Two Performance Measures for Evaluating Human Control Strategy

Jingyan Song[1], Yangsheng Xu[2,3], Michael C. Nechyba[3] and Yeung Yam[2]

[1]Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong
[2]Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong
[3]The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Abstract

In the last few years, modeling dynamic human control strategy (HCS) is becoming an increasingly popular paradigm in a number of different research areas, such as the intelligent vehicle highway system, virtual reality and robotics. Usually, these models are derived empirically, rather than analytically, from real human input-output control data. As such, there is a great need to develop adequate performance criteria for these models, as few guarantees exist about their theoretical performance. It is our goal in this paper to develop several such criteria. In this paper, we first collect driving data from different individuals through a real-time graphic driving simulator. We then model each individual's control strategy through the flexible cascade neural network learning architecture. Next, we develop two performance measures for evaluating the resulting HCS models, one dealing with obstacle avoidance, the other with tight-turning behavior. Finally, we evaluate the relative skill of different HCS models through the proposed performance criteria.

1. Introduction

HCS models, which accurately emulate dynamic human behavior, find application in a number of research areas ranging from robotics to the intelligent vehicle highway system. Because human control strategy (HCS) is a dynamic, nonlinear stochastic process, developing good analytic models of human control strategies, tends to be difficult. Therefore, recent work in modeling HCS has focussed on learning empirical models, through, for example, fuzzy logic [1,2], and neural network techniques [3]. Since these HCS models are empirical, few if any guarantees exist about their theoretical performance. Thus, performance evaluation is an integral aspect of HCS modeling research, without which it is impossible to rank or prefer one HCS controller over another.

Skill or performance can be defined through a number of task-dependent as well as task-independent criteria. Some of these criteria may conflict with one another, and which is most appropriate for a given task depends in part on the specific goals of the overall task. Therefore, rather than examine performance evaluation in the abstract, we focus on one specific HCS namely, the task of human driving.

In this paper, we first record driving data from different individuals through a dynamic driving simulator. For each driver, we then train a HCS model using the flexible cascade neural network learning architecture. Because each of the different drivers exhibits a different style or control strategy, their respective models will likewise differ. It is the goal of this work to define performance criteria by which the driving models' performance can be evaluated and ranked.

In previous work, a stochastic similarity measure, which compares model-generated control trajectories to the original human training data, has been proposed for validating HCS models [4]. While this similarity measure can ensure that the neural network model adequately captures the driving characteristics of the human operator, it does not measure a particular model's skill or performance. In other words, it does not (nor can it) tell us which model is better or worse. In this paper, therefore, we propose two measures for evaluating the performance of HCS models.

For the task of driving, many candidate performance criteria, such as average speed, driving stability, driving safety and fuel efficiency, exist. Rather than select such specific criteria, however, we prefer to decompose the driving task into subtasks, and then define criteria that measure performance for the most meaningful subtasks. For example, we can view the driving task as a combination of the following subtasks: (1) driving along a straight road, (2) turning through a curve in the road and (3) avoiding obstacles. Of these subtasks, avoiding obstacles is perhaps the most useful one, as it can measure important characteristics of the HCS model, including its ability to change speeds quickly while maintaining vehicle safety and stability. Thus, the first performance criterion we develop is based on avoiding obstacles. Our second performance criterion investigates the HCS models' behavior in executing tight turns. Both of these test the behavior of the models outside the range of training data from which the models are learned.

In this paper, we first introduce the dynamic graphic driving simulator that we use to collect human data and from which the HCS models are trained. We then show how we model each individual's human control strategy using the cascade neural network learning architecture. Finally, we develop the obstacle avoidance and tight turning performance criteria. For each of the proposed criteria, we demonstrate their use on HCS models trained from different individuals' data.

2. Experimental setup

For this work, we collect human driving data from a real-time graphic simulator, whose interface is shown in Figure 1 below. In the simulator, the human operator has independent control of the vehicle's steering as well as the brake and gas pedals. The simulated vehicle's dynamics are given by the following second-order nonlinear model:

$$\ddot{\theta} = (l_f P_f \delta + l_f F_{\xi f} - l_r F_{\xi r})/I \tag{1}$$

$$\dot{v}_\xi = (P_f \delta + F_{\xi f} + F_{\xi r})/m - v_\eta \dot{\theta} - (\mathrm{sgn} v_\xi) c_D v_\xi^2 \tag{2}$$

$$\dot{v}_\eta = (P_f + P_r - F_{\xi f}\delta)/m + v_\xi \dot{\theta} - (\mathrm{sgn} v_\eta) c_D v_\eta^2 \tag{3}$$

$$\begin{bmatrix}\dot{x}\\ \dot{y}\end{bmatrix} = \begin{bmatrix}\cos\theta & \sin\theta\\ -\sin\theta & \cos\theta\end{bmatrix}\begin{bmatrix}v_\xi\\ v_\eta\end{bmatrix}, \text{ where,} \quad (4)$$

$$\dot{\theta} = \text{angular velocity of the car} \quad (5)$$

$$v_\xi = \text{lateral velocity of the car}, \quad (6)$$

$$v_\eta = \text{longitudinal velocity of the car}, \quad (7)$$

$$F_{\xi k} = \mu F_{zk}(\tilde{\alpha}_k - (\operatorname{sgn}\delta)\tilde{\alpha}_k^2/3 + \tilde{\alpha}_k^3/27)\times \sqrt{1 - P_k^2/(\mu F_{zk})^2 + P_k^2/c_k^2}, \; k \in \{f, r\} \quad (8)$$

$$\tilde{\alpha}_k = c_k\alpha_k/(\mu F_{zk}), \; k \in \{f, r\}, \quad (9)$$

$$\alpha_f = \text{front tire slip angle} = \delta - (l_f\dot{\theta} + v_\xi)/v_\eta, \quad (10)$$

$$\alpha_r = \text{rear tire slip angle} = (l_r\dot{\theta} - v_\xi)/v_\eta, \quad (11)$$

$$F_{zf} = (mgl_r - (P_f + P_r)h)/(l_f + l_r),$$
$$F_{zf} = (mgl_f + (P_f + P_r)h)/(l_f + l_r), \quad (12)$$

$$\xi = \text{body-relative lateral axis},$$
$$\eta = \text{body-relative longitudinal axis}, \quad (13)$$

$$c_f, c_r = 50000\text{N/rad}, 64000\text{ N/rad}, \quad (14)$$

$$c_D = 0.0005\text{m}^{-1}, \quad (15)$$

$$\mu = \text{coefficient of friction} = 1, \; F_{jk} = \text{frictional forces}, \; j \in \{\xi, z\}, \; k \in \{f, r\} \quad (16)$$

$$P_r = \begin{cases}0, & P_f \geq 0\\ k_b P_f, & P_f < 0, k_b = 0.34\end{cases}, \; m = 1500\text{kg},$$

$$I = 2500\text{kg-m}^2, \; l_f = 1.25\text{m}, \; l_r = 1.5\text{m}, \; h = 0.5\text{m}, \quad (17)$$

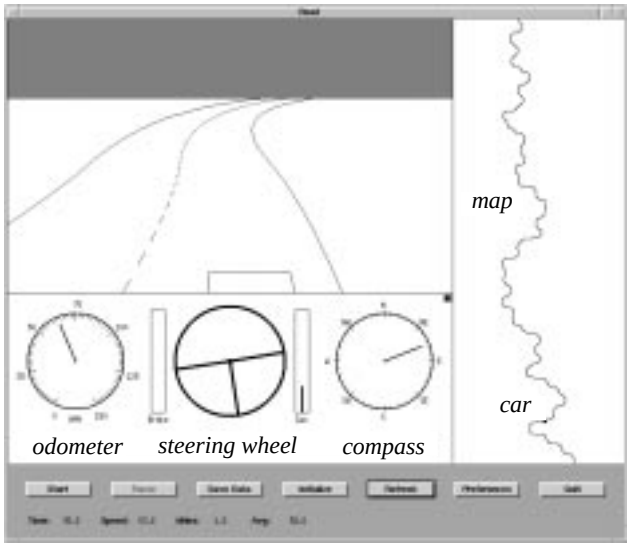

Fig. 1: The driving simulator gives the user a perspective preview of the road ahead. The user has independent controls of the steering, brake, and accelerator (gas).

and the controls are given by,

$$-8000\text{N} \leq P_f \leq 4000\text{N} \quad (18)$$

$$-0.2\text{rad} \leq \delta \leq 0.2\text{rad} \quad (19)$$

where P_f is the longitudinal force on the front tires, and δ is the steering angle.

Note that the separate brake and gas commands for the human are in fact the single P_f variable, where the sign indicates whether the brake or the gas is currently active. Each individual is asked to navigate across several randomly generated roads, which consist of a sequence of (1) straight-line segments, (2) left turns, and (3) right turns. The map in Figure 1, for example, illustrates one randomly generated 20km road for which human driving data was recorded. Each straight-line segment as well as the radius of curvature for each turn range in length between 100m and 200m. Nominally, the road is divided into two lanes, each of which has width $w_l = 10\text{m}$. The human operator's view of the road ahead is limited to 100m. Finally, the entire simulator is run at 50Hz.

3. HCS modeling

In this paper, we choose the flexible cascade neural network architecture with node-decoupled extended Kalman filtering (NDEKF) [5, 6] for modeling the human driving data. We prefer this learning architecture over others for a number of reasons. First, no *a priori* model structure is assumed; the neural network automatically adds hidden units to an initially minimal network as the training requires. Second, hidden unit activation functions are not constrained to be a particular type. Rather, for each new hidden unit, the incremental learning algorithm can select that functional form which maximally reduces the residual error over the training data. Typical alternatives to the standard sigmoidal function are sine, cosine, and the Gaussian function. Finally, it has been shown that node-decoupled extended Kalman filtering, a quadratically convergent alternative to slower gradient descent training algorithms (such as backpropagation or quickprop) fits well within the cascade learning framework and converges to good local minima with less computation [5].

The flexible functional form which cascade learning allows is ideal for abstracting human control strategies, since we know very little about the underlying structure of each individual's internal controller. By making as few *a priori* assumptions as possible in modeling the human driving data, we improve the likelihood that the learning algorithm will converge to a good model of the human control data.

In order for the learning algorithm to properly model each individual's human control strategy, the model must be presented with those state and environmental variables upon which the human operator relies. Thus, the inputs to the cascade neural network should include, (1) current and previous state information $\{v_\xi, v_\eta, \dot{\theta}\}$, (2) previous output (command) information $\{\delta, P_f\}$, and (3) a description of the road visible from the current car position. More precisely, the network inputs are,

$$\begin{aligned}&\{v_\xi(k-n_s), \ldots, v_\xi(k-1), v_\xi(k)\\ &v_\eta(k-n_s), \ldots, v_\eta(k-1), v_\eta(k)\\ &\dot{\theta}(k-n_s), \ldots, \dot{\theta}(k-1), \dot{\theta}(k)\}\end{aligned} \quad (20)$$

$$\{\delta(k-n_c), \ldots, \delta(k-1), \delta(k)$$
$$P_f(k-n_c), \ldots, P_f(k-1), P_f(k)\}, \quad (21)$$

$$\{x(1), x(2), \ldots, x(n_r), y(1), y(2), y(n_r)\}. \quad (22)$$

where n_s is the length of the state histories and n_c is the length of the previous command histories presented to the network as input. For the road description, we partition the visible view of the road ahead into n_r equivalently spaced, body-relative (x, y) coordinates of the road median, and provide that sequence of coordinates as input to the network. Thus, the total number of inputs to the network n_i are,

$$n_i = 3n_s + 2n_c + 2n_r \quad (23)$$

The two outputs of the cascade network are $\{\delta(k+1), P_f(k+1)\}$. For the system as a whole, the cascade neural network can be viewed as a feedback controller, whose two outputs control the driving of the vehicle. Figure 2 illustrates the overall structure of the model-vehicle system.

4. Obstacle avoidance performance criterion

In real driving, obstacles such as rocks and debris can unexpectedly obstruct a vehicle's path and force a driver to react rapidly. Thus, obstacle avoidance is one important performance criterion by which we can gauge a model's performance. In this section we develop and evaluate such criteria for different HCS models.

4.1 Virtual path equivalence

Since our HCS models receive only a description of the road ahead as input from the environment, we reformulate the task of obstacle avoidance as *virtual path following*. Assume that an obstacle appears τ meters ahead of the driver's current position. Furthermore, assume that this obstacle completely obstructs the entire width of the road ($2w$) and extends for d meters along the road. Then, rather than follow the path of the actual road, we wish the HCS model to follow a virtual path as illustrated in Figure 3. This virtual path consists of (1) two arcs with radius of curvature γ, which offset the road median laterally by $2w$, followed by (2)

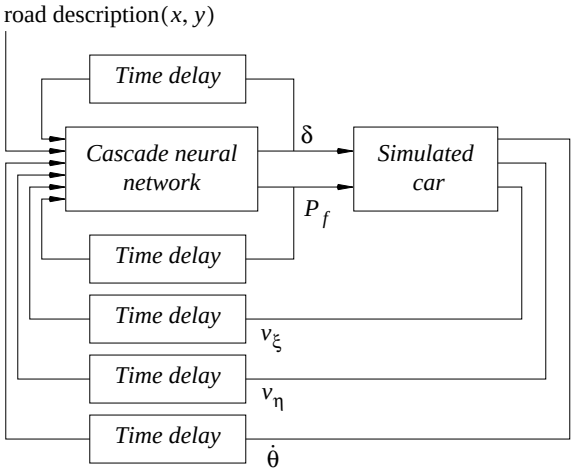

Fig. 2: The overall structure of the simulated system.

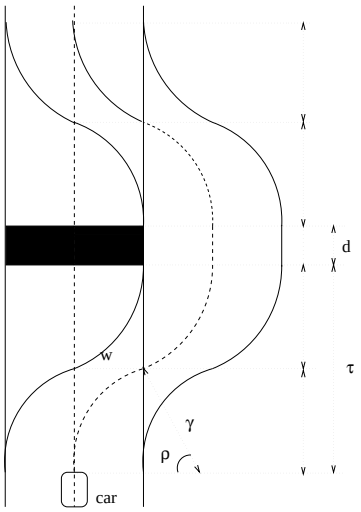

Fig. 3: Virtual path for obstacle avoidance.

a straight-line segment of length d, and (3) another two arcs with radius of curvature γ which return the road median to the original path. By analyzing the geometry of the virtual path, we can calculate the required radius of curvature γ of the virtual path segments in terms of the obstacle width $2w$ and the obstacle distance τ:

$$\gamma^2 = (\tau/2)^2 + (\gamma - w)^2 = \tau^2/2 + \gamma^2 - 2\gamma w + w^2 \quad (24)$$

$$\gamma = [\tau^2/(8w)] + w/2 \quad (25)$$

The corresponding sweep angle ρ for the curves is given by,

$$\rho = \sin^{-1}\left(\frac{\tau/2}{\gamma}\right) = \sin^{-1}\left[\tau/\left(\frac{\tau^2}{4w} + w\right)\right] \quad (26)$$

Consider an obstacle located $\tau = 60\text{m}$ ahead of the driver's current position. For this obstacle distance and $w = 5\text{m}$, γ evaluates to 92.5m. This is less than the minimum radius of curvature (100m) that we allow for the roads over which we collect our human control data. Therefore, the obstacle avoidance task in part tests each HCS model's ability to operate safely outside the range of its training data.

As an example, Figure 4 illustrates one HCS model's response to the virtual path created by an obstacle distance of 60m. Figure 4(a) plots the vehicle's lateral distance from the road median through the virtual path. We observe that on the virtual path, the vehicle deviates sharply from the road median by over 4m. In addition, Figure 4(b) shows that the velocity of the car drops substantially from approximately 35m/sec to a low of about 23m/sec on the virtual path. The model's corresponding steering (δ) and force (P_f) outputs are plotted in Figure 4(c) and (d), respectively.

4.2 Lateral offset estimation

As we observed in Figure 4(a), a driving model may deviate significantly from the center of the road during the obstacle avoidance maneuver. Below, we derive the important relationship between the obstacle detection distance τ and a model's corresponding maximum lateral deviation ψ. First, we take N measurements of ψ for different values of τ, where we denote

the ith measurement as (τ_i, ψ_i). Next, we assume a polynomial relationship of the form,

$$\begin{aligned}\psi_i &= \alpha_p \tau_i^p + \alpha_{p-1}\tau_i^{p-1} + \ldots + \alpha_1 \tau_i + \alpha_0 + e_i \\ &= \Gamma_i^T \alpha + e_i\end{aligned}, \quad (27)$$

where the e_i are additive measurement error. Then, we can write,

$$\begin{aligned}\psi_1 &= \Gamma_1^T \alpha + e_1 \\ \psi_2 &= \Gamma_2^T \alpha + e_2 \\ &\ldots \\ \psi_N &= \Gamma_N^T \alpha + e_N\end{aligned} \quad (28)$$

or, in matrix notation,

$$\Psi = \Gamma\alpha + e \text{, where,} \quad (29)$$

$$\Psi = [\psi_1, \psi_2, \ldots, \psi_N]^T, \quad (30)$$

is the observation vector,

$$\Gamma = [\Gamma_1, \Gamma_2, \ldots, \Gamma_N]^T, \quad (31)$$

is the regression matrix, and $e = [e_1, e_2, \ldots, e_N]$ is the error vector.

Assuming white noise properties for e ($E\{e\} = 0$ and $E\{e_i e_j\} = \sigma_e^2 \delta_{ij}$ for all i, j), we can minimize the least-squares error criterion,

$$V(\hat{\alpha}) = \frac{1}{2}\varepsilon^T\varepsilon = \frac{1}{2}\sum_{k=1}^{N}\varepsilon_k^2 = \frac{1}{2}(\Psi - \Gamma\hat{\alpha})^T(\Psi - \Gamma\hat{\alpha}), \quad (32)$$

with the optimal, unbiased estimate $\bar{\alpha}$,

$$\bar{\alpha} = (\Gamma^T\Gamma)^{-1}\Gamma^T\Psi, \quad (33)$$

assuming that $(\Gamma^T\Gamma)$ is invertible.

For example, consider the HCS model from Figure 4. We plot its measured ψ values for τ ranging from 20 to 100 meters in Figure 5. Superimposed on top of the measured data is the estimated fifth-order relationship ($p = 5$) between ψ and τ. We observe that the polynomial model fits the data closely and appears sufficient to express the relationship between ψ and τ.

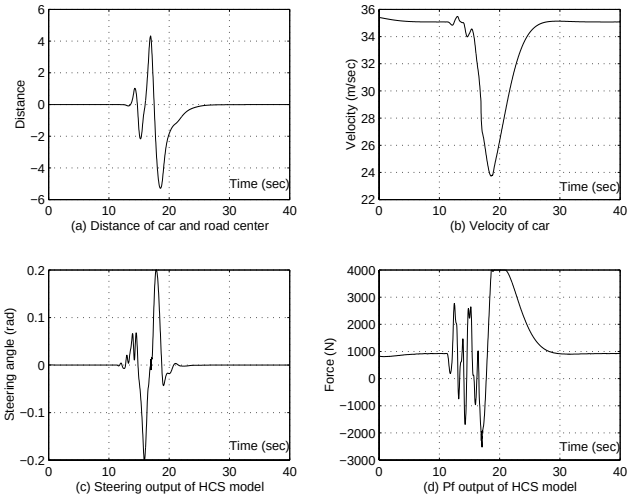

Fig. 4: Example of obstacle avoidance by a HCS model.

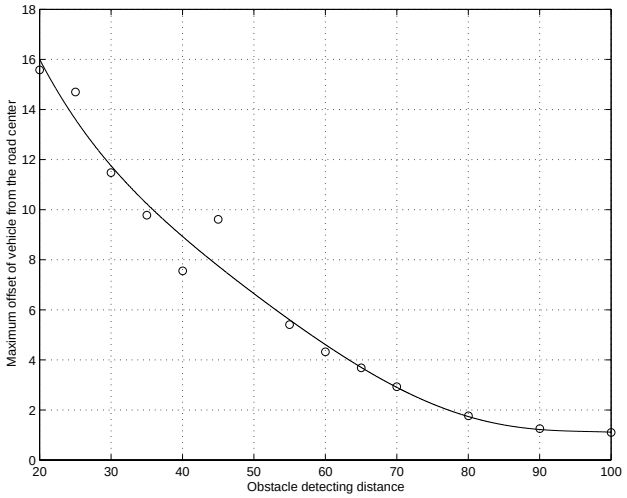

Fig. 5: Maximum lateral offset as a function of obstacle detection distance.

4.3 Obstacle avoidance threshold

We note from Figure 5 that as the obstacle detection distance decreases, the maximum lateral offset increases. Thus, for a given model and initial velocity $v_{initial}$, there exists a value τ_{min} below which the maximum offset error will exceed the lane width w_l. We define the driving control for obstacle distances above τ_{min} to be stable; likewise, we define the driving control to be unstable for obstacle distances below τ_{min}.

Now, define an obstacle avoidance performance criterion β,

$$\beta = \frac{\tau_{min}}{v_{initial}}, \quad (34)$$

where $v_{initial}$ is the velocity of the vehicle when the obstacle is first detected. The β criterion measures to what extent a given HCS model can avoid an obstacle while still controlling the vehicle in a stable manner. The normalization by $v_{initial}$ is required, because slower speeds increase the amount of time a driver has to react and therefore avoiding obstacles becomes that much easier.

Below, we calculate the β performance criterion for three HCS models, trained on real driving data from Tom, Dick, and Harry, respectively. Figure 6 plots ψ as a fifth-order function of τ for the three different models. From Figure 6, it is easy to approximate τ_{min} for each HCS model; thus, the corresponding β performance criterion for each model is,

$$\beta_{Tom} = 45/35 = 1.3 \quad (35)$$

$$\beta_{Dick} = 35/35 = 1.0 \quad (36)$$

$$\beta_{Harry} = 18/35 = 0.51 \quad (37)$$

Thus, as an obstacle avoider, Harry's model clearly outperforms Tom's and Dick's models, since β_{Harry} is the lowest performance measure for the three models.

4.4 Obstacle avoidance velocity loss

The performance criterion β measures the stability of a particular HCS model in avoiding an obstacle. It does not, however,

directly measure how skillfully the model avoids the obstacle. Consider, for example, Figure 4(b). During the obstacle avoidance maneuver, the velocity of the vehicle drops sharply so that the model can adequately deal with the tight maneuvers required. Below, we define a performance criterion J which measures the distance lost as a result of this velocity drop:

$$J = \int_{t_0}^{t_f} |v_{initial} - v_{virtual}| dt , \quad (38)$$

where $v_{initial}$ is the velocity before obstacle detection, and $v_{virtual}$ is the time-dependent velocity during the obstacle avoidance maneuver.

Consider once again the three HCS models for Tom, Dick and Harry. Each model can successfully avoid the obstacle when τ ranges from 50 to 100 meters. Figure 7 plots J for this range of τ. Once again we observe that Harry's model performs best when evaluated with the J performance criterion, since its distance loss is smaller for each τ then either Tom's or Dick's model.

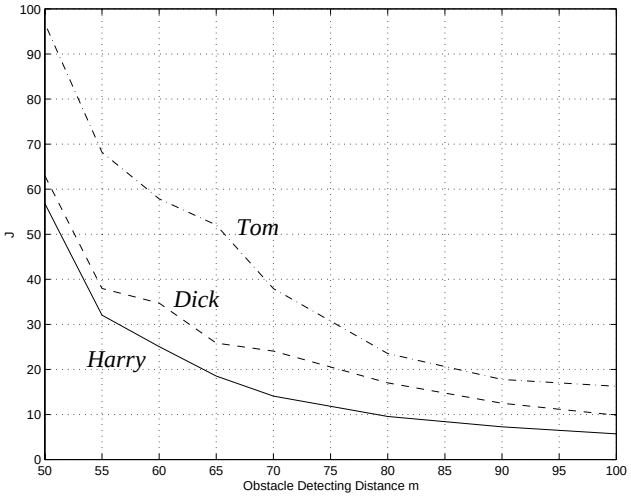

Fig. 7: Distance loss as a function of obstacle detection distance.

5. Tight turning

Here we analyze performance as a function of how well a particular HCS model is able to navigate tight turns. First, we define a special road connection consisting of two straight-line segments connected directly (without a transition arc segment) at an angle ζ. For small values of ζ, each HCS model will be able to successfully drive through the tight turn; for larger values of ζ, however, some models will fail to execute the turn properly by temporarily running off the road or losing complete sight of the road.

Figure 8 illustrates for example, how Harry's model transitions through a tight turn for $\zeta = 5\pi/36$rad. Figure 8(a) plots the two straight-line segments connected at an angle ζ. The solid line describes the road median, while the dashed line describes the actual trajectory executed by Harry's HCS model. The length of the initial straight-line segment is chosen to be long enough (150m) to eliminate transients by allowing the model to settle into a stable state. This is equivalent to allowing the vehicle to drive on a straight road for a long period of time before the tight turn appears in the road. Figure 8(b) plots the lateral offset from the road median during the tight-turn maneuver. Here, Harry's model maximally deviates about 8m from the road center. Both before and after the turn, the lateral offset converges to zero. Figure 8(c) plots the commanded steering angle for Harry's HCS model, and Figure 8(d) plots the corresponding change in velocity.

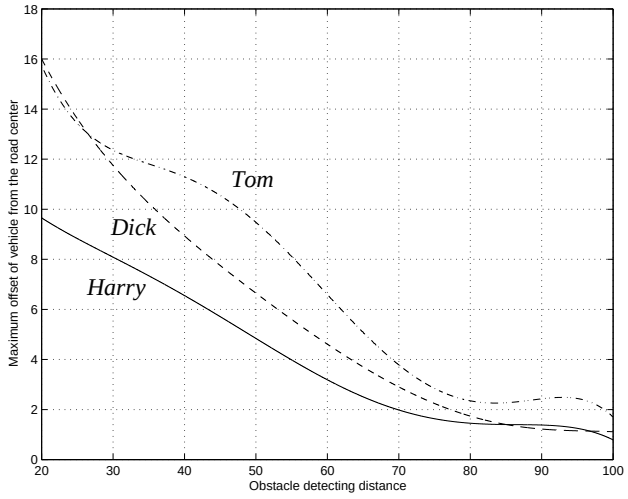

Fig. 6: Max. lateral offset for Dick's, Tom's and Harry's model.

Now, define the maximum lateral offset error corresponding to a tight turn with angle ζ to be ρ. We want to determine a functional relationship between ρ and ζ for a given HCS model. First, we take N measurements of ρ for different values of ζ, where we denote the ith measurement as (ζ_i, ρ_i). Then, similar to Section 4.2, we assume a polynomial relationship between ρ and ζ such that,

$$\rho_i = \alpha_p \zeta_i^p + \alpha_{p-1} \zeta_i^{p-1} + \ldots + \alpha_1 \zeta_i + \alpha_0 + e_i \quad (39)$$

The least-squares estimate of the model $\hat{\alpha}$ is given by,

$$\hat{\alpha} = (\hat{\zeta}^T \hat{\zeta})^{-1} \hat{\zeta} \hat{\rho} \text{ , where,} \quad (40)$$

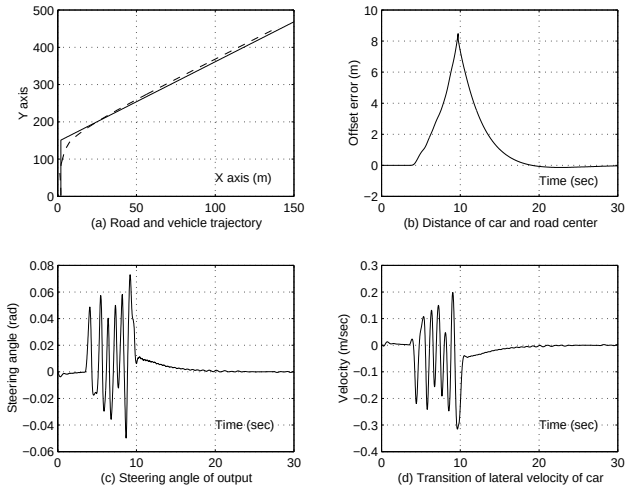

Fig. 8: Example model driving behavior through a tight turn.

$$\hat{\rho} = [\rho_1, \rho_2, \ldots, \rho_N]^T \tag{41}$$

$$\hat{\zeta} = \begin{bmatrix} \zeta_1^p & \zeta_1^{p-1} & \ldots & \zeta_1 & 1 \\ \zeta_2^p & \zeta_2^{p-1} & \ldots & \zeta_2 & 1 \\ \vdots & \vdots & & \vdots & \vdots \\ \zeta_N^p & \zeta_N^{p-1} & \ldots & \zeta_N & 1 \end{bmatrix} \tag{42}$$

$$\hat{\alpha} = [\alpha_p, \alpha_{p-1}, \ldots, \alpha_0]^T \tag{43}$$

For ζ ranging from $-4\pi/9$rad to $4\pi/9$rad and assuming a fifth-order model ($p = 5$), we arrive at the following estimate for Harry's model,

$$\rho = 2.78\zeta^5 - 0.584\zeta^4 - 0.599\zeta^3 - 4.286\zeta^2 + 11.68\zeta - 0.330 \tag{44}$$

and the following estimate for Dick's model,

$$\rho = -1.734\zeta^5 + 1.076\zeta^4 + 2.258\zeta^3 - 0.243\zeta^2 + 21.29\zeta - 0.679 \tag{45}$$

Equations (44) and (45) are plotted in Figure 9. For a given road width, we can determine the values of ζ for which each model stays on the road. For example, assume a road width of 20m. Then, the maximum allowable lateral offset is ± 10m. From Figure 10 below, where the boundaries are explicitly drawn, we observe that Harry's model can execute tight turns from -0.65rad to 1.05rad, while Dick's model can only execute tight turns from -0.45rad to 0.48rad. Thus, Harry's model generates stable driving for wider range of conditions than does Dick's model.

We note that as a first-order approximation, we can define the tight-turning performance criterion J to be,

$$J = \alpha_1 \tag{46}$$

where α_1 is the linear coefficient in the fifth order model of equations (44) and (45), and smaller values of J indicate better performance. In that case,

$$J_{Harry} = 11.68 \text{ and } J_{Dick} = 21.29 \tag{47}$$

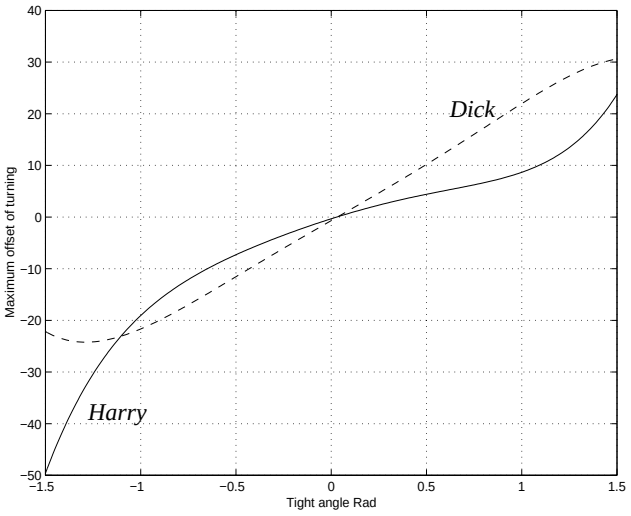

Fig. 9: Maximum lateral offset in tight turns for Dick's and Harry's model.

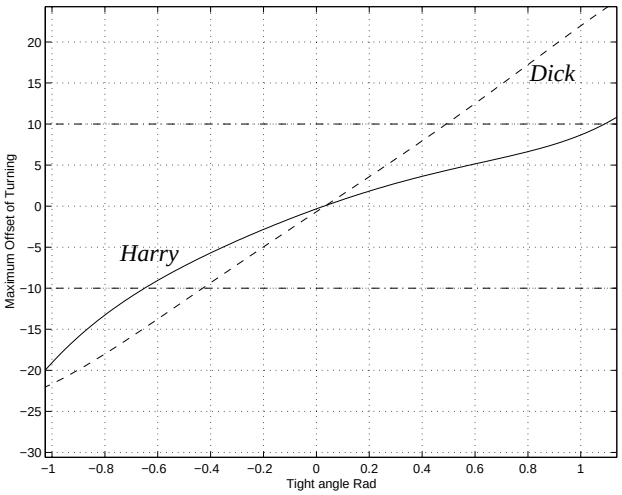

Fig. 10: Harry's model stays on the road for a greater range of tight turns.

6. Conclusion

Modeling human control strategy analytically is difficult at best. Therefore, an increasing number of researchers have resorted to empirical modeling of human control strategy as a viable alternative. This in turn requires that performance criteria be developed, since few if any theoretical guarantees exist for these models. In this paper, we develop several such criteria for the task of human driving, including obstacle avoidance and tight-turning performance criteria. We model human driving using the cascade neural network architecture, and evaluate the performance of driving models derived from different individuals using the developed performance criteria.

Acknowledgments

This work is supported in part by RGC Grant No. CUHK519/95E.

References

[1] M. Sugeno and T. Yasukawa, "A Fuzzy-Logic-Based Approach to Qualitative Modeling," *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 1, 1993.

[2] U. Kramer, "On the Application of Fuzzy Sets to the Analysis of the System-Driver-Vehicle-Environment," *Automatica*, vol. 21, no. 1, pp. 101-7, 1985.

[3] M. C. Nechyba and Y. Xu, "Human Control Strategy: Abstraction, Verification and Replication," to appear in *IEEE Control Systems Magazine*, October 1997.

[4] M. C. Nechyba and Y. Xu, "Stochastic Similarity for Validating Human Control Strategy Models," *Proc. IEEE Conf. on Robotics and Automation*, vol. 1, pp. 278-83, 1997.

[5] M. C. Nechyba and Y. Xu, "Cascade Neural Networks with Node-Decoupled Extended Kalman Filtering," *Proc. IEEE Int. Symp. on Computational Intelligence in Robotics and Automation*, vol. 1, pp. 214-9, 1997.

[6] S. E. Fahlman, L. D. Baker and J. A. Boyan, "The Cascade 2 Learning Architecture," Technical Report, CMU-CS-TR-96-184, Carnegie Mellon University, 1996.