ELSEVIER

Computer Methods and Programs in Biomedicine 55 (1998) 59-68

Computer Methods and Programs in Biomedicine

Dynamic model of the vergence eye movement system: simulations using MATLAB/SIMULINK

George K. Hung *

Department of Biomedical Engineering, Rutgers University, P.O. Box 909, Piscataway, NJ 08855-0909, USA

Received 30 July 1997; received in revised form 28 August 1997; accepted 28 August 1997

Abstract

A dynamic model of the vergence eye movement system was developed and simulated using MATLAB/SIMULINK. The model was based on a dual-mode dynamic model previously written in FORTRAN. It consisted of a fast open-loop component and a slow closed-loop component. The new model contained several important modifications. For example, in the fast component, a zero-order hold element replaced the sampler and the target trajectory estimator in the earlier model to provide more stable and accurate responses. Also, a periodicity detector was added to automatically detect periodicity in the stimulus waveform. The stored periodic stimulus, with a reduction in latency, was used to drive the fast component output. Moreover, a connection representing the efference copy signal was added from the fast component output to the disparity input to provide an accurate estimate of the stimulus waveform. Further, Robinson's model of the extraocular muscles replaced the earlier 2nd-order plant to provide more realistic muscle dynamics. The entire model, containing the fast and slow components, was simulated using a variety of stimuli such as pulses, positive and negative ramps, square-wave, and sine-wave. The responses showed dynamic characteristics similar to experimental results. Thus, this new MATLAB/SIMULINK program provides a relatively easy-to-use, versatile, and powerful simulation environment for investigating the basic as well as clinical aspects of vergence dynamics. Moreover, the simulation program has general characteristics that can be modified to represent other oculomotor systems such as the versional and accommodation systems. This provides a framework for future investigation of dynamic interactions between oculomotor systems. © 1998 Elsevier Science Ireland Ltd.

Keywords: Vergence; Dynamics; Model simulations; Fast and slow components; MATLAB

1. Introduction

The vergence system responds to change in depth of a target by oppositely directed rotations of the two eyes [1,2]. It exhibits a latency of about

* Tel.: + 1 908 4454137; fax: + 1 908 4453653.

PII S0169-2607(97)00048-5

180 ms for convergence and 200 ms for divergence [3]. Its dynamics are relatively slow compared to the saccadic eye movement system [4], and has a time constant of about 180 ms for convergence and 250 ms for divergence [5]. It is this relatively slow dynamics, along with its latency, that have made it difficult to model vergence as a continuous feedback system. Indeed, when open-loop parameters based on experimental data were input into a continuous feedback model, it exhibited instability oscillations [6]. To maintain accuracy as well as stability, Hung et al. [6] proposed a dual-mode model for the vergence eye movement system. It consisted of a fast open-loop component and a slow closed-loop component. The fast component accounted for most of the vergence response, while the slow component used feedback to reduce the small residual disparity to a minimum. Model simulation responses were shown to be stable and robust to a variety of stimuli. A detailed discussion of the model has been provided by Hung et al. [6].

However, due to the complexity of the model program, which was originally written in FORTRAN [6], it had been difficult for other researchers to implement the program on their own computers and to simulate experimental and clinical vergence responses. Recently, more powerful and versatile software platforms, such as MATLAB/SIMULINK (henceforth, called simply MATLAB), have become widely available, so that a conversion of the vergence dual-mode model from FORTRAN to MATLAB appeared worthwhile. Yet, this was not a simple undertaking because there were major differences between FORTRAN, a programming language, and MATLAB, a block-structured simulation package. Also, the logic flow could not be easily converted from one to the other. Hence, the software code needed to be completely re-written. On the other hand, powerful signal analysis programs in MATLAB were available for improving the performance of the model.

As described below, the most significant component of this MATLAB model is the m-file written for the fast component, Fast.m. All the important features, such as stimulus amplitude and velocity limits, zero-order hold (ZOH), variable latency to different velocity ramp stimuli, re-triggering by a rapidly changing stimulus, and auto- and cross-correlations used for prediction of periodic stimuli, are incorporated in Fast.m. This is in turn all embedded in the block FAST (see Fig. 1). The block diagram contains the efference copy signal [7,8] from the output of the fast component that is summed with the disparity to provide an estimate of the stimulus. Moreover, the slow component block, SLOW, contains feedback connections for closed-loop control of disparity. The outputs of the FAST and SLOW elements are input to the convergence/divergence switch and filter, which in turn drives the extraocular muscle plant [9] to give the vergence response.

This MATLAB model program provides the flexibility in block and program manipulation, powerful supporting signal analysis routines, ease-of-use, and robustness that have been lacking in previous vergence models. Indeed, it can forms the basis for the investigation of the fundamental properties of the vergence system under a consistent and coherent framework.

2. Mode of availability

The intent of this effort is to make available to researchers and clinicians a relatively easy-to-use program for simulating vergence system dynamics in the MATLAB environment that is robust to a variety of stimulus conditions. Programs can be obtained via requests accompanied by a 3.5 in. 1.44 MB diskette or by e-mail (shoane@rci.rutgers.edu).

3. Program description

3.1. Overall program

The principles of the MATLAB model program of the vergence eye movement system are based on the program previously written in FORTRAN and presented in Hung et al. [6]. It has been modified and improved upon. For example, a ZOH element (which holds the input value for one sampling interval) has been incorporated in

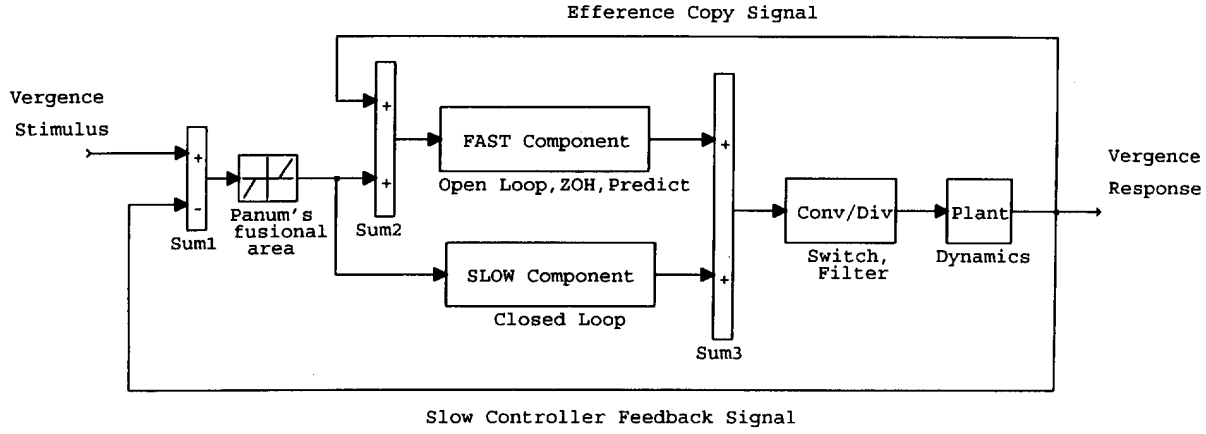

Fig. 1. Block diagram of the vergence system used in the MATLAB simulations. The difference between vergence stimulus and response, or vergence error, is input to a deadspace element, which represents Panum's fusional area. The output of the deadspace element is summed with the efference copy signal, resulting in a signal equal to the actual stimulus. This signal is used to drive the fast component. The fast component operates in an open-loop manner, uses a ZOH, and has a predictive capability for periodic stimuli. The fast component output is input to the convergence/divergence switch and filter, which in turn drives the oculomotor plant. The output of the plant provides the efference copy signal, which takes into account the effect of the filter and plant. The output of the deadspace element also drives the slow component. The slow component operates over a smaller range of vergence error amplitudes and velocities. The output of the slow component is input to the convergence/divergence switch and filter, which in turn drives the oculomotor plant. The output of the plant is fed back and is subtracted from the vergence stimulus to provide the error signal to the deadspace element. Thus, the slow component operates under a closed-loop condition. The fast and slow components operate over different stimulus regimes so that when one is active, the other is disabled. This provides robustness in the model responses.

the program in place of the sampler and target trajectory estimator in the original model. Also, it now automatically detects periodicity in the stimulus and changes to a predictive mode to reduce the phase lag between stimulus and response. In addition, the plant has been replaced by Robinson's [9] model for extraocular muscles.

The vergence controller consists of a fast and a slow component. The sum of their outputs provides the command signal. The fast component (see FAST block, Fig. 1, and flow chart, Fig. 2A) is driven by the sum of the error signal (equal to the output of the disparity signal through the deadspace operator, limits = $\pm 0.1°$ [10], which represents Panum's fusional area [11]) and the efference copy signal from the fast component output. The efference copy signal takes into account the effect of plant dynamics. This results in an open-loop signal that is equal to the original stimulus amplitude. This open-loop drive is important because it maintains stability in the presence of a relatively long latency ($\cong 200$ ms) and the requirement of an accurate initial step response. Such accuracy corresponds to very high gain in a feedback control system, which would have otherwise resulted in instability oscillations. The fast component open-loop response therefore accounts for most of the step response amplitude, with the remainder being taken up by the closed-loop slow component. The slow component (see SLOW, Fig. 1; and flow chart, Fig. 2B) operates over smaller amplitude and velocity ranges and uses negative feedback to provide the error signal for the controller. Significantly, the FAST and SLOW components operate under separate stimulus regimes, so that when one is active the other component is disabled. This provides robustness of the vergence response, since there is no cross interference (other than the residual decay of the slow component when it is disabled) between the fast and slow component outputs. Indeed, in test simulations (not shown) where both fast and slow components are simultaneously active, the model exhibits unstable, non-robust, and sometimes

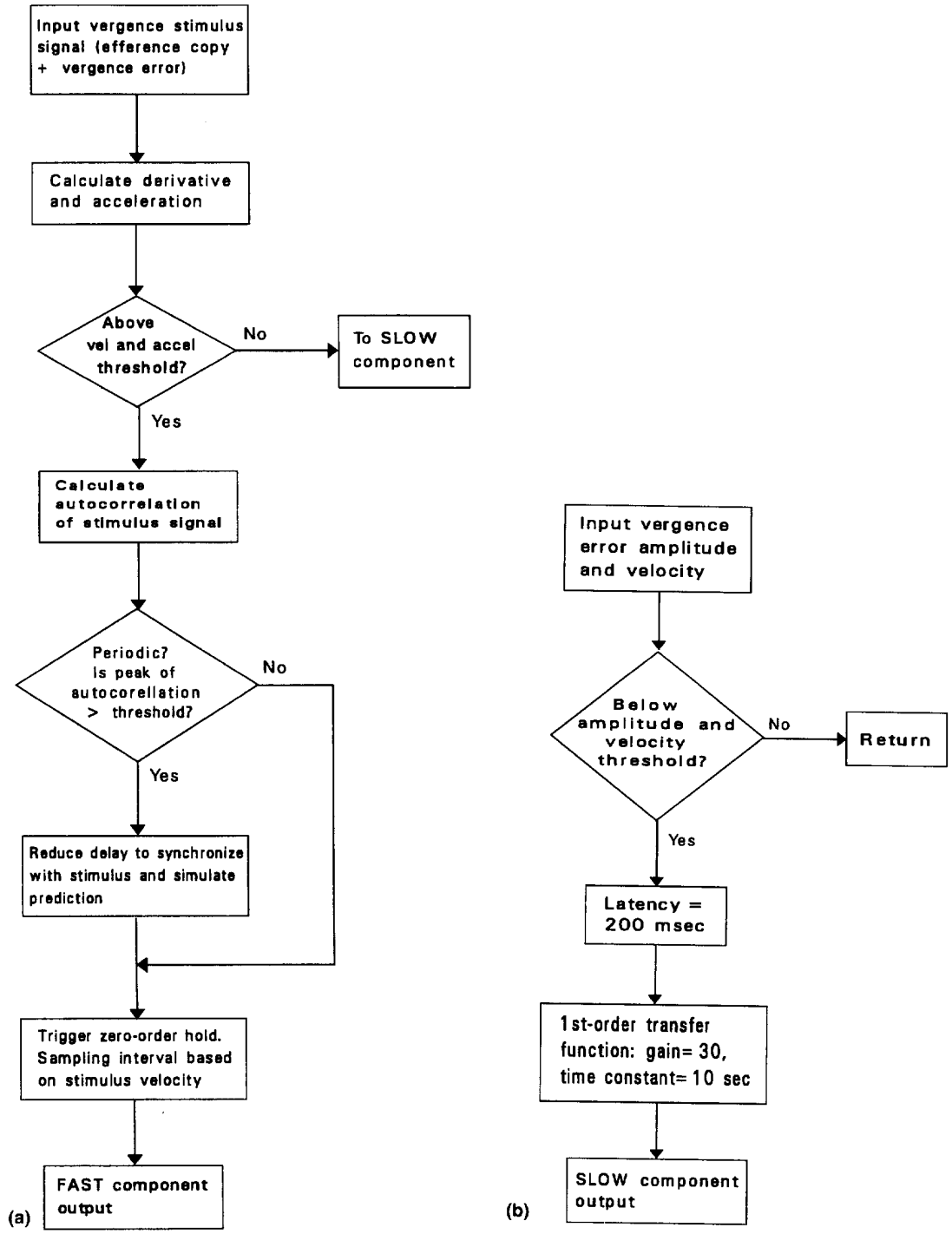

Fig. 2. Flow charts for the (A) FAST and (B) SLOW elements of the model shown in Fig. 1.

Table 1
Selected model parameters

Parameter	Value	Description
Athrsh	2500	Acceleration threshold (degrees/s) per degree for trigger of fast component for pulse and step
Pcross	0.05 or 0.20	Percentage of period for maintenance of periodicity of cross-correlation between stim and resp, for low and high vel, respectively
Pthrsh	0.75	Amplitude threshold (degree) for trigger of fast component for ramp and sine
Samptime	0.2–0.5	Sampling interval (s) of ZOH for high to low stimulus velocities, respectively
Tflag	0 or 1	Global flag for trigger of fast component and disable of slow component, and vice versa
Thrsh1	0.25 or 0.45	Threshold for initial detection of peak of autocorrelation, for low and high velocities, respectively
Thrsh2	0.15 or 0.40	Threshold for detection of maintained peak of autocorrelation, for low and high velocities, respectively
Vthrsh	2.50	Velocity threshold (deg/s) for trigger of fast component for ramp and sine

bizarre responses. The outputs of the fast and slow components are summed to provide the input to a convergence/divergence switch and filter. The switching is based on the sign of the velocity. The output of the switch and filter provides the command signal that drives the oculomotor plant. The following describes in detail the Fast.m function in the FAST block and Slow.m function in the SLOW block (Fig. 1).

3.2. Detailed description of FAST and SLOW components

The fast component function, Fast.m, consists of a ZOH element which responds to changing stimuli, such as ramps that are above the position (Pthrsh) and the velocity (Vthrsh) thresholds, or steps above the acceleration (Athrsh) threshold (see Table 1 and Fig. 2A). The sampling interval (Samptime) varies ranging from 0.2–0.5 s for high and low velocity ramp stimuli, respectively. A sudden subsequent rapid change in the stimulus, however, will re-trigger the ZOH. This provides model responses that are consistent with experimental ramp and square-wave responses. For the pulse stimulus, a double change in sign of acceleration (to distinguish it from the single change in sign for a step) within a 200 ms interval will result in a delayed (200 ms) pulse response that follows the time course of the pulse stimulus. This provides model responses that are consistent with the experimentally observed increase in response pulse amplitude and duration with increased stimulus pulse duration. For a sinusoidal stimulus, the fast component operates in conjunction with the slow component to provide the sinusoidal response. Moreover, the latency, which decreases from a maximum of 0.5 s for slow ramps to a minimum of 0.2 s for pulse, step, and fast ramps, is embedded in the Fast.m function.

Also within the function, Fast.m, the autocorrelation coefficients of the stimulus time course is calculated at each time step. The autocorrelation provides a means to quantify the periodicity in a waveform [12]. If a positive peak value (away from the zero shift position) is detected (correlation coefficient > Thrsh1), this is considered to have detected a periodic stimulus, such as a sinusoid. The periodic interval is equal to the time difference from zero shift to the shift at the positive peak. The stored periodic stimulus pattern, with a time shift advancement simulating prediction, is then used to drive the ZOH element of the fast component. The maintenance of periodicity is checked in two ways. First, the autocorrelation peak (away from the zero shift position) of the estimated stimulus timecourse is compared against a threshold (Thrsh2, which is slightly lower than that for detection). Second, the cross-correlation between the stimulus and response is calculated once every period to determine whether there is a large phase shift (i.e. whether the central peak is deviated from the center by a threshold amount by Pcross percent of the period). If the threshold is exceeded in either of the above conditions, the periodicity is considered to have been

lost. If such a subsequent loss of periodicity is detected, the operation mode is reverted back to normal and is ready to detect any subsequent periodicity.

The slow component function, Slow.m, responds to changing stimuli below the position (Pthrsh) and velocity (Vthrsh) thresholds (see Table 1 and Fig. 2B). These include slow ramps, low frequency sinusoids, and the approximately level position at the end of step responses. The slow component latency = 0.20 s, gain = 30, and 1st-order element time constant = 10 s [13]. The slow component uses negative feedback to fine-tune the response. This simulates the experimentally observed fusional process [1]. The fast and slow component functions, Fast.m and Slow.m, communicate via the global parameter Tflag. When Tflag = 1, the fast component is active and the slow component is disabled. On the other hand, when Tflag = 0, the opposite condition holds.

Thus, the fast and slow components are placed in parallel in the forward loop, with each operating over a range of stimulus amplitudes and velocities. The fast and slow component outputs are summed to provide the signal to the convergence/divergence switch and 1st-order filter (time constant = 0.07 s for convergence and 0.15 s to divergence; note that these element are input to the plant to provide the overall dynamics), whose output provides the command signal that drives the oculomotor plant [9]. The plant output serves as the efference copy signal which is fed back and summed with the error signal to provide the input to the fast component. The signal from the plant, rather than the fast component output, is used because it provides a more accurate value for the fast component output. This is appropriate since only one component (fast or slow) is active at a time, and little or no inappropriate slow component residual signal is present in the efference copy signal. For the slow component, its output is fed back as a negative feedback signal and combined with the stimulus to result in the error signal that serves as the input to the slow component.

4. Implementation

Simulations were performed on an 486-PC with 8 MB RAM operating at 66 MHz, Windows 3.10, MATLAB 4.2c, and SIMULINK 1.3c. Stimulus functions that were not available in Blocklib.m of SIMULINK, such as pulses, ramp-stop, and square-wave (above zero level), were written as m-files and called by s-function blocks.

5. Model simulation procedure

Simulations were performed for various parameter values for pulse, ramp, square- and sine-wave stimuli. In the initial simulation process, the parameters were first set at approximately the middle of the range of possible values (e.g. threshold ranges found experimentally [6,14]) to provide initial simulation responses. Then, the parameters were fine-tuned to provide appropriate responses to all the different stimulus conditions used. Following the simulations, the results were plotted on a Laserjet4 printer.

6. Simulation results

The model simulation results exhibited behavior that was similar to that found experimentally [6,15]. Pulse responses showed an increase in amplitude and duration with increased pulse duration (Fig. 3). The slow component played a relatively greater role for the shorter (0.02 and 0.04 s) rather than the longer duration pulses. This was because the slow component operated over smaller disparity amplitude and velocity ranges. For the longer duration stimuli, the fast component provided the primary contribution to the up and down steps in the pulse response. The ramp responses showed transitions from smooth tracking by the slow component for slower ramps (1 and 2°/s), to staircase steps by the fast component for intermediate velocity ramps (3, 5 and 10°/s), and single steps by the fast component for higher velocity ramp and step stimuli. This held true for both positive (Fig. 4) and negative (Fig. 5) ramps, with convergence exhibiting faster dy-

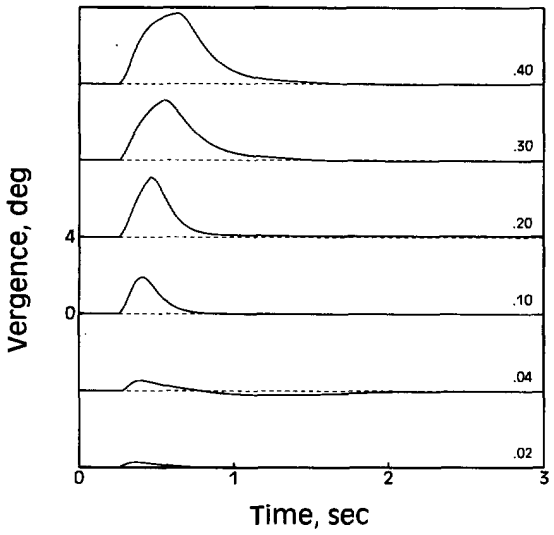

Fig. 3. Model responses to pulse stimuli (duration in s shown at right of traces) of 4° amplitude.

namics than divergence responses. Note also that there was a gradual decrease in latency with increasing ramp velocity, which was consistent with experimental results [6,15]. Square-wave responses showed a transition from a non-predictive mode

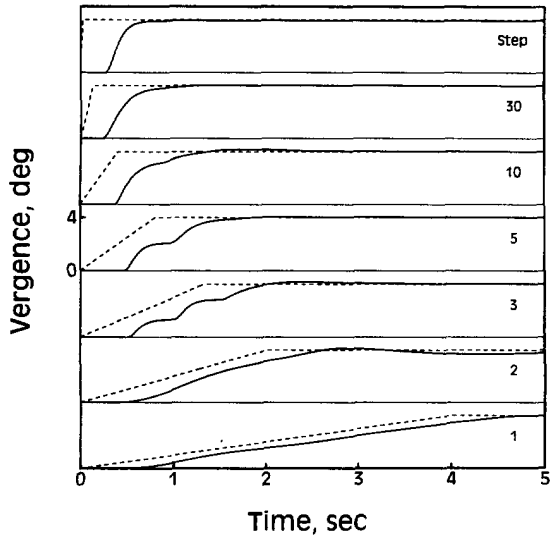

Fig. 4. Model responses to positive (convergent) ramp stimuli (velocity in degrees/s shown at right of traces) with maximum amplitude of 4°. - - -, stimulus; ——, response.

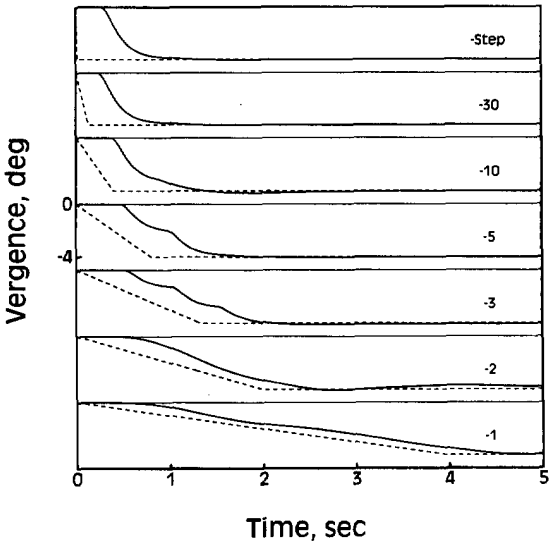

Fig. 5. Model responses to negative (divergent) ramp stimuli (velocity in degrees/s shown at right of traces) with minimum amplitude of −4° s. Responses show a similar trend as that for positive ramp stimuli (see Fig. 4). - - -, stimulus; ——, response.

for the low frequency stimulus (0.25 Hz) to a predictive mode for higher frequency stimuli (0.50 and 0.75 Hz) (Fig. 6). For the low frequency square-wave, the latencies for the convergent and divergent responses remained the same for the repeated square-wave stimulus. However, for the

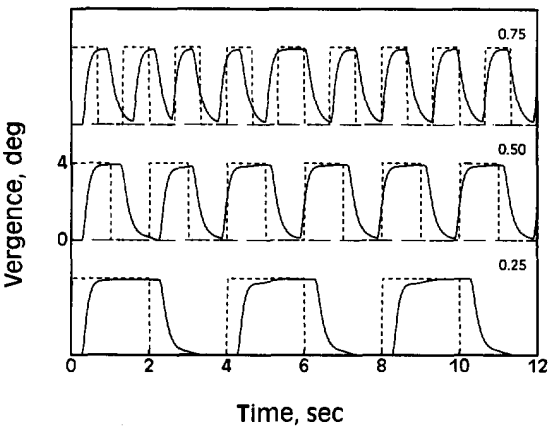

Fig. 6. Model responses to square-wave stimulation (frequency in Hz shown at right of traces) of 4° amplitude. - - -, stimulus; ——, response.

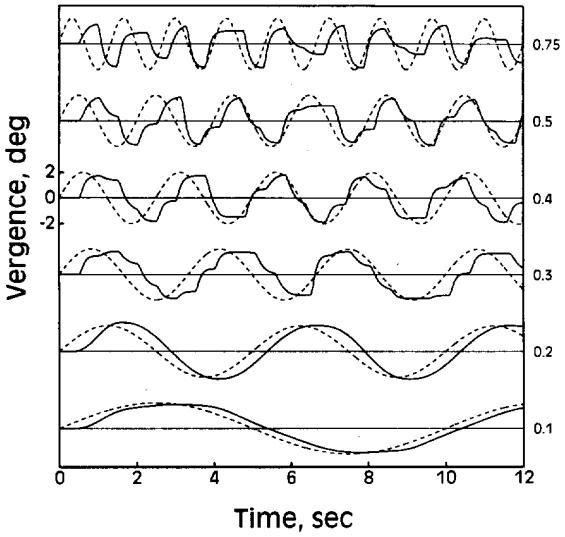

Fig. 7. Model responses to sine-wave stimulation (frequency in Hz shown at right of traces) for ± 2° peak-to-peak amplitude. - - -, stimulus; ——, response.

higher frequency square-waves, after about 2 cycles, the predictive mode came into effect, resulting in a reduction in latency so that the response approximately coincided with the on- or offset of the stimulus. The lack of prediction for the low frequency stimulus was because the period of the stimulus was at the limit of the memory length allotted for the predictive mode. Sine-wave responses showed a transition from smooth tracking by the slow component for lower frequency stimuli (0.1 and 0.2 Hz) to combined fast and slow component responses for higher frequency stimuli (above 0.2 Hz) (Fig. 7). Also, there was a transition from a non-predictive mode for the lower frequency stimuli to a predictive mode for the higher frequency stimuli. As in the square-wave responses, after about 2 cycles, the predictive mode came into effect and there was a reduction in latency in the sine-wave responses. All the above simulation responses were consistent with experimental results [5,6,15–17].

7. Discussion

The time required for the simulations was found to be about 40 s of real time per s of model simulated time. This relatively long computation time was due to the complex checks that were involved in the Fast.m function (see Section 3.2). This included the shifting and storage of stimulus and response waveforms at each time step (0.02 s), autocorrelation of the stimulus waveform at each time step, cross-correlation between stimulus and response after one period, and checking for the peaks in these functions. With newer PC's or other computers operating at 200 MHz or more, the actual time for the simulations could be reduced by a factor of at least three. In contrast, such software steps for correlations and predictive responses have their analogs in the neural networks of the higher brain centers in humans [18] that provide rapid responses to periodic stimuli in everyday life.

The model parameters were tested and fine-tuned to give good overall performance under different stimulus conditions. Hence, they represented compromise values that gave consistent performance across the different conditions. Indeed, simulation of the model showed results that were very similar to experimental data for different stimuli under a variety of conditions [6,15]. This provided a relatively comprehensive check on the accuracy and robustness of the model. The most complex part of the model was the Fast.m function used to drive the fast component in the model. It consisted of amplitude and velocity thresholds, ZOH, auto- and cross-correlation analyses, and predictive response. It was designed to be a general sampled data and predictive program, where the values of the particular parameters determine the dynamic characteristics of the responses. Hence, modification of the parameters could allow the program to be used for simulating other oculomotor systems.

There were several important assumptions in the model. The fast component was assumed to behave as a ZOH element. This provided the multiple step responses to intermediate velocity ramps seen experimentally. Each of these steps have been shown to fall on the main sequence for

vergence steps [5,15]. Without such an element, an oscillatory (i.e. sinewave-like) ramp response would not resemble the experimental data, and significantly, individual movements would not fall on the main sequence. Although Rashbass and Westheimer [1] termed their responses to ramp stimuli 'oscillations', these have been shown to be multiple-step responses [15]. Another assumption was that the efference copy signal contained the plant dynamics following the output of the fast component. Such a process may occur in the brain by means of a neural network trained to represent the overall dynamics of the eye movement [19]. This would provide an appropriate representation of the eye movement response, which when summed with the disparity signal, gave an accurate representation of the stimulus. The third major assumption was that when the fast component was active, the slow component was disabled, and vice versa. This distinct temporal separation of fast and slow component control was critical in that it permitted reasonable and robust estimation of the parameters. Otherwise, if both fast and slow components were simultaneously active, with one being open-loop and the other closed-loop, unknown and unusual behaviour often occurred, with an inability to control system performance as parameters were varied. The separation of the fast and slow component was consistent with neurophysiological results in monkeys showing near and far cells that respond maximally to targets 1° or more either in front of or behind the fixation plane, and tuned excitatory and inhibitory cells that respond maximally to targets near the fixation plane [20]. These corresponded to stimulus conditions for driving the fast and slow components, respectively. Finally, it was assumed that the peak (away from the zero shift position) of the autocorrelation function for the estimated stimulus time course could be used to assess periodicity [12]. This signal processing procedure was reasonable since the peak was relatively independent of the shape of the waveform, but was dependent on its periodicity. Also, the height of the peak gave a measure of the accuracy of the estimated period.

The saccadic eye movement system has been shown to exhibit sampled data behavior [12], and the pursuit system has been shown to exhibit prediction [12,21]. Also, the accommodative system has been shown to exhibit sampling properties [22]. As indicated above, since the Fast.m function for the vergence system contains sampling and prediction, it could be modified to reflect the characteristics of the versional and accommodation systems.

The new MATLAB model of the vergence eye movement system has significant improvements over the previous FORTRAN program model. The block diagram environment can implement more easily some important features such as deadspace operator, efference copy, slow component feedback, and plant dynamics. The m-file structure and the SIMULINK signal analysis programs allows for stable and robust implementation of complex functions such as ZOH, auto- and cross-correlation, and prediction. Also, modifications for checking and testing the model is more easily performed by simply modifying the m-file parameters and block connections. Moreover, the MATLAB platform is easily transportable from one system to another. The SIMULINK block diagram environment provides ease of use in generating stimuli, modifying connections, and displaying and recording simulated data. Thus, this program gives the researcher and clinician a relatively simple, yet powerful and versatile, tool for investigating the dynamic behavior of the vergence eye movement system.

One of the long-term goals in the development of the model is to construct a combined dynamic model of the accommodation and vergence systems in a MATLAB environment. At present, no comparable accommodation model is available. When such a combined model is constructed, it will provide a comprehensive overall model that can describe the dynamic behavior of the near response under a variety of blur and disparity cue conditions. Also, parameter variations in the model can be used in the diagnosis of clinical deficits by simulating the dynamic characteristics in amblyopia, strabismus, and near-work symptoms [23]. The other long-term goal is to construct a combined dynamic model of the versional (saccade plus pursuit) and vergence system. Some existing models will serve as the basis for the

development of this comprehensive model [24–26]. This will be used to quantify the responses to asymmetrical target displacements in space and to investigate some fundamental properties of the interactions between neurological control systems.

References

[1] C. Rashbass, G. Westheimer, Disjunctive eye movements, J. Physiol. (London) 159 (1961) 339–360.

[2] J.L. Semmlow, G.K. Hung, The near response: theories of control, in: C.M. Schor, K.J. Ciuffreda (Eds.), Vergence Eye Movements: Basic and Clinical Aspect, chap. 6, Butterworths, Boston, MA, 1983, pp. 175–195.

[3] J.L. Semmlow, P. Wetzel, Dynamic contributions of the components of binocular vergence, J. Opt. Sci. Am. 69 (1979) 639–645.

[4] A.T. Bahill, L. Stark, The trajectory of saccadic eye movements, Sci. Am. 240 (1979) 108–114.

[5] G.K. Hung, H.M. Zhu, K.J. Ciuffreda, Convergence and divergence exhibit different response characteristics to symmetric stimuli, Vis. Res. 37 (1997) 1197–1205.

[6] G.K. Hung, J.L. Semmlow, K.J. Ciuffreda, A dual-mode dynamic model of the vergence eye movement system, IEEE Trans. Biomed. Eng. 33 (1986) 1021–1028.

[7] D.S. Zee, L. Levi, Neurological aspects of vergence eye movements, Rev. Neurol. 145 (1989) 613–620.

[8] J.R. Duhamel, C.R. Colb, M.E. Goldberg, The updating of the representation of visual space in parietal cortex by intended eye movements, Science 255 (1992) 90–92.

[9] D.A. Robinson, Models of the saccadic eye movement control system, Kybernetik. 14 (1973) 71–83.

[10] G.K. Hung, K.J. Ciuffreda, Sensitivity analysis of relative accommodation and vergence, IEEE Trans. Biomed. Eng. 41 (1994) 241–248.

[11] P.L. Panum, Physiologische Untersuchungen uber das Sehen mit zwei Augen, Schwersche Buchhandlung, Kiel, Germany, 1858.

[12] L. Stark, Neurological Control Systems, Studies in Bioengineering, Plenum, New York, 1968, pp. 60–62, 236–270.

[13] V.V. Krishnan, L. Stark, A heuristic model of the human vergence eye movement system, IEEE Trans. Biomed. Eng. 24 (1977) 44–49.

[14] G.K. Hung, J.L. Semmlow, Static behavior of accommodation and vergence: computer simulation of an interactive dual-feedback system, IEEE Trans. Biomed. Eng. 27 (1980) 439–447.

[15] J.L. Semmlow, G.K. Hung, K.J. Ciuffreda, Quantitative assessment of disparity vergence components, Invest. Ophthal. Vis. Sci. 27 (1986) 558–564.

[16] B.L. Zuber, L. Stark, Dynamic characteristics of the fusional vergence eye movement system, IEEE Trans. Sys. Sci. Cybern. 4 (1968) 72–79.

[17] V.V. Krishnan, F. Farazian, L. Stark, An analysis of latencies and prediction in the fusional vergence system, Am. J. Optom. Arch. Am. Acad. Optom. 50 (1973) 933–939.

[18] J. Droulez, A. Berthoz, A neural network model of sensoritopic maps with predictive short term memory properties, Proc. Nat. Acad. Sci. USA 88 (1991) 9653–9657.

[19] K.P. Krommenhoek, A.J. van Opstal, C.C.A.M. Gielen, J.A.M. van Gisbergen, Remapping of neural activity in the motor colliculus: a neural network study, Vis. Res. 33 (1993) 1287–1298.

[20] G.F. Poggio, B. Fischer, Binocular interaction and depth sensitivity in striate cortex of behaving Rhesus monkey, J. Neurophysiol. 40 (1977) 1392–1405.

[21] E. Kowler, R.M. Steinman, The effect of expectation on slow oculomotor control. 1 Periodic target steps, Vis. Res. 19 (1979) 619–632.

[22] G.K. Hung, K.J. Ciuffreda, J.L. Semmlow, J.L. Horng, Vergence eye movements under natural viewing conditions, Invest. Ophthalmol. Vis. Sci. 35 (1994) 3486–3492.

[23] J.R. Griffin, Binocular Anomalies—Procedures for Vision Therapy, Professional Press, Chicago, IL, 1976.

[24] D.S. Zee, E.J. Fitzgibbon, L.M. Optican, Saccade-vergence interactions in humans, J. Neurophysiol. 68 (1992) 1624–1641.

[25] G.K. Hung, K.J. Ciuffreda, Schematic model of saccade-vergence interactions, Med. Sci. Res. 24 (1996) 813–816.

[26] S.S. Patel, H. Ogmen, J.M. White, B.C. Jiang, Neural network model of short-term horizontal disparity vergence dynamics, Vis. Res. 37 (1997) 1383–1399.