Efficient Modelling for a Brushless DC Motor Drive

P C K Luk[*] , C K Lee[**]

[*] School of Electronic & Electrical Engineering, The Robert Gordon University, Schoolhill, Aberdeen AB9 1FR, United Kingdom

[**] Hong Kong Polytechinc, Department of Electronic Engineering, Hung Hom, Hong Kong.

Abstract — **To limit the cost incurred in repeated prototyping of both the motors and inverters in a brushless DC (BLDC) motor drive system, a cost efficient simulation of the system at the design stage is desirable. The proposed mathematical model, developed in the *de facto* industry standard MATLAB environment, allows design engineers a quick investigation into the performance of the system when variations such as load or sampling rate of the digital controller occur. A user-friendly interface to the input of simulation parameters has been incorporated. The modular approach adopted facilitates program maintenance and further development. Some simulation and validating results are included.**

INTRODUCTION

Whilst BLDC motor drives, systems in which a permanent magnet excited synchronous motor is fed with a variable frequency inverter controlled by a shaft position sensor, are becoming commercially available on an increasingly scale, there appears a lack of commercial simulation packages for the design of controller for such BLDC motor drives. One main reason has been that the high software development cost incurred is not justified for their typical low cost fractional/integral kW application areas such as NC machine tools and robot drives, even it could imply the possibility of demagnetising the rotor magnets during commissioning or tuning stages. Nevertheless, recursive prototyping of both the motor and inverter may be involved in novel drive configurations for advance and specialised applications, resulting in high developmental cost of the drive system. Improved magnet material with high $(B.H)_{max}$ product also helps push the BLDC motors market to tens of kW application areas where commissioning errors become prohibitively costly. Modeling is therefore essential and may offer potential cost savings. The simulation package developed by the authors, based on the industry accepted simulation package MATLAB, aims to offer high portability and cost effectiveness in term of software development. To run the program, it requires only the MATLAB platform which is enjoying increasing acceptance in different branches of engineering. It is intended that the package will be developed into a Brushless DC Motor Toolbox to the MATLAB.

SYSTEM DESCRIPTION

Fig.1 shows the block diagram of the drive system under investigation. The controller can be configured either as a position or velocity control system and thus accepts a position or a speed input reference. To model a digital implementation, the controller uses a discrete state space approach where the feedback signals and control signals are sampled and generated in a discrete time domain. The controller incorporates a proportional-integral-differential (PID) control scheme and provides signal to the pulse-width modulated (PWM) inverter.

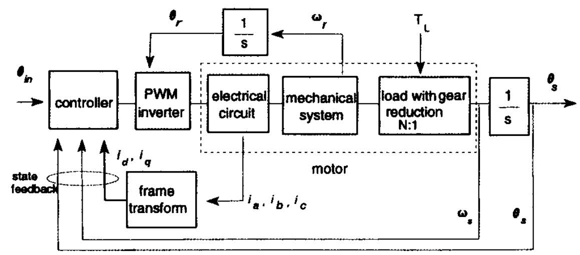


Fig.1 System block diagram

The PWM inverter, as illustrated in Fig.2, performs a one-stage power conditioning for the motor. The three-phase balanced voltages, determined by the rotor angular position which is found by the use of hall effect devices, are applied to the motor phase windings in the switching sequence shown in Fig.2. It is possible to perform a frame transformation from space *(a,b,c)* to space *(d,q)* is to improve computational efficiency [1], where space *(a,b,c)* refers to the motor's phase quantities and space *(d,q)* refers to the rotating frame of the rotor. It has however been shown that such a transformation offers no advantage in computational efficiency when the supply voltage is not sinusoidal [2]. A choice of either scheme is therefore implemented here. The phase currents, together with the signals of the angular speed and position, constitute the state feedback signals for the controller. Since the practical implementation will usually involve such frame of transformation and other real-time computations, a delay in outputing the actuation signal is incurred. Such delays are therefore included in the model. The load torque may be constant, step changing, a random variable with normal distribution or a polynomial function of the rotor speed.

0-7803-1328-3/94$03.00© 1994 IEEE

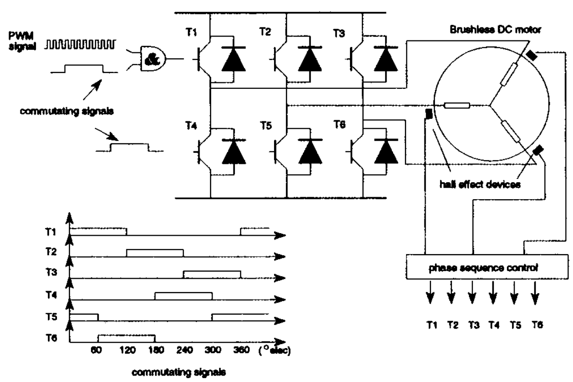


Fig.2 BLDC motor drive and scheme

MODELING OF SYSTEM

Motor and Load Model

The line voltage equations in (a,b,c) frame are defined by the following matrix equation:

$$\begin{bmatrix} v_{ab} \\ v_{bc} \\ v_{ca} \end{bmatrix} = \begin{bmatrix} R & -R & 0 \\ 0 & R & -R \\ R & 0 & R \end{bmatrix} . \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} (L-M) & (M-L) & 0 \\ 0 & (L-M) & 0 \\ 0 & (M-L) & (L-M) \end{bmatrix} . \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a - e_b \\ e_b - e_c \\ e_c - e_a \end{bmatrix} . K_e \omega_r \tag{1}$$

$$i_a + i_b + i_c = 0$$

where,
L, M are the stator self and mutual inductances;
R are the stator phase winding resistances;
$e_{a,b,c}$ are the back emfs; and
ω_r and K_e the rotor angular speed and the emf constant.

For accurate modeling, back emfs are often computed by finite element analysis. An ideal trapezoidal back emf is however used in this paper for the sake of simplicity.

The corresponding equation for the electrical torque developed by the motor is

$$T_e = K_e (i_a e_a + i_b e_b + i_c e_c) \tag{2}$$

which is expended in overcoming inertia, friction and load as

$$T_e = J \frac{d}{dt} \omega_r + B\omega_r + T_L + T_c \tag{3}$$

where J is the moment of inertia, T_L and T_c are load and cogging torques, and B is the frictional constant.

Three-Phase PWM Inverter Model

The modeling of non-linear power switching devices often leads to the possibility of numerical instability and results in an inherently stiff system. Ironically, whilst the improved characteristics of modern power switching devices means that technology has brought the 'ideal switch' one step closer, they also introduce extra computational complexity in modeling when the OFF state and ON state impedances are often in a ratio in excess of 10^{13}. A method in which the switch is model by a *L-R* circuit is adopted here to ensure numerical stability and efficient simulation. The time constant of the *L-R* circuit is carefully chosen such that the off-state impedance of an open phase has a time constant similar to that of the on-state phase. The detailed description of the inverter model warrants a treatment on its own and is therefore to be reported elsewhere. Conceptually, the PWM waveform is generated by superimposing the commutation signals with a chopper signal with controllable mark-to-space ratio as shown in Fig.2. This removes the need of a variable DC link voltage. Discrete PWM signals and commutation signals are presented to the gate of the power device according to the switching sequence.

System Model

A system equation in state space form can be obtained by rearranging equation (1) to (3) in the form:

$$\frac{d}{dt} \boldsymbol{i} = A\, \boldsymbol{i} + B\, \boldsymbol{u} \tag{4}$$

where $\boldsymbol{i}$ is the state variable of the current vector and $\boldsymbol{u}$ is the PWM input vector, A and B are transformed matrices from the resistance and inductance matrices of equation (1) accordingly.

The discrete form of equation (4) is given by

$$\boldsymbol{i}(n+1) = \Phi\, \boldsymbol{i}(n) + \Gamma\, \boldsymbol{u}(n) \tag{5}$$

Given T_s is the sampling period, since the PWM input $\boldsymbol{u}(nT_s)$ over the sampling period is constant, Γ and Φ are then functions of T_s, A and B only. The transformation of equation (4) to (5) is achieved by the MATLAB command ***c2d*** (continuous to discrete).

Controller Model

A conventional PID control law has been implemented and the output is given by

$$PID(n) = K_p E_r(n) + K_i[E_r(n)+E_r(n-1)] + K_d[E_r(n)-E_r(n-1)] \tag{6}$$

where,
$E_r(n)$, $E_r(n-1)$ are current and previous system errors; and
K_p, K_i and K_d are the proportional, integral and differential coefficients.

Advanced control algorithms such as neural net and fuzzy control can be incorporated to the program in a modular fashion.

SOFTWARE DEVELOPMENT

The proposed simulation model of the drive system constitutes the main modules as illustrated in Fig.1. In effect, each module is a sequence of MATLAB statements which make up the so-called M-file. The M-file may contain references to other M-files. As such each M-file can be broken down into suitable and manageable size.

Modular Approach

A modular approach is adopted to facilitate good software maintenance and future development. This is supported by the programming environment provided by MATLAB. Indeed the MATLAB's philosophy, where computations are dealt with exclusively in matrices, lends itself to electrical machine modeling. Matrices have always been the main mathematical tool used in the analysis and modeling of electrical machines where they can be succinctly represented. By representing the model in matrices in so far as possible, therefore, programming time has been significantly shortened while program efficiency is also guaranteed by MATLAB. A *structure chart* of the program development is shown in Fig.3. Each box represents a separate file (called the M-file in MATLAB) which is kept in easily manageable size. The chart shows how additional modules can be easily incorporated to the program. This provides a very forward looking platform where advanced features such as neural-fuzzy control, finite element analysis of rotor's back emf and generated torque, can be included in the form of modules (additional M-files).

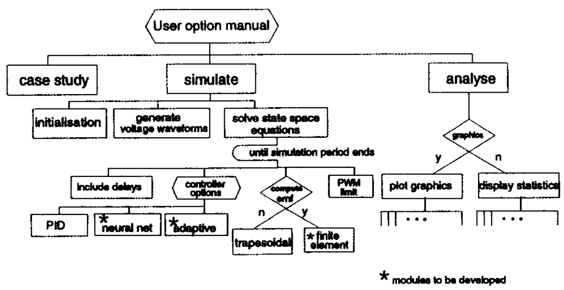


Fig.3 Structure chart for program development

Profile of features currently supported

The user is presented with an option table when the program is invoked. The '**case study**' serves as a help guide by providing an excursion to visit various features in the program. The '**simulate**' allows user-selectable system variables. These variables are broadly classified into several categories:

(a) simulation environment,
(b) electrical and mechanical parameters of the motor,
(c) PID controller settings,
(d) inverter operation mode and source voltage, and
(e) load torque magnitude and mode of fluctuations.

A default value is preset for each variable and updated data will be store in a data file for future reference.

The '**analyse**' provides rapid visual impressions of various waveforms of interest, and a comprehensive set of statistical data for further examination. Graphical displays are available for:

(a) controlled variables (position or velocity),
(b) system error,
(c) actuation command,
(d) phase currents,
(e) electromechanical torque, and
(f) harmonic spectrum of torque, voltages and currents.

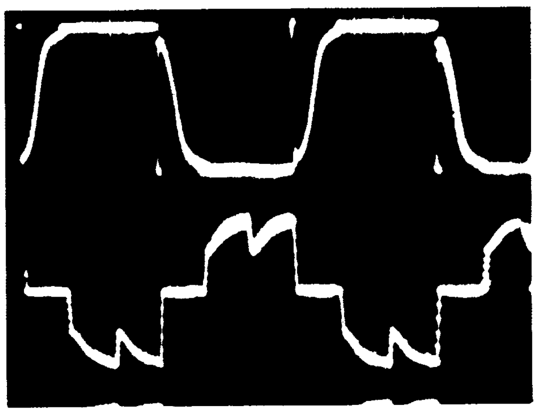
The statistical data include:

(a) integrated time square error (ITSE),
(b) peak time, overshoot,
(c) damping factor and system's natural frequency, and
(d) harmonic content of different waveforms.

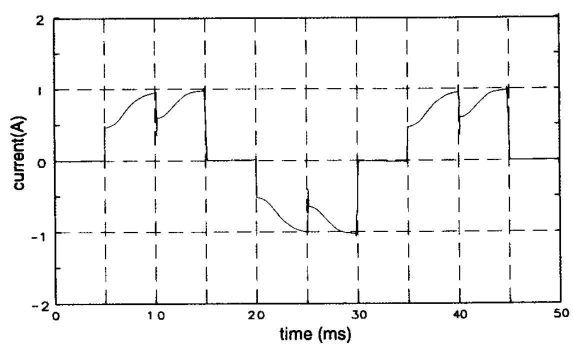
These data should form the basis of determining the optimised setting of the system parameters for a given application.

MODEL VALIDITY

To demonstrate the validity of the model used in the package, simulation results were compared with experimental results. Fig.4 shows the actual and predicted current waveforms for a 1 kW 6-pole 3-phase BLDC motor at a constant load of 1Nm in steady state. The simulated results of the motor torque and response for a step input of 200 rpm (21 rad/s) and 1 rad are shown respectively in Fig.5 and Fig.6, which highlight the effects due to an optimally-tuned and over-tuned PID controller respectively.

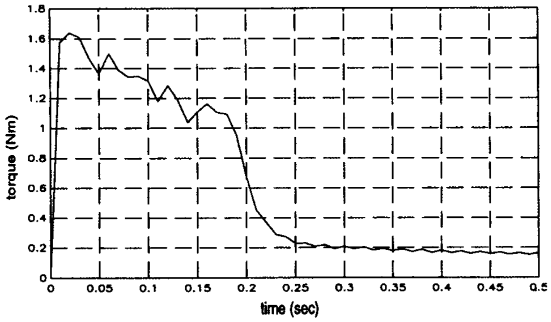


(a)

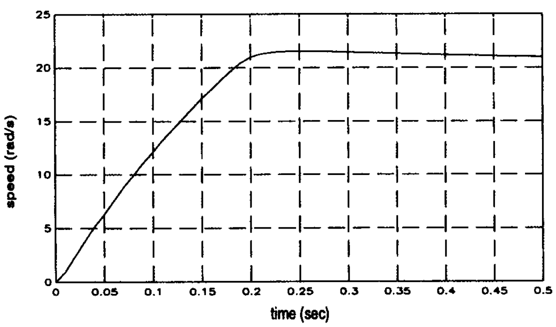


(b)

Fig.4 Validating results (a) measured - Inverter voltage (upper) ; phase current (lower) (b) predicted current

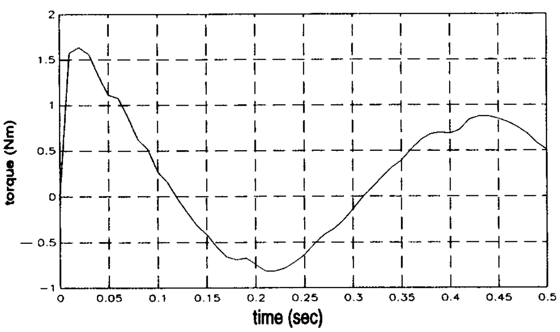


(a) torque response

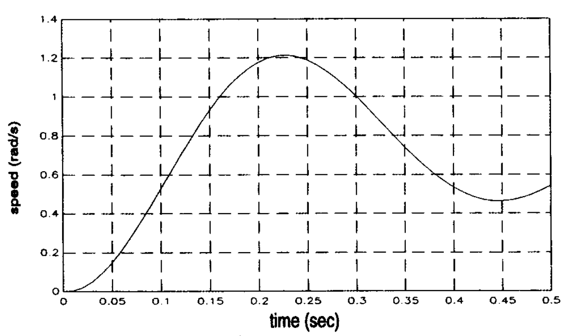


(b) speed response

Fig.5 Predicted responses for speed control (optimally-tuned)



(a) torque response



(b) position response

Fig.6 Predicted responses for positional control (over-tuned)

CONCLUSION

The developed BLDC motor drive model offers an efficient and user friendly environment to examine implementation aspects encountered in the design of such drive systems. The software allows the user to trade off implementation details of the drive system at the design stage and is particularly useful for cost-critical projects. Since it runs on the platform of the industry accepted MATLAB simulation package, high portability and good software support are assured in the future. The package is both a viable tool for designing BLDC motor control systems and has been used as an invaluable teaching tool.

REFERENCES

[1] C.K.P.Luk, M.G.Jayne, D.Rees, D.W.Schaper,"A digital model for a three-phase induction motor drive using a personal computer software package", *Computer Aided Design in Control Systems*. Selected Papers from IFAC Symposium, pp.411-416, edited by H.A.Barker, Pergamon, 1992.

[2] A.K.Wallace and R.Spee, "The effects of motor parameters on the performance of brushless DC drives," *IEEE Power Electronic Specialists Conf. Record*, 1987, Blacksburg, VA,pp.591-597.