Sliding Mode Control of Real-Time PNU Vehicle Driving Simulator and Its Performance Evaluation*

Min Cheol LEE**, Min Kyu PARK***, Wan Suk YOO****,
Kwon SON[†] and Myung Chul HAN^{††}

This paper introduces an economical and effective full-scale driving simulator for study of human sensibility and development of new vehicle parts and its control. Real-time robust control to accurately reappear a various vehicle motion may be a difficult task because the motion platform is the nonlinear complex system. This study proposes the sliding mode controller with a perturbation compensator using observer-based fuzzy adaptive network (FAN). This control algorithm is designed to solve the chattering problem of a sliding mode control and to select the adequate fuzzy parameters of the perturbation compensator. For evaluating the trajectory control performance of the proposed approach, a tracking control of the developed simulator named PNUVDS is experimentally carried out. And then, the driving performance of the simulator is evaluated by using human perception and sensibility of some drivers in various driving conditions.

Key Words: Vehicle Driving Simulator, Sliding Mode Control, Fuzzy Adaptive Network, Perturbation Compensator

1. Introduction

Recently, the interest in a realistic vehicle driving simulator has been increased. The driving simulator⁽¹⁾⁻⁽³⁾ is a virtual reality device in which a human being on board is able to feel as if they are actually driving a vehicle. Also, this simulator enables us to research and analyze the interaction between a driver and a vehicle, to develop the new vehicle systems, and so on. Therefore, it has been widely used by many automotive makers and research centers.

The primitive form of the driving simulator was developed in the 1970s⁽²⁾, and has been gradually improved due to the progression of computer technologies. Recently, the development of the NADS (national advanced driving simulator) in the United States has begun and was completed in 1999⁽³⁾. However, a full-scale driving simulator is so expensive because vehicle analysis system and graphic generation system have been composed of workstation-based systems and commercial control hardware and software.

This paper introduces an economical and effective full-scale driving simulator for study of human sensibility and development of new vehicle parts. The developed PNUVDS (Pusan National University Vehicle Driving Simulator) consists of several subsystems: the motion platform of 6 degrees of freedom parallel mechanism reappearing a vehicle motion, the motion controller based on TMS320C31 processor, the visual and audio system reproducing virtual environments, the PC-based vehicle dynamic analysis system to perform real-time simulation of vehicle dynamics, and the vehicle operating system acting as an interface between a driver and a simu-

* Received 18th October, 2002 (No. 02-5171)

** School of Mechanical Engineering, Pusan National University, San, 30, Jangjeon-dong, Kumjeong-gu, Pusan, 609-735, Korea. E-mail: mclee@pusan.ac.kr

*** Electro-Mechanical Research Institute, Hyundai Heavy Industry, 102-18, Mabbok-ri, Kuseong-eup, Yongin-shi, Gyunggi-do, 449-716, Korea. E-mail: mkpark@hhi.co.kr

**** School of Mechanical Engineering, Pusan National University. E-mail: wsyoo@pusan.ac.kr

[†] School of Mechanical Engineering, Pusan National University. E-mail: kson@pusan.ac.kr

^{††} School of Mechanical Engineering, Pusan National University. E-mail: mchan@pusan.ac.kr

lator.

The control of the motion platform is very significant to accurately reappear a various vehicle motion when maneuvering a vehicle driving simulator. However, real-time robust control may be a difficult task because the motion platform is the nonlinear complex system^{(4),(5)}. Sliding mode control is very attractive method for nonlinear systems^{(6),(7)}. It has been confirmed as an effectively robust control approach for nonlinear systems against parameters and load variations. However, its implementation in practice has been caused an inherent chattering problem. Therefore, this study proposes the sliding mode controller with perturbation compensator using the observer-based fuzzy adaptive network (FAN) to solve the chattering problem. This control algorithm is designed to select the adequate fuzzy parameters of the perturbation compensator. The perturbation compensator generates the control input for compensating the unmodeled dynamic terms and disturbance as using the observer-based FAN. The weighting parameters of the FAN are updated in on-line by the adaptive law in order to force the estimation errors to converge to zero. The robust and intelligent routine is realized by the combination of sliding mode control and FAN. For evaluating the trajectory control performance of the proposed approach, a tracking control of the motion platform is experimentally carried out by using the predicted reference trajectories through a vehicle dynamic simulation. And then, the driving performance of the developed simulator is evaluated by using human perception and sensibility of some drivers in various driving conditions.

2. System Configuration

Figure 1 shows a schematic diagram of the developed vehicle driving simulator. The driving information is generated by the vehicle operating systems such as a steering wheel, a brake pedal, and an acceleration pedal. This information is transferred to the main computer, Intel Pentium 4 2 GHz, through the RS232C. Then, the vehicle dynamics is simulated to predict the vehicle motion in real-time. The graphic system based in Linux generates the graphic images and sounds of the driving environment using the data from the vehicle dynamic analysis. Generated graphic images and sounds are presented by a projector and a loud-speaker, respectively. The controller based on the TMS320C31 controls the motion platform.

2.1 Motion platform system

The motion platform system is composed of a motion platform, a hydraulic power system and a motion controller. The platform is a Stewart plat-

Fig. 1 Schematic diagram of the driving simulator

Fig. 2 Motion platform

form^{(4),(5)} which is a closed loop form, and it provides better load-to-weight ratio and rigidity than a serial manipulator. This platform has also high positional accuracy because position errors of the actuators are not additive. The hydraulic power system which includes a power unit, actuators, servovalves, and other components has been designed through analysis of the inverse dynamics of the platform.

The maximum payload of motion platform was set at 750 kg. Figure 2 shows the motion platform with the vehicle operating system. The controller was developed to control the position of the motion platform in real-time. The control board is composed of a micro-processor and peripheral circuits with 32 bits-DIO (digital input and output), a 16 channel ADC (analog to digital converter), 6 channels DAC (digital to analog converter), a serial communication circuit, and timers. The micro-processor of the control board is the TMS320C31.

Fig. 3 Screen of driving environment

2.2 Visual system

A visual system is significant part in a driving simulator because drivers on board recognize driving environments with their eyes. Therefore, the graphic scenes with high resolution are needed to feel the virtual driving environments as real ones. A driving environment is generated by using the Performer engine and various graphic software such as 3D MAX, Photoshop, Rhino 3D, and so on⁽⁸⁾.

A visual image is presented by the texture mapping technique. Texture mapping is a powerful technique for adding realism to a computer-generated image. And, the images are transformed into a 3 dimensional model by using the 3D MAX software. Finally, a virtual environment is generated by rendering the model continuously. In this study, the Performer is used for rendering in real-time. Figure 3 shows a virtual map of the Pusan National University. The map consists of 13 000 mesh and 197 3-dimensional object model. To possess a realistic driving feeling, a virtual driving environment must be very similar to the real world. However, since it is very difficult to process this environment in real-time, the virtual driving environment is set by compromising between a realistic driving feeling and real time ability.

2.3 Vehicle dynamic analysis system

The commercial programs cannot be used in the driving simulator, because of the lack of real-time ability. Therefore, in this paper, the equations of motion for a compact vehicle is derived using velocity transformation method⁽⁹⁾. And the massless linkage technique is used for reducing a degree of freedom of the vehicle⁽¹⁰⁾. Using the velocity transformation method, the generality of the Cartesian coordinates

Table 1 Total degree of freedom of vehicle

Body	:	14	×	(+6)	=	+84D.O.F
Spherical Joint	:	4	×	(-3)	=	-12D.O.F
Translational Joint	:	3	×	(-5)	=	-12D.O.F
Revolute Joint	:	7	×	(-5)	=	-35D.O.F
Spherical-Massless Link						
Spherical	:	3	×	(-1)	=	-3D.O.F
Revolute-Spherical Massless Link						
Revolute-Spherical	:	2	×	(-2)	=	-4D.O.F
Total Degrees of Freedom						= 15D.O.F

Fig. 4 The mechanism of the simplified model

and the efficiency of the joint coordinates can be obtained. Figure 4 shows the mechanism of a simplified vehicle. This is composed of 14 bodies and 5 massless linkages as shown in Fig. 4. The front and rear wheel parts of the vehicle are composed of macpherson suspension and twisted axle, respectively. And the lateral rod is attached in rear wheel part for supporting a lateral force. The lower control arm of the macpherson suspension is presented by a revolute-spherical massless linkage. And two tie-rod and lateral rod are also spherical-spherical massless linkage. Table 1 shows the degree of freedom of the developed vehicle model. And, in the case of a tire model, AT_D full tire model of AutoDyn7⁽¹⁰⁾ is used in this study.

2.4 System integration

The system integration technology is a key factor in determining the fidelity and performance of the driving simulator. Therefore, the method of integrating the subsystems was carefully planned in the early stages of this study.

The configuration of the communication network in the driving simulator is shown in Fig. 5. Informations from the vehicle operation system which includes a steering wheel, a brake pedal, and an

Fig. 5 Communication network

accelerator pedal are transferred by a serial communication to the operating computer for analysis of the vehicle motion. The simulation for the vehicle dynamics is carried out using this information. The results from this simulation are transferred by LAN to the graphics computer for the processing of the graphic image.

The reference trajectories for the motion platform are generated by a washout algorithm⁽¹¹⁾. And then, the trajectories in space-coordinates from the washout algorithm are transformed into those in joint-coordinates by the inverse kinematics of 6 DOF parallel manipulator. And, these data are transferred into motion controller for control of the motion platform. Also, the actuator lengths are measured through AD converter in the controller.

3. Control of Motion Platform

3.1 System modeling and observer design

The 2nd order nominal model of the platform is expressed as

$$M_{TL}\ddot{x} + C_{TL}\dot{x} = u + \Psi \quad (1)$$

where M_{TL} and C_{TL} are the equivalent mass and the equivalent damp, respectively, Ψ is composed of unmodel terms, nonlinear terms, and external disturbance, and u is control input. Equation (1) is re-expressed as

$$M_{TL}\ddot{x} + C_{TL}\dot{x} = \mu \quad (2)$$

$$\mu = u + \hat{\Psi}$$

where μ and $\hat{\Psi}$ are the modified control input and the estimated perturbation, respectively.

The observer task is to derive the state toward a desired state in spite of these perturbations. It is assumed that the perturbations are bounded in upper limit by a known continuous function of the state. A sliding state observer for SISO system is a robust observer which estimates the state of a nonlinear system. The state space representation of a second order SISO system is as follows:

$$\dot{x}_1 = x_2 \quad (3)$$

$$\dot{x}_2 = -\frac{C_{TL}}{M_{TL}}x_2 + \frac{1}{M_{TL}}\mu$$

where $x = [x_1, x_2]$ is the state vector. The observer task is to estimate the state x in spite of the uncertainties. The sliding state observer is presented as

$$\dot{\tilde{x}}_1 = \tilde{x}_2 - k_1 \text{sat}(\tilde{x}_1) - \alpha_1 \tilde{x} \quad (4)$$

$$\dot{\tilde{x}}_2 = -\frac{C_{TL}}{M_{TL}}\tilde{x}_2 + \frac{1}{M_{TL}}\mu - k_2 \text{sat}(\tilde{x}_1) - \alpha_2 \tilde{x}_1$$

where k_1, k_2, α_1 , and α_2 are positive number and $\tilde{x}_1 = \tilde{x}_1 - x_1$ is the estimation position error. Throughout the text of equations, " $\sim$ " refers to estimation errors whereas " $\hat{\cdot}$ " symbolizes the estimated quantity. And, the saturation function is defined as

$$\text{sat}(\tilde{x}_1) = \begin{cases} \tilde{x}_1/|\tilde{x}_1|, & \text{if } |\tilde{x}_1| \geq \varepsilon_{oj} \\ \tilde{x}_1/\varepsilon_{oj}, & \text{if } |\tilde{x}_1| \leq \varepsilon_{oj} \end{cases}$$

where ε_{oj} is small observer constant as the boundary layer of sliding mode. Using Eqs.(3) and (4), the estimation error dynamics are derived as

$$\dot{\tilde{x}}_1 = \tilde{x}_2 - k_1 \text{sat}(\tilde{x}_1) - \alpha_1 \tilde{x}_1 \quad (5)$$

$$\dot{\tilde{x}}_2 = -k_2 \text{sat}(\tilde{x}_1) - \alpha_2 \tilde{x}_1 - \frac{C_{TL}}{M_{TL}}\tilde{x}_2.$$

3.2 Sliding mode control

In this section, a sliding mode controller is designed. The estimation sliding function is defined as

$$\hat{s}_j = \hat{e}_j + c_j \hat{e}_j \quad (6)$$

where c_j is the desired control bandwidth and always positive, $\hat{e}_j (= \hat{x}_{1dj} - x_{1dj})$ is the estimation position error, and $[x_{1dj} \ x_{1dj}]^T$ is the desired motion cue for the j th degree of freedom. The actual sliding function is presented by

$$s_j = \dot{e}_j + c_j e_j \quad (7)$$

where $e_j (= x_{1j} - x_{1dj})$ is the actual position error.

The modified control input μ_j of Eq.(2) is selected by using time derivative of the Lyapunov function candidate being given by $\dot{\hat{s}}_j \hat{s}_j < 0$ to satisfy the boundary layer attraction condition.

A desired $\hat{s}_j$ is selected as

$$\dot{\hat{s}}_j = -K_j \text{sat}(\hat{s}_j) \quad (8)$$

where K_j is positive constant, and $\text{sat}(\hat{s}_j)$ is defined as

$$\text{sat}(\hat{s}_j) = \begin{cases} \hat{s}_j/|\hat{s}_j|, & \text{if } |\hat{s}_j| \geq \varepsilon_{cj} \\ \hat{s}_j/\varepsilon_{cj}, & \text{if } |\hat{s}_j| \leq \varepsilon_{cj}. \end{cases}$$

sat ($\hat{s}_j$) is effectively used for anti-chattering problem. ε_{cj} is small constant as the boundary layer of sliding mode.

Using the results of the previous sections such as Eqs. (4), (5), and (8), it is possible to compute control input as

$$\mu_j = M_{TL} \left\{ \frac{C_{TL}}{M_{TL}} \ddot{x}_2 - K_j \text{sat}(\hat{s}_j) - (k_{1j}/\varepsilon_{oj}) \tilde{x}_{2j} + [(k_{2j}/\varepsilon_{oj}) + c_j(k_{1j}/\varepsilon_{oj}) - (k_{1j}/\varepsilon_{oj})^2] \tilde{x}_{1j} + \ddot{x}_{1dj} - c_j(\tilde{x}_{2j} - \dot{x}_{1dj}) \right\}. \quad (9)$$

3.3 Perturbation compensator

The perturbation function (Ψ) in Eq. (1) is approximated by using the FAN. The parameters of the FAN for minimizing the estimation error and estimation velocity error of each actuator are adjusted by an adaptive control law. The fuzzy system consists of some fuzzy rules and a fuzzy inference system. The fuzzy inference system uses the fuzzy rules to perform a mapping from an input linguistic vector $z = [z_1 \ z_2 \ \dots \ z_n] \in R^n$ to an output linguistic variable $y \in R$. The i th fuzzy IF-THEN rule is written as

$$R^{(i)}: \text{if } z_1 \text{ is } A_1^i \text{ and } z_2 \text{ is } A_2^i \text{ and } \dots \text{ and } z_n \text{ is } A_n^i \text{ then } y \text{ is } B^i \quad (10)$$

where A_1^i, A_2^i, A_n^i , and B^i are fuzzy sets. First, ϕ_i which is calculated by using product inference is defined as

$$\phi_i = \prod_{j=1}^n \mu_{A_j^i}(z_j) \quad (11)$$

where $\mu_{A_j^i}(z_j)$ is membership function value of the fuzzy variable (z_j). Second, fuzzy basis value ($\bar{\phi}_i$) is obtained by using normalization process such as

$$\bar{\phi}_i = \frac{\phi_i}{\sum_{i=1}^h \phi_i}. \quad (12)$$

Finally, the output of fuzzy logic system is obtained by using singleton defuzzifier such as

$$y(z) = \sum_{i=1}^h \theta_i \bar{\phi}_i = \theta^T \bar{\phi} \quad (13)$$

where h is the number of the total fuzzy IF-THEN rules, $\theta^T = [\theta_1, \theta_2, \dots, \theta_h]$ is adjustable parameter vector, and $\bar{\phi}_i = [\bar{\phi}_1, \bar{\phi}_2, \dots, \bar{\phi}_h]$ is fuzzy basis vector. The fuzzy logic approximator based on neural networks is established. Figure 6 shows the configuration of the FAN.

This network has five layers. At layer 1, nodes represent the values of the membership function of total linguistic variables. Usually, a bell-shaped membership function is widely used. Every node at layer 2 multiplies the incoming signals and sends the product result to next layer. Every node at layer 3 calculates the fuzzy basis using normalization process. The i th node calculates the ratio of the i th rule's firing strength to the sum of all rules' ones. At layer 4, every node including weighting factor of adjustable parameter multiplies the fuzzy basis. At layer 5, the single node computes the overall output as the summation of all incoming signals.

Fig. 6 Structure of observer based FAN

Fig. 7 Perturbation compensator using observer based FAN

ter multiplies the fuzzy basis. At layer 5, the single node computes the overall output as the summation of all incoming signals.

Figure 7 shows the perturbation compensator using the observer based FAN. Input variables are estimation position and velocity, and output variable is the estimated perturbation. The perturbation function is presented by

$$\tilde{\Psi}_{FAN} = \theta^T \phi(\hat{x}) \quad (14)$$

The adaptive law for updating adjustable parameters is chosen as

$$\dot{\theta} = -\gamma(\bar{w}_1 \times \tilde{x}_1 + \bar{w}_2 \times \tilde{x}_2) \phi(\hat{x}) \quad (15)$$

where $\bar{w}_1$ and $\bar{w}_2$ are weighting factors, and γ is learning constant. Substituting the control input ($u = \mu - \tilde{\Psi}$) into the nominal dynamics Eq. (1), the following relationship is obtained as

$$\begin{aligned} M_{TL} \ddot{x} + C_{TL} \dot{x} &= \mu - \tilde{\Psi}_{FAN} + \Psi \\ M_{TL} \ddot{x} + C_{TL} \dot{x} - \mu &= \Psi - \tilde{\Psi}_{FAN} = \tilde{\Psi}. \end{aligned} \quad (16)$$

Finally, the perturbation function is represented by

$$\tilde{\Psi} = \tilde{\Psi}_{FAN} + \tilde{\Psi}. \quad (17)$$

Figure 8 shows the overall scheme of the

proposed sliding mode control with the perturbation compensator using the observer based FAN.

3.4 Experiment

A simple tracking control is performed to evaluate the proposed control algorithm. The tuned sliding state observer parameters $k_1, k_2, \alpha_1, \alpha_2$, and ε_{oj} in Eq. (4) are 0.12, 2.4, 3, 5, and 0.002, respectively. And, the tuned control parameters c_j, K_j , and ε_{cj} in Eq. (9) are 20, 50, and 0.02, respectively. The layer structure of perturbation compensator using the observer based FAN is a 2-6-9-9-1 as shown in Fig. 7. The input variables are an estimated cylinder position and velocity. The output variable is the estimated perturbation. The observer based FAN has 6 membership functions which are the bell shapes.

Input variables of the FAN are the normalized estimation position and velocity. The range of values for these variables is from -1 to 1 . And, a learning constant γ of Eq. (15) in on-line learning paradigm is set as 0.95. And the tuned weighting parameters $\bar{w}_1$ and $\bar{w}_2$ are 100 and 100, respectively.

The payload of the Stewart platform is about 500 kg. The sampling time interval for control is selected by 10 msec.

The reference position and velocity trajectory are shown in Fig. 9. The experimental results of the

Fig. 8 Sliding mode control with perturbation compensator

Fig. 9 Reference position and velocity trajectory

proposed sliding mode control algorithm in actuator 4 are shown in Figs. 10 - 14, which have larger errors than other actuators. Figures 10 and 11 show position error and velocity error for each cylinder, respectively. Figure 12 shows the estimated position errors. And, the compensated perturbations for each cylinder are shown in Fig. 13. Figure 14 shows the control

Fig. 10 Position error

Fig. 11 Velocity error

Fig. 12 Estimated position error

Fig. 14 Control input of the proposed control algorithm

Fig. 13 Compensated perturbation

Fig. 15 Position error without perturbation compensator

input. The estimated error is converged within 1.2 mm as shown in Fig. 12. And, the maximum position error is 3 mm and converged to zero as shown in Fig. 10. However, without the perturbation compensator, the maximum position error in the actuator 4 is 10 mm and not converged to zero as shown in Fig. 15. And in this case, an inherent chattering is larger than the proposed algorithm. Therefore, the proposed controller is able to provide a superior performance over the conventional sliding mode controller.

4. Performance Evaluation

In the case of the virtual university map, it is impossible for driving in high speed. Therefore, performance evaluation is carried out by using the developed PNU vehicle driving simulator in a wide country road. Figure 16 shows a graphic scene of the road. This road includes many driving conditions such as a lane change, a vehicle velocity change, a J-turn and so on. Drivers are composed of young men, old men, males, females, and so on. And, numbers of the test drivers are 13 persons. The evaluation items

are composed of "movement" of the motion platform and "overall feel" of the vehicle simulator in each driving condition⁽¹²⁾. Figure 17 shows the results of the performance evaluation. In the cases of the J-turn, and the virtual feeling the developed simulator give a realistic feel to test drivers. However, the test drivers complaint that the steering wheel is very sensitive. Also, the sound effects are disrupted by the noise of the hydraulic unit. In the cases of the lane change and the velocity change there exists 20% in "different" as shown in Fig. 17 because the conditions on the simulator don't perfectly match on real conditions. This is caused by the simple virtual images which are composed of mostly driving roads and mountains. It may also be caused by the driving skill of a driver. If the virtual images were additionally generated background images such as tree and building in beside of road, the overall feel will be improved in velocity change condition. In spite of these defects, the test drivers over 70% felt that it had satisfactory realism.

Fig. 16 Virtual driving image

Fig. 17 Results of evaluation for reappearance of motion perception of driving simulator

5. Conclusions

To develop an economical and effective vehicle driving simulator, the subsystems such as the motion platform, the controller, the visual and audio system, and the vehicle dynamic analysis system were developed in personal computer. And then, all subsystems are integrated by LAN and serial communication. For improving motion tracking performance, the sliding mode control algorithm with a perturbation compensator using the observer-based FAN is proposed and carried out experimentally. This control algorithm was designed to solve the chattering problem of a sliding mode control by using the perturbation compensator. Also, the performance of the vehicle driving simulator was evaluated by some

drivers. The results of evaluation were comparatively good.

Acknowledgements

This research was partially supported by the Korea Science and Engineering Foundation (KOSEF, basic research grants: 97-02-00-10-01-5), and by the Ministry of Science and Technology of Korea for financial support in the form of grant (M1-0203-00-0017-02J0000-00910) under the NRL (National Research Laboratory).

References

- (1) Bertollini, G.P. et al., The General Motors Driving Simulator, '94 SAE Paper 940176 (1994).
- (2) Richter, B., Driving Simulator Studies: The Influence of Vehicle Parameters on Safety in Critical Situations, '74 SAE Paper 741105 (1974).
- (3) Haug, E.J. et al., Feasibility Study and Conceptual Design of a National Advanced Driving Simulator: Final Report, NHTST Report DOTHS 807596 (1990).
- (4) Stewart, D., A Platform with Six Degree of Freedom, Proc. of the Institute of Mechanical Engineering, Vol. 180 (1996), pp. 317-386.
- (5) Fichter, E.F., A Stewart Platform-based Manipulator General theory and Practical Construction, Journal of Robotics Research, Vol. 5, No. 2 (1986), pp. 157-182.
- (6) Slotine, J.J., The Robust Control of Robot Manipulators, Int. J. of Robotics Research, Vol. 4, No. 4 (1985), pp. 49-64.
- (7) Hashimoto, H. et al., A Microprocessor-Based Robot Manipulator Control with Sliding Mode, IEEE Trans. Industrial Electronics, Vol. 34, No. 1 (1987), pp. 11-18.
- (8) SGI Computer System, OpenGL Performer Programmer's Guide, (1997).
- (9) Kim, S.S. and Vanderploeg, M.J., A General and Efficient Method for Dynamic Analysis of Mechanical Analysis of Mechanical Systems Using Velocity Transformations, Transaction of the ASME, Vol. 108 (1986), pp. 176-182.
- (10) Kim, K.S. et al., Development of Vehicle Dynamics Program AutoDyn7(I) - Structure and Algorithm, Transaction of KSAE, Vol. 7, No. 3 (1999), pp. 321-330.
- (11) Suetomi, T., Horiguchi, A., Okamoto, Y. and Hata, S., The Driving Simulator with Large Amplitude Motion System, '91 SAE Paper 910113 (1991).
- (12) Nemire, K., Simulation Fidelity of a Virtual Environment Display, HUMAN FACTORS, Vol. 36, No. 1 (1994), pp. 79-93.