



University of HUDDERSFIELD

University of Huddersfield Repository

Ford, Derek G., Castaneda, Veimar Yobany Moreno, Longstaff, Andrew P., Pislaru, Crinela and Myers, Alan

Computer numerical control vertical machining centre feed drive modelling using the transmission line technique

Original Citation

Ford, Derek G., Castaneda, Veimar Yobany Moreno, Longstaff, Andrew P., Pislaru, Crinela and Myers, Alan (2014) Computer numerical control vertical machining centre feed drive modelling using the transmission line technique. *Proceedings of the Institution of Mechanical Engineers Part C Journal of Mechanical Engineering Science*. pp. 1-22. ISSN 09544062

This version is available at <http://eprints.hud.ac.uk/id/eprint/21509/>

The University Repository is a digital collection of the research output of the University, available on Open Access. Copyright and Moral Rights for the items on this site are retained by the individual author and/or other copyright owners. Users may access full items free of charge; copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational or not-for-profit purposes without prior permission or charge, provided:

- The authors, title and full bibliographic details is credited in any copy;
- A hyperlink and/or URL is included for the original metadata page; and
- The content is not changed in any way.

For more information, including our policy and submission procedure, please contact the Repository Team at: E.mailbox@hud.ac.uk.

<http://eprints.hud.ac.uk/>

Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science

<http://pic.sagepub.com/>

Computer numerical control vertical machining centre feed drive modelling using the transmission line technique

DG Ford, VYM Castaneda, AP Longstaff, C Pislaru and A Myers

Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science published online 28 August 2014

DOI: 10.1177/0954406214546540

The online version of this article can be found at:

<http://pic.sagepub.com/content/early/2014/08/08/0954406214546540>

Published by:



<http://www.sagepublications.com>

On behalf of:



[Institution of Mechanical Engineers](http://www.institutionofmechanicalengineers.org)

Additional services and information for *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* can be found at:

Open Access: Immediate free access via SAGE Choice

Email Alerts: <http://pic.sagepub.com/cgi/alerts>

Subscriptions: <http://pic.sagepub.com/subscriptions>

Reprints: <http://www.sagepub.com/journalsReprints.nav>

Permissions: <http://www.sagepub.com/journalsPermissions.nav>

Citations: <http://pic.sagepub.com/content/early/2014/08/08/0954406214546540.refs.html>

>> [OnlineFirst Version of Record](#) - Aug 28, 2014

[What is This?](#)

Computer numerical control vertical machining centre feed drive modelling using the transmission line technique

Proc IMechE Part C:
J Mechanical Engineering Science
0(0) 1–22
© IMechE 2014
Reprints and permissions:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/0954406214546540
pic.sagepub.com



DG Ford, VYM Castaneda, AP Longstaff, C Pislaru and A Myers

Abstract

This study presents a novel application of the Transmission Line Matrix Method (TLM) for the modelling of the dynamic behaviour of non-linear hybrid systems for computer numerical control (CNC) machine tool drives. The application of the TLM technique implies the dividing of the ball-screw shaft into a number of identical elements in order to achieve the synchronisation of events in the simulation, and to provide an acceptable resolution according to the maximum frequency of interest. This entails the use of a high performance computing system with due consideration to the small time steps being applied in the simulation. Generally, the analysis of torsion and axial dynamic effects on a shaft implies the development of independent simulated models. This study presents a new procedure for the modelling of a ball-screw shaft by the synchronisation of the axial and torsion dynamics into the same model. The model parameters were obtained with equipments such as laser interferometer, ball bar, electronic levels, signal acquisition systems, etc. The MTLM models for single and two-axis configurations have been simulated and matches well with the measured responses of machines. The new modelling approach designated the Modified Transmission Line Method (MTLM) extends the TLM approach retaining all its inherent qualities but gives improved convergence and processing speeds. Further work since, not the subject of this paper, have identified its potential for real-time application.

Keywords

Machine Tool, feed drive modelling, TLM, MTLM, model optimisation

Date received: 30 July 2013; accepted: 9 July 2014

Background

Ford¹ proposed that the design process of any high precision computer numerical control (CNC) machine tool must embrace error avoidance, error measurement, and error compensation techniques in order to achieve an ever increasing demand for greater accuracy and therefore a more stringent performance specification. The measurement of the repeatable time and spatial errors can allow compensation methods to be applied to the machine for correction of those errors provided sufficient resolution has been allowed for in the design, with the absolute limit on accuracy for a particular machine being defined by its measured repeatability figures and the responsiveness of its compensating axes. The main areas of concern affecting component accuracy are environmental effects, user effects and the machine tool static and dynamic accuracy. The sources of error confined to a machine tool are geometrical inaccuracies, thermally induced errors and load errors.

Machine tool feed drive modelling and system identification

Pislaru et al.² stated that the modelling used for designing high performance CNC machine tool feed drives should accurately represent the system consisting of three cascaded closed loops for acceleration, velocity and position control. They utilised a modular approach and a hybrid modelling technique which included for distributed load, explicit damping coefficients, backlash and friction. The model was a combination of distributed and lumped elements described by partial differential equations and ordinary

Centre for Precision Engineering, University of Huddersfield, Huddersfield, UK

Corresponding author:

DG Ford, University of Huddersfield, Queensgate, Huddersfield HD1 3DH, UK.

Email: d.ford@hud.ac.uk

differential equations as proposed by Bartlett and Whalley.³

Holroyd et al.⁴ modelled the dynamic behaviour of a ball-screw with moving nut using a finite element approach. The ball-screw was divided into a large number of elements and contact/boundary conditions for each element and adjacent ones were studied. An important conclusion was that the natural frequencies of the ball-screw system vary in time due to two causes: (a) the lateral restraint produced by the nut when the screw was transversally vibrated and (b) the relation between screw torsion, axial motion and worktable/saddle tilt.

Transmission line matrix method

The transmission line matrix (TLM) method belongs to the general class of differential time domain numerical modelling methods. It is used to solve time-dependent or transient problems, thus involving ordinary and partial differential equations (PDE).

The application of TLM to the modelling of fluid systems by Boucher and Kitsios⁵ and mechanical systems by Partridge et al.⁶ implies the same representation used in the hybrid approach by Pislaru,² i.e. lumped parameter modules, components with localised dynamic effects and distributed parameter modules. The difference resides on the fact that transmission line equations introduce natural time delays between components due to the wave propagation speed in a transmission line. Therefore, distributed components must be divided into a number of identical elements in order to achieve the synchronisation of events in the simulation, and to produce acceptable resolution according to the maximum frequency of the simulation. This characteristic gives the possibility to include the effect of the nut movement suggested in Holroyd's et al.⁴ approach.

Johns and Beurle⁷ described the two equivalent circuits used in the TLM technique: the stub and the link circuits. TLM links are two-port one-dimensional building blocks that can be used for one-, two- or three-dimensional modelling. TLM stubs are one-port units which can be used for solving circuits and equations, and are used in multi-dimensional modelling to complement TLM links. Links are used for modelling distributed parameter elements and stubs for lumped parameter elements.

Christopoulos⁸ stated that any electrical circuit could be represented as a network of transmission line circuits by simply replacing the reactive elements with corresponding stubs. Variables such as voltage and current are regarded as discrete pulses bouncing at a velocity to and from nodes of these stubs at each time step. The voltage and current in each element are determined from the incident and reflected pulses in each stub. He demonstrated the transmission line and its Thevenin equivalent.

The MTLM stub, taking into account changes of sample time was applied to the modelling of a rotor shaft arrangement by Partridge et al.,⁶ where Castaneda⁹ demonstrated improved robustness and accuracy of the simulation. A reduction of 40% in the number of mathematical operations and almost 35% reduction in the mean square modelling error figures were achieved. This method of approach was utilised for this application.

Single axis CNC test rig

The single-axis test rig (Figure 1a) represents the y-axis of a vertical machining centre (VMC). The single axis is fitted with a Heidenhain TNC-426PB motion controller, a SIMODRIVE-611 Siemens inverter, and a ball-screw arrangement directly coupled to a permanent magnet synchronous motor.

The Heidenhain¹⁰ TNC-426PB motion controller offers digital control of functions such as interpolation, position control, speed control, current control and provides PWM signals to the inverter unit. The control of each axis is implemented with an algorithm following the principle of cascade control in conjunction with velocity and acceleration feed forward.

The SIMODRIVE-611 consists of a common feed module that provides the DC voltage link from the power supply mains and a set of drive modules that activate each motor. In the case of the test rig, the drive module consists of a power module (inverter) and an interface card that communicates with the TNC-426PB motion controller.

The nut of the ball screw system is preloaded and the screw shaft is mounted on preloaded bearings in a fixed-fixed configuration.

The model of the axis feed drive is built by interconnecting the TLM models of the rig elements. The dynamics acting on the axis feed drive are defined by the interrelation of three blocks: The motion controller, the inverter and motor electrical equations, and motor mechanical equations and mechanical transmission elements as illustrated in Figure 1(b). Table 1 contains the test rig specification.

TLM method applied to CNC machine tool feed drive

Motion controller

Heidenhain¹⁰ operates with position and velocity control in cascade with feed forward capability as shown in Figures 2 and 3. Seborg et al.¹¹ concluded that feed forward control improves trajectory tracking compensating for the effect of disturbances before they affect the response of the controlled system. The velocity feed forward path connects the reference position to the velocity loop through the gain, k_{ff} (Figure 3). When the position profile changes, the velocity feed forward transfers immediately that change to the

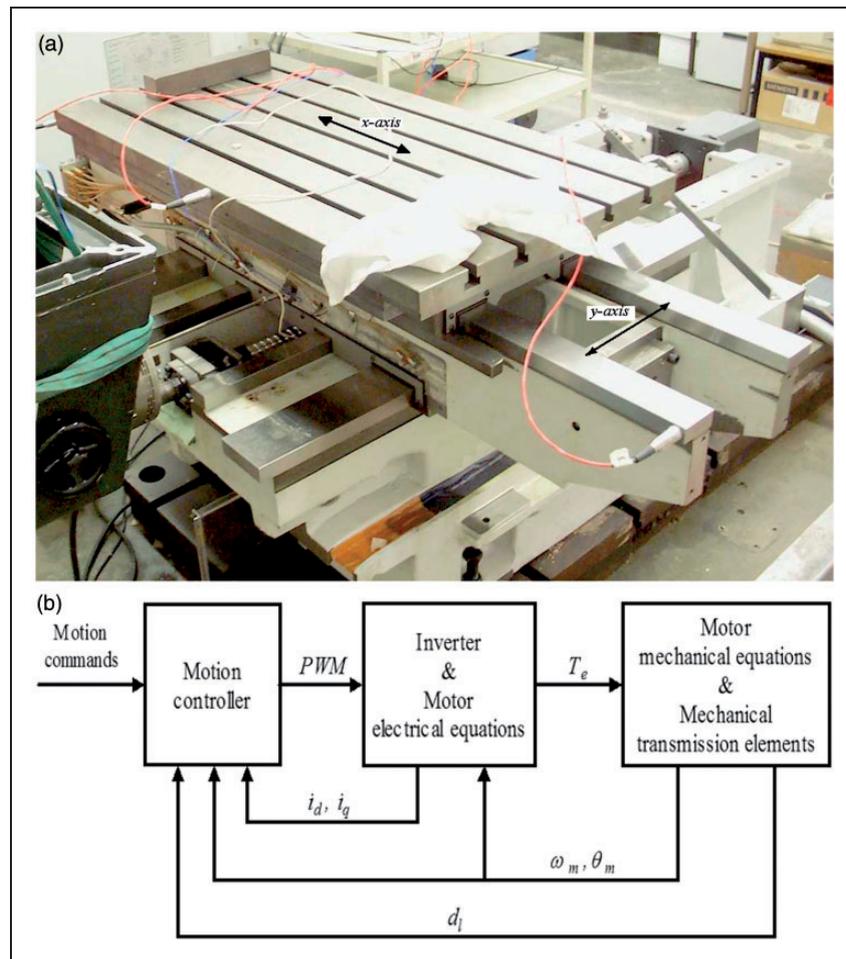


Figure 1. (a) Test rig and (b) test rig block model.

Table 1. Test rig specification.

Motion controller	HEIDENHAIN TNC 426PB (280 476-24)
Electrical drive	SIEMENS Simodrive 611 (6SN1123-1AA00-0CA1)
Motor	SIEMENS IFT6082-1AF7-1AG1
Guide ways	THK SNS45-LC
Front bearing	Double Row Angular Contact Ball Bearing ZKLF 3080.2RS PE
Rear bearing	Double Row Angular Contact Ball Bearing ZKLF 3080.2RS PE
Ballscrew	THK BIF (BNFN) 4016-5 RRG0S-1346(1065)L-C3-E
Coupling	ROBA – ES 28 940.000
Rotary encoder	HEIDENHAIN ERN 1387
Linear encoder	HEIDENHAIN AE LS486C ML620
Transverse	500 mm
Rapid Traverses	30,000 mm/min
Feed Rates	100 – 12000 mm/min
Table Size	115 mm × 58 mm
Table mass	312 kg
Saddle mass	524 kg
Load mass	854 kg

velocity command. This speeds up the system response, thus not relying solely on the position loop. The primary shortcoming of velocity feed forward is that it induces overshoot when changes on the velocity direction occur. Acceleration feed forward eliminates the overshoot caused by velocity feed forward without reducing loop gains. The basic principle of feed forward control is to inject the position and velocity set points and their correspondent first derivatives at appropriate points in the control loops. Heinemann and Papiernik¹² demonstrated that the following errors are minimised, thus achieving a significant increase in the dynamic response to position and velocity set point changes.

Altinas¹³ demonstrated that the interpolator (Figure 2) calculates a velocity profile $v(t)$, according to defined motion commands: feed rate (f_r), maximum acceleration (a_{max}), maximum jerk (j_{max}), and desired displacement (d). The velocity profile is then used to generate the set of reference positions in order to achieve a smooth velocity transition.

Koren¹⁴ and Bullock¹⁵ demonstrated that linear and circular interpolation, respectively, are performed to keep the tool path velocity constant at the given feed rate along a straight line in a plane of motion.

order of the filter and the tolerance for contour transitions.

- The PT₂, second order element was used to include a delay in the reference current (i_{ref}) to damp the frequency interference oscillations (circa: 0.3–2 ms).
- Sliding friction was compensated within the range of the velocity controller by compensating the sliding friction at low velocity and at the rated velocity of the motor. The compensation at low velocity was achieved by feeding forward the reference current value (measured at approximately 10 rpm) at every change in direction. The compensation at the rated velocity was achieved by feeding forward the current $i_{ff_{rv}}$ according to the value of the reference velocity (equation (2)). A delay filter was included to prevent overcompensation when the traverse direction was reversed at high feed rates.

$$i_{ff}(k) = i_{ff_{l}} + i_{ff_{rv}}(k) \quad (1)$$

$$i_{ff_{rv}}(k) = \begin{cases} i_{ffm} & v_{ref}(k) \geq i_{ffm} \\ v_{ref}(k)k_{iff} & \text{for } |v_{ref}(k)| < i_{ffm} \\ -i_{ffm} & v_{ref}(k) \leq -i_{ffm} \end{cases} \quad (2)$$

where

i_{ffm} is the reference current measured at the rated velocity of the motor; $i_{ff_{l}}$ is the reference current at 10 rpm and k_{iff} is the scaling factor.

Position controller with velocity feed forward. The position controller evaluates the difference between the reference and actual position value d_{act} (from the rotary or linear encoder) to calculate the position error d_e . A reference velocity value (v_{ref}) is then generated every t_s seconds according to the following

$$v_{ref} = d_e k_v + v_{ff} \quad (3)$$

$$d_e = d_{ref} - d_{act} \quad (4)$$

$$v_{ff} = k_{rff} \frac{d}{dt} d_{ref} \quad (5)$$

The signal d_{act} is calculated by equation (6) if the linear encoder is used as a position feedback system. Equation (7) gives the value for d_{act} when the rotary encoder is used instead

$$d_{act}(k) = d_l(k) \quad (6)$$

$$d_{act}(k) = k_b \theta_m(k) \quad (7)$$

where

k_b is the force to torque constant of the ball screw.

The TLM model for this module is represented by equations (3) to (7) for a sample time $t_s = 3$ ms.

By a combination of Christopoulos⁸ and Castaneda⁹ theory

$$v_{ff}(k) = d_{ref}(k)Z_{vff} + E_{vff}^i(k) \quad (8)$$

$$E_{vff}^i(k+1) = -d_{ref}(k)Z_{vff} \quad (9)$$

$$v_{ref}(k) = k_v d_e(k) + v_{ff}(k) \quad (10)$$

$$d_e(k) = d_{ref}(k) - d_{act}(k) \quad (11)$$

where $Z_{vff} = k_{vff}/t_s$ and Z_{vff} is the characteristic impedance of the stub and $E_{vff}^i(k)$ is the incident pulse.

Velocity controller with acceleration feed forward. The velocity controller evaluates the difference between the reference and actual velocity value (v_{act}) measured with a rotary encoder attached to the motor. A reference current value (i_{ref}) is generated every t_s s by a Proportional-Integral-Differential (PID) strategy.

Acceleration feed-forward is used in parallel with the velocity controller in order to minimise the spikes caused by changes in velocity direction. A signal i_{ht} (holding current) is added to the acceleration feed forward to counterbalance the axis load when the axis is in the vertical position.

Equations (13) to (18) represent the TLM model for this module ($t_s = 0.6$ ms). The following relationship calculates the reference current signal

$$i_{ref} = i_p + i_{int} + i_{der} + a_{ff} + i_{ht} \quad (12)$$

where

$$i_p = k_p v_e \quad (13)$$

$$i_{int} = k_i \int v_e dt \quad (14)$$

$$i_{der} = k_d \frac{dv_e}{dt} \quad (15)$$

$$a_{ff} = k_{aff} \frac{dv_{ff}}{dt} \quad (16)$$

$$v_e = v_{ref} - v_{act} \quad (17)$$

Here k_p , k_i and k_d are the proportional, integral and derivative gains of the velocity error v_e , respectively. k_{aff} is the acceleration feed forward gain. Applying the Castaneda⁹ TLM transform (equations (19) to (26)) for the cycle time t_s gives

$$i_{ref}(k) = i_p(k) + i_{int}(k) + i_{der}(k) + a_{ff}(k) \quad (18)$$

$$i_p(k) = k_p v_e(k) \quad (19)$$

$$i_{\text{int}}(k) = Z_i v_e(k) + E_i^i(k) \quad (20)$$

$$E_i^i(k+1) = i_{\text{int}}(k) \quad (21)$$

$$i_{\text{der}}(k) = Z_d v_e(k) + E_d^i(k) \quad (22)$$

$$E_d^i(k+1) = -Z_d v_e(k) \quad (23)$$

$$a_{\text{ffi}}(k) = Z_{\text{aff}} v_{\text{ff}}(k) + E_{\text{aff}}^i(k) \quad (24)$$

$$E_{\text{aff}}^i(k+1) = -Z_{\text{aff}} v_{\text{ff}}(k) \quad (25)$$

$$v_e(k) = v_{\text{ref}}(k) - v_{\text{act}}(k) \quad (26)$$

where

$$Z_i = k_i t_s; \quad Z_d = k_d / t_s \quad \text{and} \quad Z_{\text{aff}} = k_{\text{aff}} / t_s$$

Band-stop filter. Heidenhain¹⁷ states that a filter is generally used in the velocity feedback to damp the fundamental frequency of the control system, when it is higher than 500 Hz. A first-order low-pass filter is used when the oscillation frequency is between 500 and 700 Hz. A second-order low-pass filter is used if the oscillation frequency is higher than 700 Hz.

When the controlled system is insufficiently damped, it will be impossible to attain a sufficiently short settling time without inducing oscillations in the step response of the velocity controller. The step response will oscillate even with a low proportional factor. A second-order lag element (PT₂) is used to include a delay in the reference current (i_{ref}) to damp the frequency interference oscillations.

A band-rejection filter is included in series with the PT₂ element to damp oscillations that cannot be compensated by the differential factor of the velocity controller, the PT₂ element or the low-pass filter.

Mathworks¹⁸ state these digital filters are generally modelled as implementations of the standard difference equation

$$\begin{aligned} a(1)i_{q\text{ref}}(k) &= b(1)i_{\text{ref}}(k) + b(2)i_{\text{ref}}(k-1) + \dots \\ &+ b(2N+1)i_{\text{ref}}(k-2N) - a(2)i_{q\text{ref}}(k-1) - \dots \\ &- a(2N+1)i_{q\text{ref}}(k-2N) \end{aligned} \quad (27)$$

where

b and a represent the numerator and denominator coefficients of the filter and N is the order of the filter.

The band-stop filter was implemented as indicated by Mathworks¹⁹ in the transposed direct-form II structure of equation (27). This is a canonical form that has the minimum number of delay elements.

At sample k , the routine computes the difference equations as follows

$$i_{q\text{ref}}(k) = \text{num}(1)i_{\text{ref}}(k) + \text{zz}_1(k-1) \quad (28)$$

$$\text{zz}_1(k) = \text{num}(2)i_{\text{ref}}(k) + \text{zz}_2(k-1) - \text{den}(2)i_{q\text{ref}}(k) \quad (29)$$

$$\begin{aligned} \text{zz}_{\text{len}-3}(k) &= \text{num}(\text{len}-2)i_{\text{ref}}(k) + \text{zz}_{\text{len}-2}(k-1) \\ &- \text{den}(\text{len}-2)i_{q\text{ref}}(k) \end{aligned} \quad (30)$$

$$\begin{aligned} \text{zz}_{\text{len}-2}(k) &= \text{num}(\text{len}0)i_{\text{ref}}(k) + \text{zz}_{\text{len}0}(k-1) \\ &- \text{den}(\text{len}0)i_{q\text{ref}}(k) \end{aligned} \quad (31)$$

$$\text{zz}_{\text{len}0}(k) = \text{num}(\text{len})i_{\text{ref}}(k) - \text{den}(\text{len})i_{q\text{ref}}(k) \quad (32)$$

$$\text{len} = \text{len}0 + 1 \quad (33)$$

where $\text{len}0$ is the filter order, and num and den represent the numerator and denominator filter coefficients. The delay outputs $\text{zz}_i(1)$, $i = 1, \dots, \text{len}0$ are initialised to 0. This is equivalent to assuming that both past inputs and outputs are zero.

Current controller. The current controller is an implementation of the model when the integral term k_{ci} is equal to zero.

Castaneda⁹ developed the model shown and converted to the TLM transform of equations which are presented as

$$i_{de}(k) = i_{d\text{ref}}(k) - i_{d\text{act}}(k) \quad (34)$$

$$e_{d\text{ref}}(k) = k_{cp}i_{de}(k) \quad (35)$$

$$i_{qe}(k) = i_{q\text{ref}}(k) - i_{q\text{act}}(k) \quad (36)$$

$$e_{q\text{ref}}(k) = k_{cp}i_{qe}(k) \quad (37)$$

Inverter and motor (electrical)

Moreton²⁰ describes the voltage relationships and provided the general equation for a permanent magnet synchronous motor (PMSM). Simon²¹ explained the Clarke /Park transformation and their inverse relating the vector currents used for the generation of electromagnetic torque (Figure 4a).

The equations of performance for a PMSM in the d - q coordinate system representation are

$$L_d \frac{d}{dt} i_d = e_d - R i_d + L_q \omega_e i_q \quad (38)$$

$$L_q \frac{d}{dt} i_q = e_q - R i_q - L_d \omega_e i_d - \lambda \omega_e \quad (39)$$

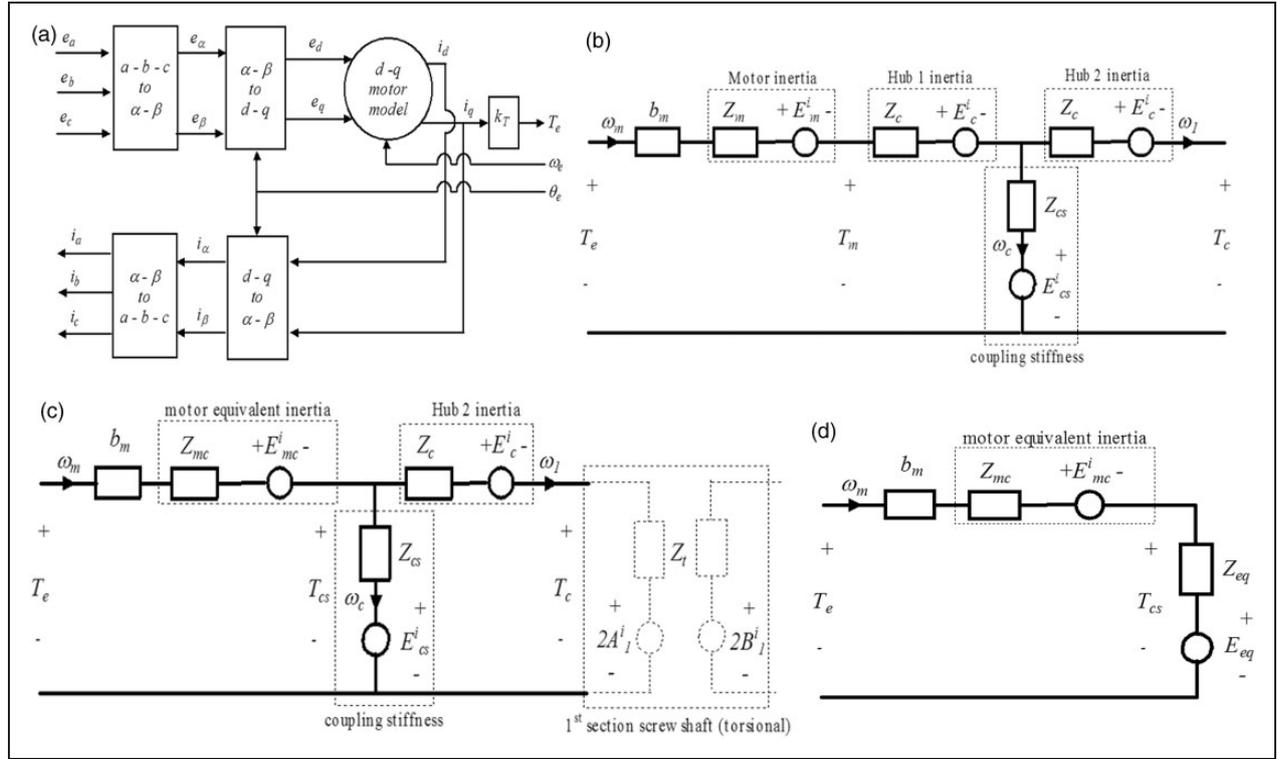


Figure 4. TLM model for the motor and coupling. (a) Motor electrical equations, (b) TLM model of the motor and coupling, (c) reduced TLM model and (d) Thevenin equivalent circuit.

in which λ signifies the amplitude of the flux induced by the permanent magnets of the rotor in the stator phases; L_d and L_q are the d and q axis inductances. Subscripts d and q are used to identify the direct and quadrature currents/voltages, respectively. The generated electromagnetic torque T_e is defined as

$$T_e = 1.5p(\lambda i_q + (L_d - L_q)i_d i_q) \quad (40)$$

if the current $i_d \rightarrow 0$, $T_e \rightarrow 1.5p\lambda i_q$; hence, the generated electromagnetic torque can be approximated using the torque constant of the motor, k_T

$$T_e = k_T i_q \quad (41)$$

Castaneda⁹ considered a motor with a round rotor and a simulation time step t_s , and derived the TLM transforms of equations (38), (39) and (41) giving

$$Z_L i_d(k) + E_d^i(k) = e_d(k) - R i_d(k) + L \omega_e(k) i_q(k) \quad (42)$$

$$\text{where : } Z_L = L/t_s \quad (43)$$

$$E_d^i(k+1) = -i_d(k) Z_L \quad (44)$$

$$Z_L i_q(k) + E_q^i(k) = e_q(k) - R i_q(k) - \omega_e(k)(L i_d(k) + \lambda) \quad (45)$$

$$E_q^i(k+1) = -i_q(k) Z_L \quad (46)$$

$$T_e(k) = k_T i_q(k) \quad (47)$$

Rearranging equations (43) and (45) provides

$$e_d(k) - E_d^i(k) - (Z_L + R) i_d(k) = -\omega_e(k) L i_q(k) \quad (48)$$

$$e_q(k) - E_q^i(k) - (Z_L + R) i_q(k) = \omega_e(k)(L i_d(k) + \lambda) \quad (49)$$

For simulation purposes, it can be assumed that the terms on the left side of equations (48) and (49) are the resultants of the operations on the right side of the equations. Thus, these equations are rewritten as

$$i_d(k) = M_{dq}(e_d(k) - E_d^i(k) + \omega_e(k-1)L i_q(k-1)) \quad (50)$$

$$i_q(k) = M_{dq}(e_q(k) - E_q^i(k) - \omega_e(k-1)(L i_d(k-1) + \lambda)) \quad (51)$$

where

$$M_{dq} = 1/(Z_L + R) \quad (52)$$

Motor mechanical equations and coupling

The inertias of the rotary encoder and the coupling (hub 1) can be added to the inertia of the motor to

simplify the calculations. Hence

$$J_{mc} = J_m + J_c + J_{re} \quad (53)$$

where

J_{mc} = motor plus coupling inertia; J_m = motor inertia and J_{re} = rotary encoder inertia.

This reduction of the model is possible because those inertias are modelled as lumped parameter elements. The resultant TLM model is illustrated in Figure 4(c) (reduced from Figure 4b). This electric circuit is solved finding the Thevenin equivalent⁸ with respect to T_{cs} (Figure 4d), thus

$$E_{eq}(k) = Z_{Ecs}E_{cs}^i(k) + Z_{Ect}E_{ct}(k) \quad (54)$$

$$Z_{eq} = \frac{Z_{cs}Z_{ct}}{Z_{cs} + Z_{ct}} \quad (55)$$

where

$$Z_{Ecs} = Z_{ct}/(Z_{cs} + Z_{ct}) \quad (56)$$

and

$$Z_{Ect} = Z_{cs}/(Z_{cs} + Z_{ct}) \quad (57)$$

and

$$Z_{ct} = Z_c + Z_t \quad (58)$$

$$E_{ct}(k) = E_c^i(k) + 2A_1^i(k) \quad (59)$$

$$\omega_m(k) = M_{\omega m}(T_c(k) - E_{mc}^i(k) - E_{eq}(k)) \quad (60)$$

$$T_{cs}(k) = \omega_m(k)Z_{eq} + E_{eq}(k) \quad (61)$$

$$\omega_1(k) = M_{\omega 1}(T_{cs}(k) - E_{ct}(k)) \quad (62)$$

$$E_{cs}^i(k+1) = T_{cs}(k) \quad (63)$$

$$E_c^i(k+1) = -Z_c\omega_1(k) \quad (64)$$

$$E_{mc}^i(k+1) = -Z_{mc}\omega_m(k) \quad (65)$$

$$B_1^i(k+1) = \omega_1(k)Z_t + A_1^i(k) \quad (66)$$

where

$$M_{\omega m} = 1/(b_m + Z_{mc} + Z_{eq}) \quad (67)$$

and

$$M_{\omega 1} = 1/Z_{ct} \quad (68)$$

and

$$Z_{mc} = J_{mc}/t_t \quad (69)$$

Equations (54) to (69) represent the TLM model of the motor mechanical equations and coupling.

Screw shaft torsional model

From the theory outlined by Partridge et al.,⁶ Holroyd et al.,⁴ and Castaneda,⁹ the following model was derived.

The presence of the supporting bearings in the TLM model of the shaft generates the reflection of pulses arriving to the sections where they are placed. In that case, the propagation of pulses in the TLM model takes place on two specific zones (loops) as it is graphically represented in Figure 5(a). The front bearing is placed on section f_b , the nut is on section n_t and the rear bearing is on section h_t , where

$$h_t = \text{round}(l_r/l_{tor}) \quad (70)$$

$$l_{end} = l_{ss} - h_t l_{tor} \quad (71)$$

$$J_{end} = l_{end}I_o \quad (72)$$

$$f_b = \text{round}(l_f/l_{tor}) \quad (73)$$

$$n_t = \text{ceil}(l_n/l_{tor}) \quad (74)$$

The inclusion of the nut in the model will cause the reflection of pulses arriving to section n_t , and therefore splitting the *zone 2* in two loops as shown in Figure 5(b). The model is then reduced to the calculation of the angular velocity on section one, f_b , n_t and h_t ; and the propagation of pulses on the other sections.

The velocity of the front bearing (ω_{fb+i}) is calculated including the TLM model derived for the bearing's friction²² (see Figure 6(a)), thus

$$T = 2(B_{fb}^i(k) - A_{fb+i}^i(k)) \quad (75)$$

$$\omega_{fb+i}(k) = \begin{cases} 0 & \text{for } |T| < T_{fb1} \\ (T - \text{sign}(T)T_{fb1})/Z_{fbq} & \text{for } |T| \geq T_{fb1} \end{cases} \quad (76)$$

where

$$Z_{fbq} = 2Z_t + b_{fb} \quad (77)$$

$$A_{fb}^i(k+1) = B_{fb}^i(k) - \omega_{fb+i}(k)Z_t \quad (78)$$

next pulses

$$B_{fb+i}^i(k+1) = A_{fb+i}^i(k) + \omega_{fb+i}(k)Z_t \quad (79)$$

The angular velocity ω_{nt+i} is calculated according to the procedure specified in the screw shaft, nut and table section.

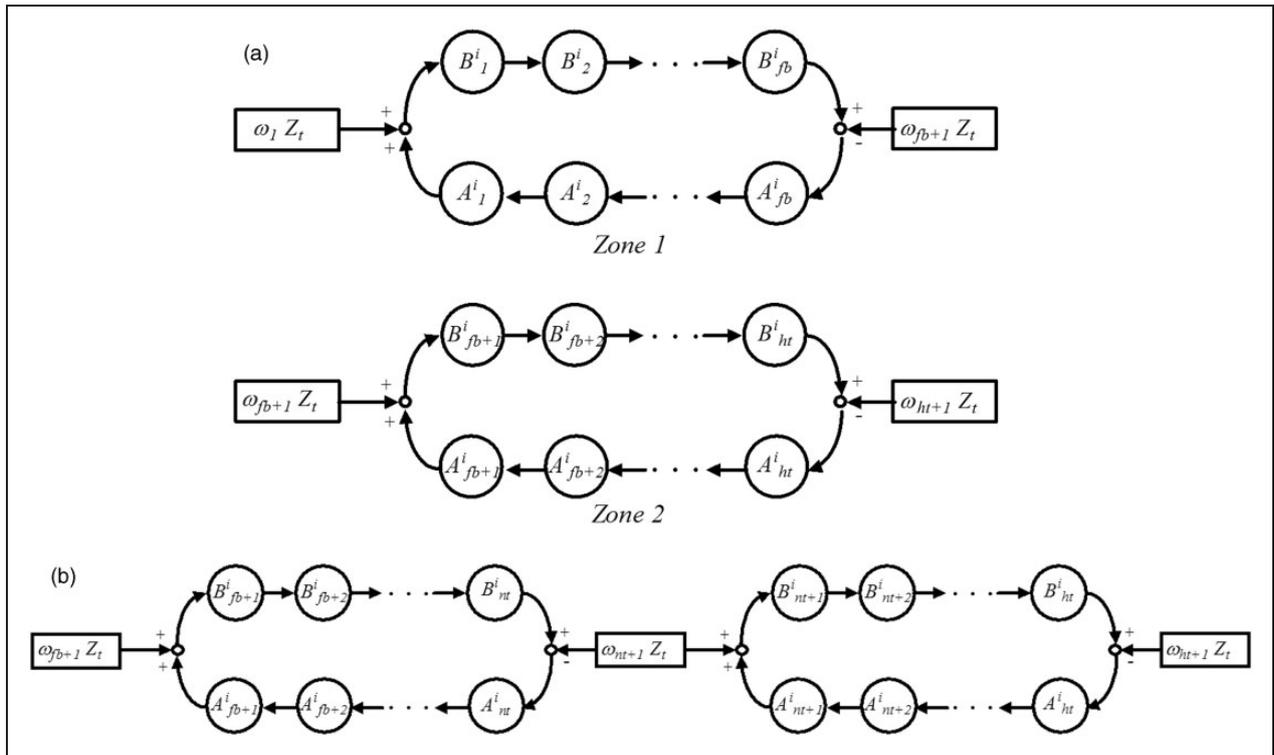


Figure 5. Pulses propagation model for the screw shaft torsional model with moving nut. (a) Without nut and (b) Zone 2 including the nut.

The pulse propagation is defined by

$$B^i_{nt+1}(k+1) = A^i_{nt+1}(k) + \omega_{nt+1}(k)Z_t \quad (80)$$

$$A^i_{nt}(k+1) = B^i_{nt}(k) - \omega_{nt+1}(k)Z_t \quad (81)$$

The velocity of the rear bearing (ω_{ht+1}) is calculated using the procedure applied to the front bearing (see Figure 6b).

$$T = 2B^i_{ht}(k) - E^i_{Jend}(k) \quad (82)$$

$$\omega_{ht+1}(k) = \begin{cases} 0 & \text{for } |T| < T_{rb1} \\ (T - \text{sign}(T)T_{rb1})/Z_{rbeq} & \text{for } |T| \geq T_{rb1} \end{cases} \quad (83)$$

$$Z_{rbeq} = Z_t + b_{rb} + Z_{Jend} \quad (84)$$

next pulses

$$A^i_{ht}(k+1) = B^i_{ht}(k) - \omega_{ht+1}(k)Z_t \quad (85)$$

$$E^i_{Jend}(k+1) = -\omega_{ht+1}(k)Z_{Jend} \quad (86)$$

The propagation of A^i and B^i pulses on the other sections are given by

$$B^i_j(k+1) = B^i_{j-1}(k) \quad \text{for } j = 2, \dots, h_t \\ j \neq f_b + 1, n_t + 1 \quad (87)$$

$$A^i_j(k+1) = A^i_{j+1}(k) \quad \text{for } j = 1, \dots, h_t - 1 \\ j \neq f_b, n_t, h_t \quad (88)$$

Screw shaft axial model

From the theory outlined by Partridge et al.⁶ and Holroyd et al.,⁴ Castaneda⁹ derived the following model.

The presence of the supporting bearings and the nut generates the reflection of pulses arriving to the sections where they are placed. This leads to the propagation of pulses on one zone (loops) in the axial model as it is graphically represented in Figure 7.

The inclusion of the nut in the model will cause the reflection of pulses arriving to section n_a , and therefore splits the model in two loops as shown in Figure 7(b). The front bearing is placed on the first section, the nut is on section n_a and the rear bearing is on section h_a , where

$$h = \text{round}(l_r/l_{axial}) \quad (89)$$

$$l_{end} = l_{ss} - l_{axial}h \quad (90)$$

$$f_{ba} = \text{round}(l_f/l_{axial}) \quad (91)$$

$$l_{front} = f_{ba}l_{axial} \quad (92)$$

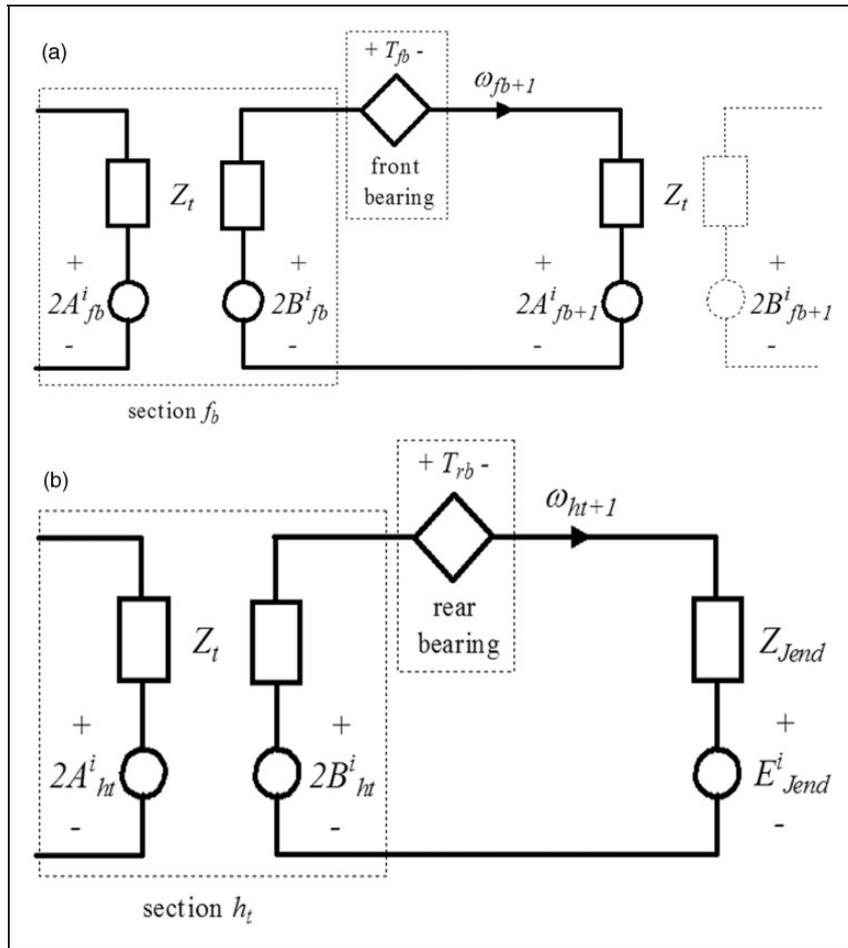


Figure 6. (a) Section f_b of the torsional model. (b) Section h_t of the torsional model.

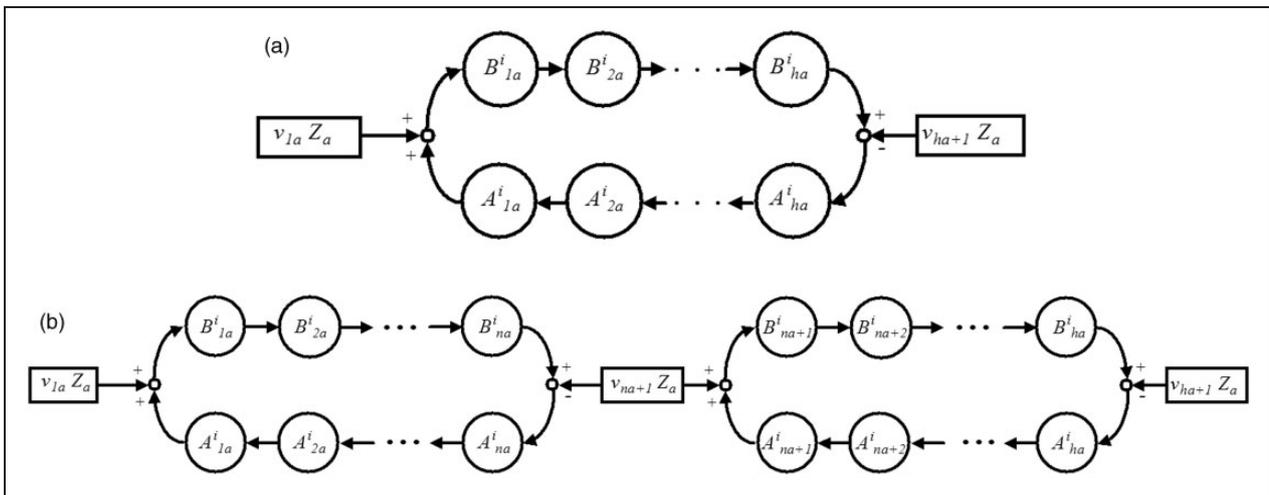


Figure 7. Pulse propagation model for the screw shaft axial model with moving nut. (a) Without nut and (b) including the nut.

$$h_a = (h - f_{ba}) \tag{93}$$

$$n_a = \text{ceil}((l_n - l_{front})/l_{tor}) \tag{94}$$

The model is reduced to the calculation of the velocities v_{1a} , v_{na+1} and v_{ha+1} ; and the pulse propagation on the other sections as defined by the procedure presented by Castaneda⁹ showing the derivation of the velocity v_{1a} , velocity v_{na+1} and the pulse propagation.

Screw shaft, nut and table

Figure 8 illustrates the connection of the axial and torsional TLM screw shaft models with the nut and table models, thus

$$\omega_{nt+1}(k) = \begin{cases} 0 & |T| \leq T_p \\ M_{wnt1}(T - \text{sign}(T)T_p) & |T| > T_p \end{cases} \text{ for } \quad (95)$$

$$v_{nt+1}(k) = k_b \omega_{nt+1}(k) \quad (96)$$

$$T_a(k) = k_b F_a(k - 1) \quad (97)$$

where

$$T = 2BA - T_a(k) \quad (98)$$

$$BA = B_{nt}^i(k) - A_{nt+1}^i(k) \quad (99)$$

$$M_{wnt1}(k) = 1/(2Z_t) \quad (100)$$

The components of the frictional force F_{gw} are calculated as follows

$$F_{Ry} = m_{ly} + F_{cz} \quad (101)$$

$$F_{Ly} = |F_{cx}| \quad (102)$$

$$F_E = X_y F_{Ry} + Y_y F_{Ly} \quad (103)$$

$$F_0 = F_{gw0} + b_{gw} F_E \quad (104)$$

$$F_1 = b_{gw} v_l \quad (105)$$

The velocity of the table (v_l) is calculated finding the Thevenin equivalent⁸ with respect to F_a (Figure 8c):

$$E_{eq} = v_{n+1}(k)Z_{eq} + Z_{CD}CD + Z_{Ens}E_{ns}^i(k) \quad (106)$$

$$CD = B_{na}^i(k) - A_{na+1}^i(k) \quad (107)$$

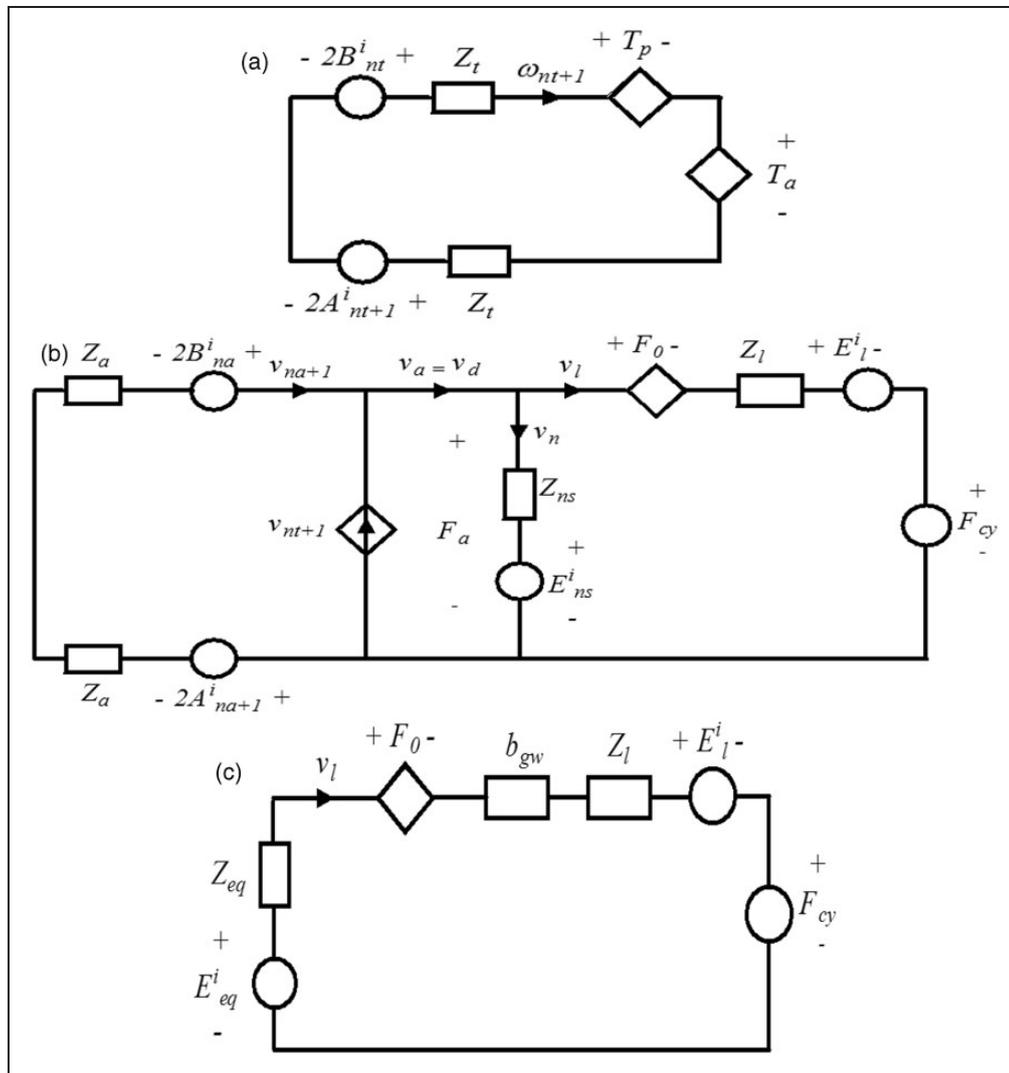


Figure 8. TLM model of the connection between nut and screw shaft. (a) Torsional model connexion, (b) axial load connexion and (c) Equivalent model for the connection between nut and screw shaft.

where

$$Z_{eq} = 2Z_a Z_{ns} / (2Z_a + Z_{ns}) \quad (108)$$

$$Z_{CD} = 2Z_{ns} / (2Z_a + Z_{ns}) \quad (109)$$

$$Z_{Ens} = 2Z_a / (2Z_a + Z_{ns}) \quad (110)$$

$$v_l(k) = \begin{cases} 0 & |F| \leq F_0 \\ M_{vl}(F - \text{sign}(F)F_0) & |F| > F_0 \end{cases} \quad \text{for} \quad (111)$$

$$F = E_{eq} - E_j^i(k) - F_{cy}(k) \quad (112)$$

where

$$M_{vl} = 1 / (Z_{eq} + b_{gw} + Z_l) \quad (113)$$

$$F_a(k) = E_{eq} - v_l(k)Z_{eq} \quad (114)$$

$$E_j^i(k+1) = -v_l(k)Z_l \quad (115)$$

$$E_{ns}^i(k+1) = F_a(k) \quad (116)$$

$$v_{na+1}(k) = M_{vna1}(2CD - F_a(k)) \quad (117)$$

$$M_{vna1} = 1 / (2Z_a) \quad (118)$$

$$d_l(k) = v_l(k)t_a + E_{dl}^i(k) \quad (119)$$

$$E_{dl}^i(k+1) = d_l(k) \quad (120)$$

It must be noted that a new ballscrew with preload is assumed to have no or minimal backlash. Preload is lost over time due to wear of balls and ball races and non-inclusion of backlash term would reduce the value of the model. The model is therefore made applicable to both cases:

- ball-screw with pretension in the nut (Figure 8) $Backlash = 0$ and
- ball-screw without pretension in the nut (Figure 9) $T_p = 0$ and $Backlash \neq 0$.

The backlash model presented has been reduced to the following two possible states:

- when the screw shaft is not in contact with the nut (Figure 9a) and
- when the screw shaft is in contact with the nut (Figure 9b).

The state in which the axis will start at the beginning of the simulation depends on the following conditions:

- Non-contact if:
 - $d_d \neq d_a$
 - $d_d = d_a$ and the direction of motion is negative (nut moving towards the motor).
 - $d_d = d_a + Backlash$ and the direction of motion is positive.
- Contact if:
 - $d_d = d_a$ and the direction of motion is positive.

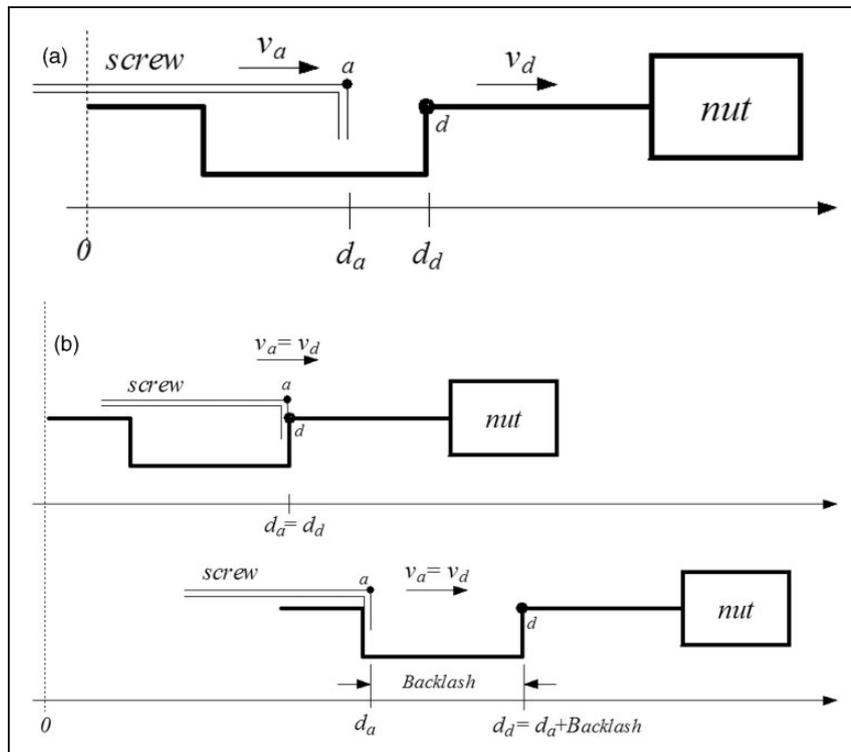


Figure 9. Backlash model. (a) Non-contact and (b) in contact.

- $d_d = d_a + \text{Backlash}$ and the direction of motion is negative.

Variables T_a and F_a are zero if the screw shaft is not in contact with the nut. Thus the values for the velocities are given by the following equations

$$\omega_{nt+1}(k) = BA/Z_t \quad (121)$$

$$v_{nt+1}(k) = k_b \omega_{nt+1}(k) \quad (122)$$

$$v_{na+1}(k) = CD/Z_a \quad (123)$$

$$v_a(k) = v_{nt+1}(k) + v_{na+1}(k) \quad (124)$$

$$v_n(k) = -E_{ns}^i(k)/Z_{ns} \quad (125)$$

$$v_l(k) = \begin{cases} 0 & |F| \leq F_0 \\ M_{vl}(F - \text{sign}(F)F_0) & |F| > F_0 \end{cases} \quad \text{for} \quad (126)$$

$$v_d(k) = v_l(k) + v_n(k) \quad (127)$$

$$F = -E_l^i(k) - F_{cy}(k) \quad (128)$$

where

$$M_{vl} = 1/(b_{gw} + Z_l) \quad (129)$$

$$E_{ns}^i(k+1) = 0 \quad (130)$$

$$F_a(k) = 0 \quad (131)$$

$$E_l^i(k+1) = -v_l(k)Z_l \quad (132)$$

The positions d_a and d_d are calculated by integrating the velocities for the sampling time t_a (equations (124) and (127)). d_l is calculated as in the case for preloaded nut. The model remains in this state if $0 < d_d(k) - d_a(k) < \text{Backlash}$; otherwise the model changes to the contact state (Figure 9 with $T_p = 0$). The model will switch to the non-contact state when the sign of velocity v_{nt+1} changes.

Implementation of two-axis TLM models to a CNC machine tool

Figure 10(c) shows the three-axis vertical machining centre where the unidirectional Z-axis maximum tool travel is 560 mm, and the bidirectional machining tables X-axis maximum travel and Y-axis maximum travel are both 510 mm. The Siemens 840D CNC controls all the machine tool functions. The spindle speed can be controlled from 60 RPM (revolutions per minute) to 8000 RPM and the maximum axis feed-rates are 12 m/min (fast traverse 30 m/min). The machine utilises Siemens 611 electrical feed drives with permanent magnet synchronous motors (Types 1FK6063 on X- and Y-axis and Type 1FT6064 on

Z-axis). The motors for each axis are flexibly coupled to a ball-screw supported by bearings at both ends and utilise linear bearing guide-ways. The Siemens spindle drive unit provides four quadrant operations through a separately excited Siemens DC motor providing a constant torque range 60 RPM to 1500 RPM and above 1500 RPM constant power range (9 KW) to the maximum speed of 8000 RPM.

The VMC-500 is fitted with a Siemens SINUMERIK 840D motion controller, which commands the CNC kernel functions for interpolation and position control. The motion controller is connected to the drives and I/O units via a PROFIBUS-DP interface as shown by Siemens.²³ Each axis integrates a SIMODRIVE-611 Siemens inverter and a ball-screw arrangement directly coupled to a permanent magnet synchronous motor. The ball-screw systems incorporate a preloaded nut and the screw shaft mounted on a fixed-supported bearing configuration.

The SIMODRIVE-611 unit consists of a common feed module that provides the DC voltage link from the power supply mains and a set of drive modules that activate each motor. Every drive module consists of a power module (inverter) and a closed-loop plug-in unit. The closed-loop plug-in unit is dedicated to velocity control, current control and PWM generation functions. Table 2 contains the technical data for the VMC-500 Machine centre.

The main differences between the TNC 426PB and the SINUMERIK 840 D²⁴ in terms of modelling its behaviour are as follows:

- The SINUMERIK 840D includes a velocity response matching filter (first-order delay-filter) used to delay the velocity feed forward signal according to the equivalent position time constant of the closed velocity control loop.
- The SINUMERIK 840D configuration for this machining centre does not include acceleration (torque) feed forward.
- A velocity filter is included to damp the resonant frequencies in the closed position loop.
- A velocity limitation in the form of saturation is included in the position loop.
- Torque and current limitations in the form of saturation are included in the velocity loop.
- The velocity controller does not include differential term (PI control).
- Two additional filters are included in the velocity loop in order to get a filtering process with better time/frequency response. For example, Filter 1 can be configured as the PT₂ filter in the TNC 426 PB and filters 2–4 can be combined to get a band-rejection filter with better damping and frequency properties than the band-rejection filter in the TNC 426 PB.
- The current controllers include integral term (see Motion controller section).

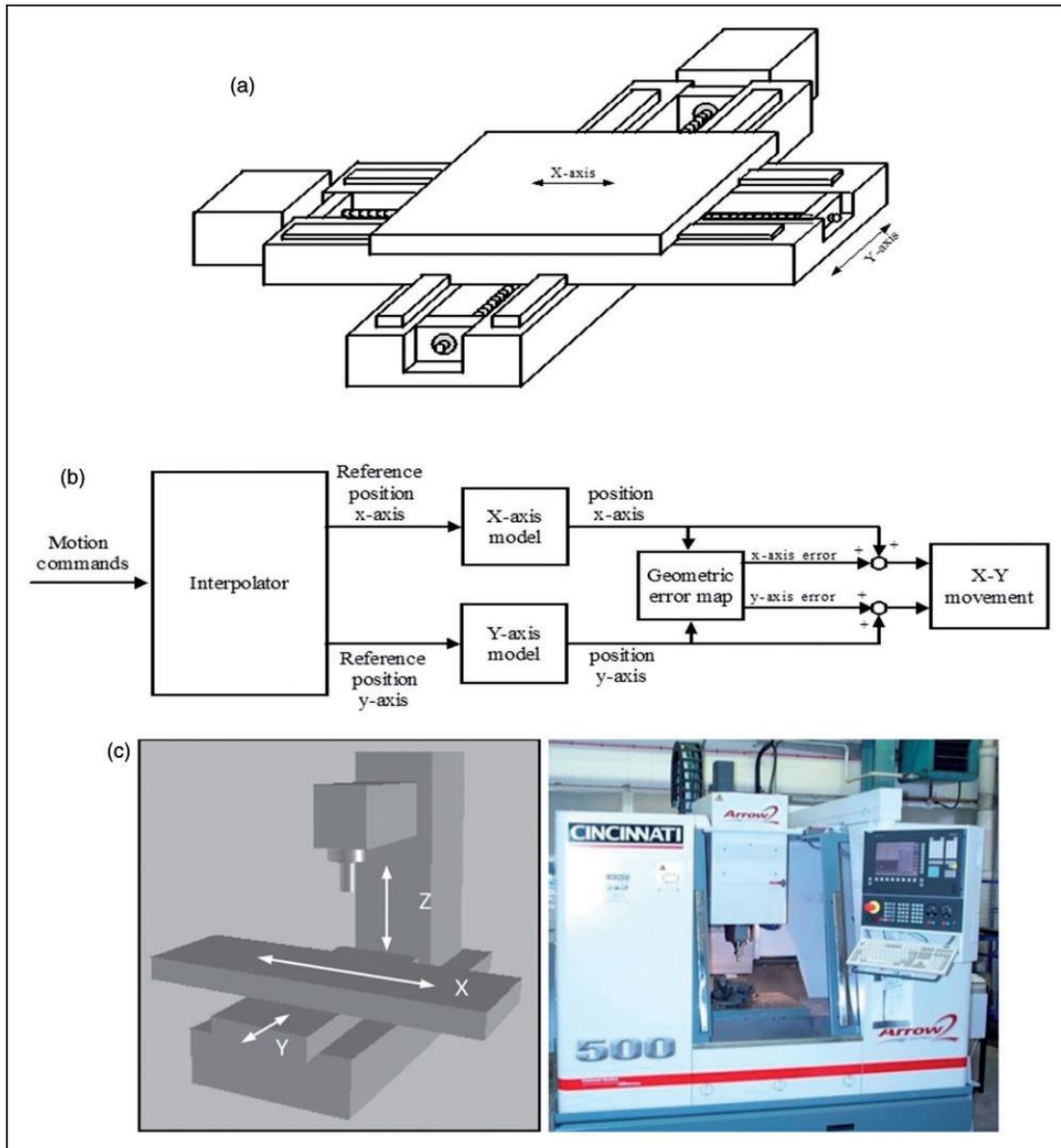


Figure 10. Machine tool. (a) x–y axis feed drive, (b) block diagram and (c) machine tool and coordinate system.

- The sample time for the interpolator and position controller is 4 ms (cf. 3 ms).
- The sample time for the velocity controller is 0.125 ms (cf. 0.6 ms).
- The sample time for the current controller is 0.125 ms (cf. 0.2 ms).

Although the control algorithm is distributed in two different units (SINUMERIK 840D and the plug-in control units), the dynamics acting on the x- and y-axes are modelled as in the single-axis test rig case. The model for the rear bearing mounting has been updated to reproduce the fixed-supported bearing configuration.

The two-axis system is configured as shown in Figure 10(a). Linear and circular interpolation methods described in the Motion controller section are included in the interpolator to coordinate the movement of the axes.

Ford et al.²⁵ derived algorithms for the real-time correction of time and spatial errors in a similar vertical machining centre configuration. Pre-calibrated geometric errors are included in the form of an error map that is used to correct the control movements over the working zone. Figure 10(b) represents the block diagram for the two-axis model. These errors are measured by using linear interferometers, straight edge, electronic levels, dial gauges, ballbars and granite squares.^{25–27}

The two-axis equations for the X–Y plane are derived from Postlethwaite²⁶ considering only the 13 geometric components associated with x- and y-axes, thus

$$E_x = \Lambda_x(x) + \Lambda_x(y) + \Lambda_x(z) + \phi_y(x, y)D_z + \phi_y(y, x)D_z + \phi_z(y, x)D_y + \theta_{xy}(x, y)D_y \quad (133)$$

Table 2. VMC-500 machine specifications.

Specification	X-axis	Y-axis	Z-axis
Motion Controller	SIEMENS SINUMERIK 840D (S7-315-2DP)		
Electrical Drive	SIEMENS SIMODRIVE 611D (6SN1123-1AA00-0CA1)		
Motor	SIEMENS 1FK6063-6AF71-1AG2		SIEMENS 1FT6064-6AH71-4AB1
Guideways	THK HSR35 A2 SS CO QZ + 1090 L H II		
Front bearing	Double Row Angular Contact Ball Bearing NSK BSB 025062		
Rear bearing	Single-row deep groove ball bearing NSK 6305 TB	Single-row radial ball bearing NSK 6005 – 2RS	Single-row deep groove ball bearing NSK 6305 TB
Ballscrew	THK		
Coupling	ROTEX 24 GS spider 98 Shore A		
Rotary encoder	HEIDENHAIN ERN 1387		
Linear encoder	HEIDENHAIN AE LSI86C ML620		
Transverse	510 mm		
Rapid Traverses	30000 mm/min		
Feed Rates	300–12,000 mm/min		
Table Size	700 mm × 520 mm		
Table mass	327 Kg		
Saddle mass		160 Kg	
Carrier mass			212 Kg
Motor mass			152 Kg
Load mass			364 Kg

Table 3. Measurements of the geometric and non-rigid errors.

Axis	Component	Units	Geometric effect
X	Pitch: $\phi_y(x)$	$\mu\text{m/m}$	0.0484
	Yaw: $\phi_z(x)$	$\mu\text{m/m}$	$(-5 \text{ to } 2.8) \cdot 10^{-3}$
	Roll: $\phi_x(x)$	$\mu\text{m/m}$	$(-7 \text{ to } 6) \cdot 10^{-3}$
	Linear positioning: $\Lambda_x(x)$	μm	-0.05 to 9.15
	Horizontal straightness: $\Lambda_y(x)$	μm	-3.46 to 3.5
	Vertical straightness: $\Lambda_z(x)$	μm	-8.84 to -0.64
Y	Pitch: $\phi_x(y)$	$\mu\text{m/m}$	-0.0185
	Yaw: $\phi_z(y)$	$\mu\text{m/m}$	$(-5 \text{ to } 2.8) \cdot 10^{-3}$
	Roll: $\phi_y(y)$	$\mu\text{m/m}$	0 to 0.028
	Linear positioning: $\Lambda_y(y)$	μm	-0.14 to -13.92
	Horizontal straightness: $\Lambda_x(y)$	μm	-0.1 to -6.53
	Vertical straightness: $\Lambda_z(y)$	μm	-0.05 to -3.5
	XY squareness: $\theta(x,y)$	$\mu\text{m/m}$	-26

$$E_y = \Lambda_y(x) + \Lambda_y(y) + \Lambda_y(z) + \phi_x(x,y)D_z + \phi_x(y,x)D_z - \phi_z(x,y)D_x \quad (134)$$

The procedure presented by Ford et al.²⁷ was used to isolate the load errors from the effects of the rigid body geometric errors. Table 3 contains the summary of the maximum values of the geometric and non-rigid errors measured from the actual machine.

Simulation approach for TLM models

Choice of simulation method

One of the aspects that lead to the study of the TLM method for the modelling of feed drives was the possibility of formulation of comprehensive models which would be implemented in real time. MATLAB^{28,29} was used as the computational environment where high-level programming and

visualisation functions were integrated for modelling, simulation and analysis of dynamic systems. The models were formulated in SIMULINK³⁰ and then into a program or as block diagrams using the Graphical User Interface (GUI).

SIMULINK contains a large library of pre-defined blocks that supports the modelling of linear and non-linear systems in continuous time, sampled time, or a hybrid of the two. Systems can also be multi-rate, i.e. have different parts that are sampled or updated at different rates. SIMULINK features a tool called Real Time Workshop (RTW), which automatically generates C code from the SIMULINK models to produce platform-specific code.

Simulation method and development

Five SIMULINK multi-rate subsystem blocks, i.e. position controller, velocity controller, current controller, torsional loop and axial loop (the latter two including for mechanical vibrations and non-linearity's such as backlash and friction forces) were implemented into the software for each axis. The variables interfacing the multi-rate subsystems were implemented in Data Stored Memory Blocks (DSM). A DSM defines a memory region for use by the data store read and data store write blocks. The feature gives access of the memory region to the different sub-systems in order to read from or write to at pre-determined designed variable sample rates.

As an example to the approach, implementation of the single-axis model for the test rig in SIMULINK had to deal with the multi-rate requirements (see the Motion controller section): (a) position controller: v_{ref} at a rate $t_p = 3$ ms; (b) velocity controller: i_{qref} at a rate $t_v = 0.6$ ms and (c) current controller: e_{dqref} at a rate $t_c = 0.2$ ms.

In addition, each PWM signal is composed of seven e_{dq} voltages (switching states) calculated according to the expected currents to be induced in the motor. The duration of each e_{dq} voltage is specified in multiples of the propagation time for the torsional model (t_t). To accomplish this, t_t is made equal to the sampling period on the PWM signal (t_{pwm}). The propagation time on the axial model is a sub-multiple of the torsional propagation time as defined by the Castaneda method⁹ which describes in detail the synchronisation of the torsional and axial models. These actions result in five multi-rate subsystems per axis which were implemented in software for multi-axis system application.

Model development

The modelling development was carried out according to the following methodology:

- The model for the single axis test rig was built in SIMULINK in order to validate the modelling

approach. The validation of the model was achieved by comparing simulated results with the experimental data recorded from the controller.

- Following this step, the single and two axis models for the CNC machine were implemented in SIMULINK taking as a basis the TLM model for the test rig. The x- and y-axes models were validated against experimental data recorded from the controller and the axis drives. The controller's facilities included an integrated oscilloscope and the ability to input stimuli such as PRBS, jerk, and linear/circular interpolation. It also provided an additional capability to perform spectral analysis. The two axis model was validated against experimental data recorded in real time for a ball bar circular test and jerk limited responses (see section on Validation).
- The models to include for the effect of geometric errors measured with specialised equipment such as laser interferometer, ball-bar, electronic precision levels, precision squares, etc. (Table 3). The simulated results were compared with the measured ones (see section on Validation).
- Built-in functions on the CNC machine motion controller enabled spectral analysis to be carried out for the optimisation of the position, velocity and current closed loops for each axis. The control frequency characteristics were calculated by entering a PRBS at the set point of each control loop. Four frequency responses were measured, i.e. closed position controller loop, closed velocity controller loop, system and mechanical frequency response. The closed loop responses for the position and velocity controller were analysed and used for the optimisation of their parameters. The control system and mechanical frequency responses were used to set up the parameters for the filters that are used to damp the system resonant frequencies. The data measured were included in the system simulation and used also for the identification of axis drive modal parameters.
- Other stimuli waveforms were entered to identify parameters suitable for the model such as step (velocity bandwidth overshooting check), triangular (backlash/ dead band), trapezoidal (position gain/coulomb friction) and random white noise/ swept sine (resonant frequencies/damping coefficients).
- The x-axis of the CNC machine was further modified in order to explore the possibility of a real-time implementation for the models. In this regard, the real-time workshop capability of MATLAB/SIMULINK was used to generate a real-time application targeting a RTI-1005 dSPACE environment. The RTI-1005 dSPACE platform was selected because it features the possibility for data logging, control and monitoring of systems in real time.

MATLAB/SIMULINK programs

SIMULINK programs compiled and developed

- A bench mark MATLAB/SIMULINK program was compiled and applied to a previously simulated rotor shaft already in the public domain which was then used to evaluate the reduction of mathematical operations and mean square modelling errors when comparing TLM and MTLM techniques (Conclusions section).
- Single Axis test rig model simulation (see section on TLM method applied to CNC machine tool feed drive)
- Two axis CNC machine tool model simulation (see section on Implementation of two- axis TLM models to a CNC machine tool)
- XY geometric calculation in a SIMULINK block. Developed for insertion of measured geometric errors at the machine into the simulated two axis model (see sections on Implementation of two- axis TLM models to a CNC machine tool and Validation).

MATLAB programs compiled and developed for the following

- Structure of data for the pulse propagation on the axial model (see Screw shaft axial model section)
- Parameters and initialisation code for the torsional loop subsystem (see Screw shaft torsional model section)
- The PWM generating function (see sections on Inverter and motor (electrical) and Motor mechanical equations and coupling)
- Calculation and nut monitoring blocks (see Screw shaft, nut and table section)
- Parameters and initialisation code for the axial loop subsystem (see Screw shaft axial model section)
- Identification of drive modal parameters using the Continuous Wavelet Transform (CWT) for improved performance when compared to other available analysis techniques
- Automatic modal parameter identification using the CWT
- Reference signal profiles swept sign, sinusoidal, white noise, jerk, linear interpolation and circular interpolation (for application comparison)
- Axis block parameters and initialisation code (see section on Motion controller)
- Calculation of polynomial coefficients for the geometric errors (see section on Implementation of two- axis TLM models to a CNC machine tool)
- Calculation of the dilation parameters associated with CWT
- Identification of damping factors and resonant frequencies to be used for axis control system frequency response.

The complexity and detail required for this section is beyond the scope of this publication. To address this issue, we have put all the detail for drive data, measurement methodology, parameter determination, coefficient analysis and a full description of relevant MATLAB/SIMULINK programs online. Most of the detail being contained within the appendices of that report.⁹

Validation

- Step velocity stimuli were tested for 500 mm/min (see Figure 11a) and 1000 mm/min where the simulated velocity and current responses (red line) match the machine measured values (green line). A 3% error was achieved in the transient area (0–0.02 s). The % error increases to 5% when the motor reaches constant velocity (Figure 11b).
- The jerk-limited response was used to verify the performance of the axis drive during acceleration, deceleration and operation at constant velocity. For this purpose, position and velocity responses were measured for different displacements (10, 20, 100, 200, 400, 400 mm) and corresponding feed rates (500, 1000, 5000, 10,000, 20,000 and 40,000 mm/min). Figure 11(c) shows the comparison between measured and simulated position following error for a displacement of 200 mm and a feed rate of 10,000 mm/min. The position versus time measured at the machine and simulated responses matched very well with no significant error but the following error differed minimally by 2 μm at its peak in the acceleration zone and 8 μm at its peak in the deceleration zone. Figure 11(d) shows the comparison between the measured (green line) and simulated (red line) actual velocity signals over the zoom acceleration zone.
- The same reference position and velocity feed-forward signals (radius 150 mm, feed rate of 1000 mm/min) were used for the circular interpolation in the X - Y plane at the machine and for the validation of the x - and y - axis TLM model responses (i.e. response to sinusoidal and cosinusoidal demands).
- The two-axis model was simulated and the model position responses were introduced to produce the trace of x against y . The difference between the trace and the circle is illustrated in Figure 11(e) in the form of a circular plot analogous to that measured using a double ball bar.
- The same sinusoidal stimuli used for the simulation was applied at the machine and the position outputs measured using a Renishaw ball bar system (Figure 11f). An angular overshoot of 180° before and after data capture for a two cycle 360° operation was utilised to minimise uncertainty of test synchronisation.
- Figure 11(e) plot has an oval shape as a result of the squareness error (–26 $\mu\text{m}/\text{m}$) which is consistent with Figure 11(f). The TLM model shows that the introduction of the geometric errors generates a

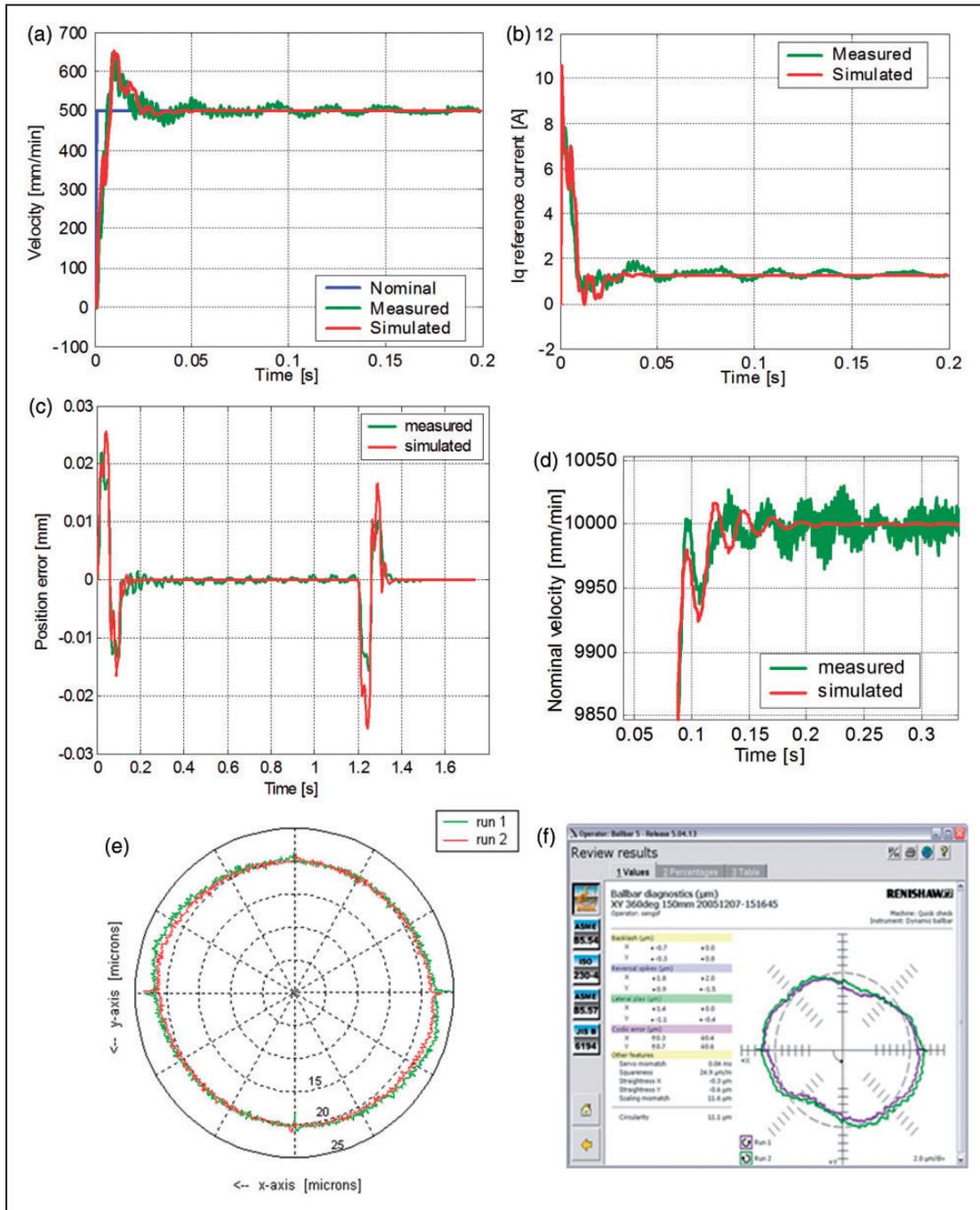


Figure 11. Validation responses comparing TLM model with measured machine. (a) Step velocity response, (b) reference current response to step, (c) position error response jerk-limited input, (d) velocity response to jerk limited input, (e) simulated ball bar trace and (f) measured machine ball bar trace.

progressive error deviation when the machine worktable prescribes a circle. The simulated results do not match the machine measurements on the arcs described in the (45–90°) and (135–270°) zones but with sensitivity analysis this could be further improved. The difference may be due to divergences on the straightness measurements for the x -axis. The reversal spikes match closely on both the x - and y -axes.

Conclusions

- A new MTLM stub was developed which improves the convergence and computational processing speed of the original stub algorithm. A reduction of 40% in the number of mathematical operations and almost 35% reduction in the mean square modelling error figures were achieved over the original TLM stub model.

- Comprehensive transmission line models for the elements of a typical arrangement of a CNC feed drive have been described. All non-linear functions including geometric and load errors have been included as calculated and identified by the measurements undertaken. Equipment such as laser interferometer, ball bar, electronic levels, signal acquisition systems and others were used to obtain parameter data.
- A TLM model for a CNC single-axis feed drive including a digital controller has been developed and then extended to a two-axis drive of a machining centre including geometric and load errors, thus forming the basis for universal CNC machine tool models.
- The simulation of the single-axis and two-axis models to various feed rates and displacements including linear and circular interpolation matches well in comparison with the measured response of the machines under study (see Validation section).
- The application of the MTLM transform and the torsional/axial model's synchronisation approach to the modelling of single- and two-axis drives led to a real time implementation of the feed drive models. Subsequent work not reported in this paper has shown the feasibility of developing the model for implementation of the two- and three-axis models running in real time.
- The x -axis of the machine was modified later in order to explore the possibility of a real-time implementation for the models. In this regard, the real time workshop capability of MATLAB/SIMULINK was used to generate a real-time application targeting a RTI-1005 dSPACE environment. The RTI-1005 dSPACE platform was selected because it features the possibility for data logging, control and monitoring of systems in real time.

Funding

We acknowledge the financial support from the EPSRC Ref. No: GR/S07827/01 and the collaborating companies and also thank the University of Huddersfield provided additional finance to enable further progress.

Conflict of interest

None declared.

References

1. Ford DG. Machining to microns- error avoidance or compensation? In: *Conference proceedings of LAMDAMAP on laser metrology and machine performance II*, Southampton Institute, July 1995, pp. 41–52, Computational Mechanics Publications UK.
2. Pislaru C, Ford DG and Holroyd G. Hybrid modelling and simulation of a CNC machine tool axis drive. *Proc IMechE, Part I: J Systems and Control Engineering* 2004; 218: 111–120.
3. Bartlett H and Whalley R. The response of distributed-lumped parameter systems. *Proc IMechE, Part C: J Mechanical Engineering Science* 1998; 202: 421–429.
4. Holroyd G, Pislaru C and Ford DG. Modelling the dynamic behaviour of a ball-screw system taking into account the changing position of the ball-screw nut. In: *Proceedings of international conference on laser metrology and machine tools CMM and robot performance LAMDAMAP 2003*, Huddersfield, UK, July 2003, pp. 337–348. WIT Press.
5. Boucher RF and Kitsios EE. Simulation of fluid network dynamics by transmission line modelling. *Proc IMechE, Part C: J Mechanical Engineering Science* 1986; 200: 21–29.
6. Partridge GJ, Christopoulos C and Johns PB. Transmission line modelling of shaft system dynamics. *Proc IMechE, Part C: J Mechanical Engineering Science* 1987; 210: 271–278.
7. Johns PB and Beurle RL. Numerical solution of 2-dimensional scattering problems using A transmission line matrix. *Proc IEEE* 1971; 118: 1203–1208.
8. Christopoulos C. *The transmission line modelling method (TLM)*. (The IEEE/OUP Series on Electromagnetic Wave Theory). 1st ed. New York: IEEE Press/Oxford University Press, 1995.
9. Castaneda VYM. *Transmission line modelling applied to non-linear control systems*. PhD thesis, University of Huddersfield, 2006.
10. Heidenhain. *Technical manual TNC 426PB – NC software 280 472, Section 4.8: The control loop*, pp.166–189, 2000. www.heidenhain.co.uk
11. Seborg DE, Edgar TF and Mellichamp DA. *Process dynamics and control*. New York: John Wiley & Sons, 1989.
12. Heinemann G and Papiernik W. Direct linear servo drives for machine tool applications. In: *Intelligent motion proceedings*, PCIM 97 Germany, 1997, pp.93–102.
13. Altintas Y. *Manufacturing automation*. 1st ed. New York: Cambridge University Press, 2000, pp.179–203.
14. Koren Y. *Computer control of manufacturing systems*. Singapore: McGraw Hill, 1983.
15. Bullock T. Command generation, www.control.com/control_com/Papers/Command_Generation_html (2000, accessed 24 October 2003).
16. Hugh J. *Integration and automation of manufacturing systems – Chapter 13: Motion control*. Michigan: Grand Valley State University, 2002.
17. Heidenhain. *Technical manual TNC 426/CB/PB/M – NC Software 280 476, Section 6.8: The control loop*, pp.109–149. Available at: www.heidenhain.uk (2002).
18. The Math Works Inc. *Filter design toolbox for use in MATLAB- Chapter 10: Quantized filtering analysis examples, User's Guide version 2*, 2002, pp.3–11, Available at: www.mathworks.co.uk/help/matlab.
19. The MathWorks Inc. *Processing toolbox for use in MATLAB, User's Guide version 3*, MA, USA, 2002 Available at: www.mathworks.co.uk/help/matlab.
20. Moreton P. *Industrial brushless servomotors*. 1st ed. Oxford, UK: Newnes, 2000, pp.101–115.
21. Simon, N. Implementation of a speed field oriented control of 3-phase PMSM motor using TMS320F240, Texas Instruments- Digital control systems, application report SPRA588, 1999.

22. INA. *Bearings for screw drives*. Herzogenaurach: ZAE Publications, 1999, pp.11–20.
23. Siemens – simodrive 611 6SN1123-1AA00-0CA1, mall.ad.siemens.com (accessed 20 March 2004).
24. Siemens – 1FT6082-1AF7-1AG1, mall.ad.siemens.com (accessed 25 March 2004).
25. Ford DG, Postlethwaite SR, Allen JP, et al. Compensation algorithms for the real time correction of time and spatial errors in a vertical machining centre. *Proc IMechE, Part B: J Engineering Manufacture* 2000; 214: 221–235.
26. Postlethwaite SR. Dynamic calibration of CNC machine tools. *Int J Machine Tool Manuf* 1997; 37: 287–294.
27. Ford DG, Postlethwaite SR and Blake MD. The identification of non-rigid errors in a vertical machining centre. *Proc IMechE, Part B: J Engineering Manufacture* 1999; 213: 555–566.
28. MATLAB. *System identification toolbox for use with MATLAB*. MA: Mathworks Inc, 1998.
29. *Control System Tool Box, for use with MATLAB, users guide version 4*. MA, USA: Mathworks, Inc, 1998.
30. SIMULINK. *Dynamic system simulation for MATLAB, using SIMULINK version 2*. MA, USA: The Mathworks, Inc, 1997.

Appendix I

Notation

a_{ff}	acceleration feed forward (m/s^2)
a_{max}	maximum acceleration (m/s^2)
A^i, B^i	incident pulse (link)
A^i_j, B^i_j	incident voltages associated to Z_i
A^i_{ja}, B^i_{ja}	incident voltages associated to Z_a
A^r, B^r	reflected pulse (link)
b_{gw}	guideway friction coefficient (N-s/m)
b_m	coefficient of friction (motor bearings) (N-m-s/rad)
d	displacement (m)
d_a	displacement of the end d of the nut (m)
d_{act}	actual position value (rotary or linear encoder) (mm)
d_b	displacement of the end b of the bearing (m)
d_{bh}	displacement of the end bh (bearing housing) (m)
d_d	displacement of the end d of the nut (m)
d_e	position error (mm)
den	denominator filter coefficients vector
d_l	actual table position (from linear encoder) (mm)
d_{nom}	reference position signal (before position filter)
d_{prof}	axis-drive position profile
d_{ref}	reference position (position demand) (mm)
D_x	x-axis position (mm)
D_y	y-axis position (mm)
D_z	z-axis position (mm)
e_β	β motor voltage (V)

e_α	α motor voltage (V)
e_a, e_b, e_c	motor line voltages (V)
e_d	direct motor voltage (V)
e_{dqref}	reference voltage to PWM (V)
e_m	motor inertia effort
e_q	quadrature motor voltage (V)
e_{qref}	reference voltage q -component (V)
E^i	incident pulse (stub)
E^i_{aff}	incident pulse associated to Z_{aff}
E^i_c	incident pulse associated to Z_c
E^i_{cs}	incident pulse associated to Z_{cs}
E^i_d	incident pulse associated to Z_L (d -component)
E^i_{de}	incident pulse associated to Z_{ci} (d -component)
E^i_{dl}	incident pulse for d_l calculation
E^i_{end}	incident pulse associated with Z_{end}
E^i_{fbh}	incident pulse associated with Z_{fbh}
E^i_i	incident pulse associated to Z_i
E^i_l	incident pulse associated to Z_l
E^i_m	incident pulse associated to Z_m
E^i_{mfb}	incident pulse associated with Z_{mfb}
E^i_{mrb}	incident pulse associated to Z_{mrb}
E^i_{ns}	incident pulse associated to Z_{ns}
E^i_q	incident pulse associated to Z_L (q -component)
E^i_{qe}	incident pulse associated to Z_{ci} (q -component)
E^i_{rb}	incident pulse associated to Z_{rb}
E^i_{rbh}	incident pulse associated to Z_{rbh}
E^i_{thm}	incident pulse for θ_m calculation
E^i_{vff}	incident pulse associated to Z_{vff}
E^r	reflected pulse (stub)
E_x	actual error movement of the x-axis (μm)
E_y	actual error movement of the y-axis (μm)
E_z	actual error movement of the z-axis (μm)
f_l	bearing load coefficient
f_o	bearing lubrication method
f_r	feed rate (mm/min)
F	force (N)
h	number of sections
h_a	number of sections of the axial model
h_t	number of sections torsional model
i	electric current (A)
i_β	β motor current (A)
i_α	motor current α -component (A)
i_a, i_b, i_c	motor phase currents (A)
i_d	direct motor current (A)
i_{dact}	actual current d -component (A)
i_{de}	current error d -component (A)
i_{der}	derivative component of i_{ref} (A)
i_{dref}	reference current d -component (A)
i_{ht}	holding current (A)
i_{int}	integral component of i_{ref} (A)
i_p	proportional component of i_{ref} (A)
i_q	quadrature motor current (A)

i_{qact}	actual current q -component (A)	n_{fa}	number of sections first zone (axial model)
i_{qe}	current error q -component (A)	n_l	number of sections on the left (second zone torsional model)
i_{qref}	reference current q -component (A)	R	resistance (ohm)
i_{ref}	reference current (current demand) (A)	t_a	propagation time of axial waves (s)
j	jerk (m/s^3)	t_c	current control loop cycle time (s)
j_{max}	maximum jerk (m/s^3)	t_i	time (s)
k	step number	t_p	position control loop cycle time (s)
k_{aff}	acceleration feed forward gain ($A\cdot s^2/rad$)	t_s	simulation time step
k_b	ball screw force to torque constant	t_t	propagation time for torsional waves (s)
k_{ci}	Integral gain current controller (V/A-s)	t_v	velocity control loop cycle time (s)
k_{cp}	Proportional gain current controller (V/A)	T	torque (N-m)
k_{cs}	Torsional stiffness of the coupling (N-m/rad)	T_a	torque due to external load (N-m)
k_d	Velocity controller derivative gain ($A\cdot s^2/rad$)	T_b	bearing frictional torque (N-m)
k_e	Electric constant of the motor (V-s/rad)	T_c	input torque first section (torsional model) (N-m)
k_{eq}	Bearing mounting stiffness	T_e	generated electromagnetic torque (N-m)
k_{ff}	Feed forward gain	T_{f0}	T_b velocity-dependent component (N-m)
k_i	velocity controller integral gain (A/rad)	T_{f1}	T_b load-dependent component (N-m)
k_n	nut rigidity (N/m)	T_{fb}	front bearing frictional torque (N-m)
k_p	velocity controller proportional gain (A-s/rad)	T_{ld}	load torque (N-m)
k_{rb}	bearing stiffness	T_m	motor load torque (N-m)
k_{rbh}	bearing housing stiffness	T_p	nut pre-loading torque (N-m)
k_{rn}	resulting rigidity of the preloaded nut with mounting bracket (N/m)	T_{rb}	rear bearing frictional torque (N-m)
k_T	torque constant of the motor (N-m/A)	u	propagation velocity (m/s)
k_v	gain of the position controller (m/min-mm)	u_a	velocity of propagation of axial waves (m/s)
k_{vff}	feed forward velocity gain (rad/mm)	u_t	velocity of propagation of torsional waves (m/s)
l	length (m)	v	velocity (m/s)
l_{axial}	length of each section in the axial model (m)	v_a	velocity of the screw shaft point a (m/s)
len	number of filter coefficients	v_{act}	actual velocity (rad/s)
$len0$	filter order	v_b	displacement of the end b of the bearing (m/s)
l_{end}	length end portion of the screw shaft (m)	v_{bh}	displacement end bh of the bearing housing (m/s)
l_f	positions of the front bearing (m)	v_{bmb}	relative velocity between bearing ends (m/s)
l_n	positions of the nut (m)	v_d	velocity of the end d of the nut (m/s)
l_o	absolute reference point for the nut movement (m)	v_e	velocity error (rad/s)
l_r	positions of the rear bearing (m)	v_{fb}	front bearing velocity (m/s)
l_s	ball screw stroke length (m)	v_{ff}	velocity feed forward (rad/s)
l_{ss}	screw shaft length (m)	v_{ha}	rear bearing velocity (m/s)
l_{tor}	length of each section in the torsional model (m)	v_l	load velocity (m/s)
n_{la}	number of sections on the left (second zone axial model)	v_m	maximum possible velocity (m/s)
n_m	number of sections second zone (torsional model)	v_n	nut velocity (relative velocity between ends) (m/s)
n_{ma}	number of axial sections per torsional section	v_{na}	velocity of the nut contact point (m/s)
n_t	number of axial sections per torsional section	v_{ref}	reference velocity value (velocity demand) (rad/s)
num	numerator filter coefficients vector	x	distance (m)
n_f	number of sections first zone (torsional model)	x	x-coordinate (mm)
		y	y-coordinate (mm)
		zz_i	filter delay output
		Z_a	characteristic impedance axial model

Z_{aff}	characteristic impedance associated to k_{aff}	θ_m	mechanical position (from rotary encoder) (rad)
Z_c	characteristic impedance associated to J_c	$\theta_{xy}(x, y)$	squareness in the XY plane ($\mu\text{m}/\text{mm}$)
Z_{ci}	characteristic impedance associated to k_{ci}	$\theta_{xz}(x, z)$	squareness in the XZ plane ($\mu\text{m}/\text{mm}$)
Z_{cs}	characteristic impedance associated to k_{cs}	$\theta_{yz}(y, z)$	squareness in the YZ plane ($\mu\text{m}/\text{mm}$)
Z_d	characteristic impedance associated to k_d	$\Lambda_x(x)$	x-axis linear positioning error (μm)
Z_{end}	characteristic impedance associated with J_{end}	$\Lambda_x(y)$	y-axis straightness in the x-axis direction (μm)
Z_{fbh}	characteristic impedance associated with k_{fbh}	$\Lambda_x(z)$	z-axis straightness in the x-axis direction (μm)
Z_i	characteristic impedance associated to k_i	$\Lambda_y(x)$	x-axis straightness in the y-axis direction (μm)
Z_L	characteristic impedance associated to L	$\Lambda_y(y)$	y-axis linear positioning error (μm)
Z_l	characteristic impedance associated to m	$\Lambda_y(z)$	z-axis straightness in the y-axis direction (μm)
Z_m	characteristic impedance associated to J_m	$\Lambda_z(x)$	x-axis straightness in the z-axis direction (μm)
Z_{mfb}	characteristic impedance associated with m_{fb}	$\Lambda_z(y)$	y-axis straightness in the z-axis direction (μm)
Z_{mrb}	characteristic impedance associated to m_{rb}	$\Lambda_z(z)$	z-axis linear positioning error (μm)
Z_{ns}	characteristic impedance associated to k_n	ϕ	incremental position for a rotary encoder
Z_o	characteristic impedance	$\phi_x(x)$	x-axis rotation about x-axis ($\mu\text{m}/\text{mm}$)
Z_{rb}	characteristic impedance associated to k_{rb}	$\phi_x(y)$	y-axis rotation about x-axis ($\mu\text{m}/\text{mm}$)
Z_{rbh}	characteristic impedance associated to k_{rbh}	$\phi_x(z)$	z-axis rotation about x-axis ($\mu\text{m}/\text{mm}$)
Z_t	characteristic impedance torsional model	$\phi_y(x)$	x-axis rotation about y-axis ($\mu\text{m}/\text{mm}$)
Z_{vff}	characteristic impedance associated to k_{vff}	$\phi_y(y)$	y-axis rotation about y-axis ($\mu\text{m}/\text{mm}$)
θ_l	coupling displacement at the screw shaft side (rad)	ω_l	coupling velocity at the screw shaft side (rad/s)
θ_m	actual (rad)	ω_m	mechanical velocity (motor angular velocity) (rad/s)
		ω_a	angular velocity of the screw shaft point a (rad/s)
		ω_c	coupling angular velocity (rad/s)
		ω_e	electrical velocity (rad/s)
		ω_{fb}	front bearing angular velocity (rad/s)
		ω_h	rear bearing angular velocity (rad/s)
		ω_n	angular velocity of the nut contact point (rad/s)