, {T_1}^3} \left(36\, T_1 T_b^2 - 54\, T_1^2\, T_b - 3\, T_1^3\, \alpha + 8\, T_b^3\, \alpha \right.$$
$$\left. +27\, T_1^3 - 8\, T_b^3 - 36\, T_1\, T_b^2\, \alpha + 54\, T_1^2\, T_b\, \alpha \right) \tag{38}$$

$$s' = -\frac{F}{384 T_1^3} (96\, T_1\, T_b^3 + 216\, T_1^3\, T_b + 3\, T_1^4\, \alpha + 16\, T_b^4\, \alpha$$
$$-81\, T_1^4 - -16\, T_b^4 - 216\, T_1^2\, T_b^2 + ...$$
$$... + 216\, T_1^2\, T_b^2\, \alpha - 96\, T_1\, T_b^3\, \alpha - 216\, T_1^3\, T_b\, \alpha) \tag{39}$$

Using Eq. 14, Eq. 38 and Eq. 39 can be expressed in terms of F and α as:

$$v' = \frac{F}{48} \left(-8\alpha^4 - 4\alpha^3 + 6\alpha^2 + 29\alpha + 1 \right) \tag{40}$$

$$s' = \frac{F}{384} T_1 \left(-16\alpha^5 - 16\alpha^4 + 8\alpha^3 + 16\alpha^2 + 85\alpha + 1 \right) \tag{41}$$

Solving Eq. 27 with Eq. 39 results in the maximum TCP error for the 3-FIR filter case as follows:

$$\varepsilon_{TCP} = \frac{\sqrt{2}}{384} \sqrt{F^2\, {T_1}^2\, (\cos\theta_{TCP} + 1) \left(-16\, \alpha^5 - 16\, \alpha^4 + 8\, \alpha^3 + 16\, \alpha^2 + 85\, \alpha + 1 \right)^2} \tag{42}$$

Funding This research was supported by the Advanced Manufacturing Research Centre members and EPSRC (grant EP/L016257/1).

Availability of data and materials Data are available from Rob Ward with the permission of the University of Sheffield AMRC. The data that support the findings of this study are available from the corresponding author, RW, upon reasonable request.

Declarations

Consent for publication The authors declare that they all consent to publication.

Conflict of interest The authors declare no competing interests.

References

1. Altintas Y (2016) Virtual high performance machining 46:372–378
2. Armendia M, Ghassempouri M, Ozturk E, Peysson F (2019) Twin-control, a digital twin approach to improve machine tools lifecycle, 1st ed Springer

3. Layegh K. SE, Erdim H, Lazoglu I (2012) Offline force control and feedrate scheduling for complex free form surfaces in 5-axis milling, Procedia CIRP, vol. 1, no. 1, pp. 96–101. [Online]. Available: https://doi.org/10.1016/j.procir.2012.04.015
4. Berglind L, Plakhotnik D, Ozturk E (2017) Discrete cutting force model for 5-axis milling with arbitrary engagement and feed direction, Procedia CIRP, vol. 58, no. Cl, pp. 445–450. [Online]. Available: https://doi.org/10.1016/j.procir.2017.03.250
5. Nie Z, Lynn R, Tucker T, Kurfess T (2009) Voxel-based analysis and modeling of MRR computational accuracy in milling process, CIRP. J. Manuf. Sci. Technol. 27:78–92
6. Sencer B, Altintas Y (2009) Modeling and control of contouring errors for Five-Axis machine tools—part II: precision contour controller design. J. Manuf. Sci. Eng. 131(3):31007
7. Altintas, Y, Kersting, P, Biermann, D, Budak, E, Denkena, B, and Lazoglu, I, Virtual process systems for part machining operations, CIRP Ann., vol. 63, no. 2, pp. 585–605, jan 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0007850614001899
8. Altintas, Y, Manufacturing automation. Cambridge: Cambridge University Press, 2011. [Online]. Available: http://ebooks.cambridge.org/ref/id/CBO9780511843723
9. Choi YK, Banerjee A (2007) Tool path generation and tolerance analysis for free-form surfaces. Int. J. Mach. Tools Manuf. 47(3-4):689–696
10. Sencer B, Altintas YÃ, Croft E (2008) Feed optimization for five-axis CNC machine tools with drive constraints. vol 48, pp 733–745
11. Altintas, Y and Tulsuan, S, Prediction of part machining cycle times via virtual CNC,CIRP Ann., vol. 64, pp. 361–364, 2015. [Online]. Available: https://doi.org/10.1016/j.cirp.2015.04.100
12. Erkorkmaz K, Altintas Y (2001) High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation 41:1323–1345
13. Beudaert, X, Lavernhe, S, and Tournier, C, Feedrate interpolation with axis jerk constraints on 5-axis NURBS and G1 tool path, Int. J. Mach. Tools Manuf., vol. 57, pp. 73–82, 2012. [Online]. Available: https://doi.org/10.1016/j.ijmachtools.2012.02.005
14. Jeong SY, Choi YJ, Park PG, Choi SG (2005) Jerk limited velocity profile generation for high speed industrial robot trajectories. IFAC Proc 16:595–600
15. Barre PJ, Bearee R, Borne P, Dumetz E (2005) Influence of a jerk controlled movement law on the vibratory behaviour of high-dynamics systems. J Intell Robot Syst Theory Appl 42(3):275–293
16. Erkorkmaz K, Yeung CH, Altintas Y (2006) Virtual CNC system. Part II. High speed contouring application. Int J Mach Tools Manuf 46(10):1124–1138
17. Tulsyan, S and Altintas, Y, Cycle time prediction for milling free-form surfaces by considering dynamics of CNC systems
18. Tajima, S, Sencer, B, and Shamoto, E, Accurate interpolation of machining tool-paths based on FIR filtering, Precis Eng, vol. 52, no. August 2017, pp. 332–344, 2018. [Online]. Available: https://doi.org/10.1016/j.precisioneng.2018.01.016
19. Heidenhain (2017) TNC 640, HSCI The contouring control for milling-turning and machining centers, Inf. Mach. Tool Build. Man., no June
20. Heidenhain (2011) Machining accuracy of machine tools. Heidenhain White Pap., no September
21. Mitsubishi Electric, M700V/M70V Series Programming Manual (Machining Center System)
22. Siemens (2020) Sinumerik One Dynamics, software version NCU-SW. vol 6, p 14
23. Tajima, S and Sencer, B, Accurate real-time interpolation of 5-axis tool-paths with local corner smoothing, Int J Mach Tools Manuf, vol 142, no February, pp 1–15, 2019. [Online]. Available: https://doi.org/10.1016/j.ijmachtools.2019.04.005
24. Tajima, S and Sencer, B, Real-time trajectory generation for 5-axis machine tools with singularity avoidance, CIRP Ann., vol. 00, pp. 50–53, 2020. [Online]. Available: https://doi.org/10.1016/j.cirp.2020.04.050
25. Lambrechts P, Boerlage M, Steinbuch M (2005) Trajectory planning and feedforward design for electromechanical motion systems. Control Eng Pract 13(2):145–157
26. Smith, SW, Digital signal processing: a practical guide for engineers and scientists, Smith, SWBTDSP, Ed. Boston: Newnes, 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780750674447500522
27. Biagiotti, L and Melchiorri, C, FIR filters for online trajectory planning with time- and frequency-domain specifications, Control Eng Pract, vol 20, no 12, pp 1385–1399, 2012. [Online]. Available: https://doi.org/10.1016/j.conengprac.2012.08.005
28. Blinchikoff HJ, Zverer AI (1976) Filtering in the Time and Frequency Domains. Wiley, New York
29. Ur-Rehman, R, Richterich, C, Arntz, K, and Klocke, F, Numerical techniques for CAM strategies for machining of mould and die, In Proc. 36th Int. MATADOR Conf., Hinduja, S and Li, L, Eds. London: Springer London, 2010, pp. 259–263
30. Siemens PLM Software, Siemens NX 10

Affiliations

Rob Ward[1] · **Burak Sencer**[2] · **Bryn Jones**[3] · **Erdem Ozturk**[4]

Burak Sencer
burak.sencer@oregonstate.edu

Bryn Jones
b.l.jones@sheffield.ac.uk

Erdem Ozturk
e.ozturk@amrc.co.uk

[1] Industrial Doctorate Centre in Machining Science, Advanced Manufacturing Research Centre, University of Sheffield, Rotherham, S60 5TZ, UK

[2] Mechanical Engineering Department, Oregon State University, Corvallis, OR, USA

[3] Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, S1 3JD, England

[4] Advanced Manufacturing Research Centre, University of Sheffield, Rotherham, S60 5TZ, UK