The Society for engineering in agricultural, food, and biological systems

This is not a peer-reviewed paper.

Paper Number: 01-1162
An ASAE Meeting Presentation

Virtual Design Tools: A Technique for Performing Qualitative Human-in-the-Loop System Design

William R. Norris[1], Ramavarapu Sreenivas[2], Qin Zhang[1]

[1]Department of Agricultural Engineering, [2]Department of General Engineering,
University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA,
wnorris / rsree / qinzhang@uiuc.edu

Jose C. Lopez–Dominguez

Hydraulics and Fabrications Business Unit, Caterpillar, Inc. Joliet, Illinois 60434-0504, USA,
Lopez-Dominguez_Jose@cat.com

Written for presentation at the
2001 ASAE Annual International Meeting
Sponsored by ASAE
Sacramento Convention Center
Sacramento, California, USA
July 30-August 1, 2001

Abstract. *One of the major deterrents in designing control systems for interactive use with a human operator is the lack of a systematic procedure for modeling and incorporating human behavior directly in the design process. In many cases, direct human operator feedback was difficult for qualitative design without a costly, high–fidelity physical prototype. The objective of this paper is to present an overview of a modular framework, known as virtual design tools, developed to design the electrohydraulic steering system of an articulated wheel loader. The aim was to adapt the electromechanical interface between the operator's command signal and the fixed control and dynamic systems. The adaptation was performed with the intent of modifying the system performance so that it would be more acceptable to a human operator without altering the control system or vehicular components.*

In order to achieve this objective, there were several significant accomplishments. The first was the development of a nontrivial model that provided adequate real time simulations for human-in-the-loop systems using a standard control system design package. A virtual operator has been realized that will navigate the vehicle along a predetermined trajectory mimicking consistent human performance characteristics while using a human-like interpretation of position and orientation errors. Several different structures have been studied for use as an adaptive interface between the operator and the vehicle and a proof of concept for virtual design tools has been attained. The successful employment of virtual design tools to the wheel loader steering design application and an overview of the results are discussed.

Keywords. Virtual Environments, Human-in-the-loop Systems, Qualitative Design, Electro-mechanical Systems, Steer-by-wire systems, Autonomous Systems.

Introduction

The purpose of this paper is to present a design framework, known as virtual design tools, which uses virtual environments to incorporate human-preferred performance characteristics in the qualitative design process. Closing the system design loop, virtual design tools are composed of several user-designed modular components that can be removed or replaced depending on the design objectives (Norris, 2001a).

Current virtual prototyping techniques allow for the evaluation of a new product before the manufacture of a physical model. Design flaws can be determined ahead of time before the expense of a costly prototype (Purschke et al., 1998). One of the limitations of contemporary virtual prototyping is that there is a trade-off between the complexity of the dynamics and the simulation detail. Detailed simulations are only achieved with special-purpose representations that do not suit other applications (Willhelm and Nagalla, 1996). In many industrial applications, the dynamics are ignored and the simulations are limited to a three-dimensional visualization of static objects. Despite the limitations, virtual environments have proven to be an excellent technique for immersing the operator in a realistic, controlled environment (Connacher et al., 1996).

Based on current technology, several obstacles have to be overcome before virtual environments can be implemented as a viable system rapid prototyping and design tool. The first obstacle is the need for real-time simulations. The dynamic control system, the machine controls that interface with the operator and the graphics must operate as a package providing a video-like frame rate of thirty frames per second. As a design tool, the scenarios should be realistic and the dynamics remain nontrivial.

Another challenge is the need for a real time human operator performance model (HOPM), or virtual operator, that is trainable and consistently models human performance. The HOPM would serve several purposes including: as an autonomous algorithm, a method for evaluating the efficacy of the design process, and as a tool for developing adaptive algorithms for capturing the performance of the human operator.

The other requisite is an appropriate metric and adaptive algorithm that improves the machine performance. The technique would be independent of the mechanical structure and would capture the human operator's preferred performance characteristics.

The virtual design tool framework was developed to overcome these obstacles and uses virtual environments to their full capacity. Virtual environments provide a safe medium for evaluating design prototypes. Any scenario can be created where the human operator's effectiveness may be evaluated or improved in a controlled environment.

With the addition of a modifiable interface between the operator's control response and the vehicle dynamics, operator-vehicle synergy will be qualitatively improved. The modifiable design will provide an interface for incorporating safety boundaries in the control system design based on environmental conditions. Among the possibilities are designs for rollover protection, DUI detection, obstacle avoidance, and traction control.

The particular application addressed in this paper is a steer-by-wire design problem implemented as an electrohydraulic steering system on an articulated wheel loader. The approach and a description of the design components are provided in the following sections.

The Virtual Design Tool Topology

The generalized virtual design tool topology, as illustrated in Figure 1, was implemented in the design process for an articulated wheel loader steering system. There were three phases of operation including virtual operator training (VOT), virtual designer development (VDD), and the human operator performance based design (HOPBD). The system components were both modifiable and replaceable throughout these phases. For example, the HOPBD phase involved replacing the virtual operator with a human operator in the virtual environment.

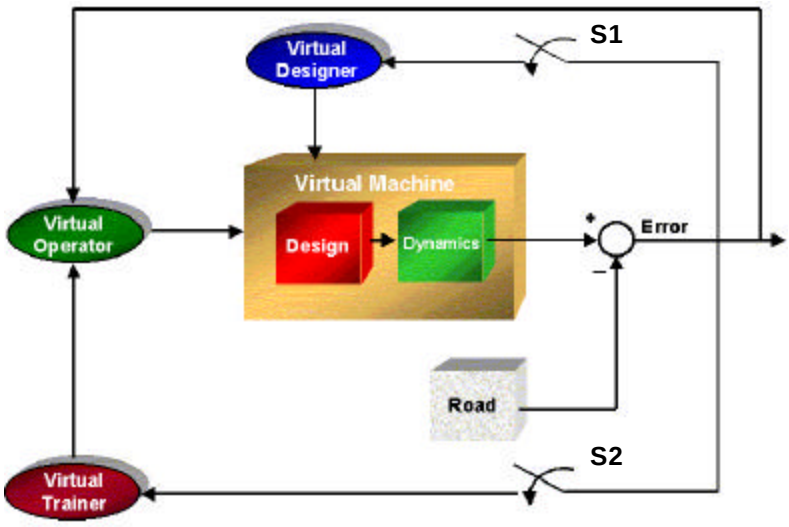

Figure 1: System Topology for Virtual Design Tools.

The specific components as applied to the wheel loader steering problem and their corresponding generalized virtual design tool topology elements are provided below.

1.) The Road was the trajectory or disturbance tracked by the system. The error based on the vehicle's position and orientation relative to the trajectory was transmitted to other components depending on the design phase.

2.) The Virtual Machine consisted of the model's dynamic system that interfaced with the human operator, as well as the adaptable qualitative design component.

3.) The Virtual Operator consisted of the Human Operator Performance Model (HOPM). The HOPM had three components:
   a.) The Virtual Trajectory Planner generated the trajectory or disturbance for the Road.
   b.) The Virtual Interpreter provided error data to the human decision making model (HDMM) and the Virtual Designer.
   c.) The Human Decision-Making Model (HDMM) provided the corresponding steering correction based on the interpreted error.

4.) The Virtual Trainer consisted of various techniques for training the HDMM. For this study, the training was performed manually.

5.) The Virtual Designer consisted of various adaptive techniques for optimizing the system design. For this study, the virtual designer was a neural network that adjusted the parameters of the gain between the operator's signal and the vehicular system in order to perform the qualitative design.

During the VOT phase, switch 1 (S1) was open and Switch 2 (S2) was closed. The virtual designer was out of the information loop and the error between the desired and the actual states were passed to the virtual trainer, which optimized the HDMM. In this mode, the overall system and the design were fixed. The intent was to develop a human performance model or an autonomous control that interpreted feedback from the system and then controlled the system to meet the prescribed objectives.

During the VDD phase, S1 was closed and S2 was open, the virtual trainer was removed from the information loop. The virtual operator was fixed and the designer modified the design using the error between the desired criteria and the actual state of the system. The overall system was not modified. The intent was to optimize the design algorithm to adapt to the given consistent virtual operator based on a set of prescribed performance criteria.

In the HOPBD phase, S1 was closed and S2 was open. The virtual trainer was removed from the information loop a human operator replaced the virtual operator. The designer modified the design using the error between the desired criteria and the actual state of the system. The overall system was not modified. The intent was to optimize the design to the human operator based on a set of prescribed performance criteria developed in the VDD phase.

The Virtual Machine

Vehicle Model

The developmental objective for the wheel loader model was to obtain appropriate detail to provide real-time virtual simulation performance, yet maintain enough complexity to be useful in the design process (Norris, 2001a). A typical front-end wheel loader is presented in Figure 2. The standard wheel loader is a four-wheel, single bucket, articulated vehicle. An articulation joint (AJ) connects the two separate parts of the vehicle. The front part of the vehicle will be referred to as the Non-Engine End Frame (NEEF). The rear part of the vehicle, which contains the engine, will be referred to as the Engine End Frame (EEF). Two hydraulic cylinders adjust the orientation of the NEFF and the EEF effectively steering the vehicle.

The wheel loader model consisted of a dynamic model (Pauling and Larson, 1988), Pacejka tire models (Bakker et al., 1987) and an electrohydraulic steering system (Norris et al., 2001f). While balancing the requirement of satisfying real time constraints with a nontrivial model and current technology limitations, the vehicle model was designed to exhibit the following characteristics:

1.) the model was two-dimensional.

2.) the vehicle moved at low speed with the corresponding neglect of lateral slip.

3.) the vehicle was unloaded.

4.) kinematically, the vehicle was modeled as two rigid bodies with a revolute joint.

5.) Pacejka tire models were used for the cornering forces and moments.

6.) the effects of dry friction and rolling resistance were incorporated.

7.) the hydraulic system model consisted of two hydraulic cylinders and an electrohydraulic spool model.

8.) the steering system was assumed symmetric due to the cylinder model and was a velocity based steer-by-wire system. The rate of flow of the hydraulic fluid into the pistons was controlled.

Figure 2: A typical front-end wheel loader.

A top view of the wheel loader model in the absolute coordinate system is presented in Figure 3. From Newton's Laws of motion the sum of forces and moments were derived to represent the linear accelerations of the vehicle in the global x and y directions. A Newtonian approach was selected in order to incorporate wheel slip into the model.

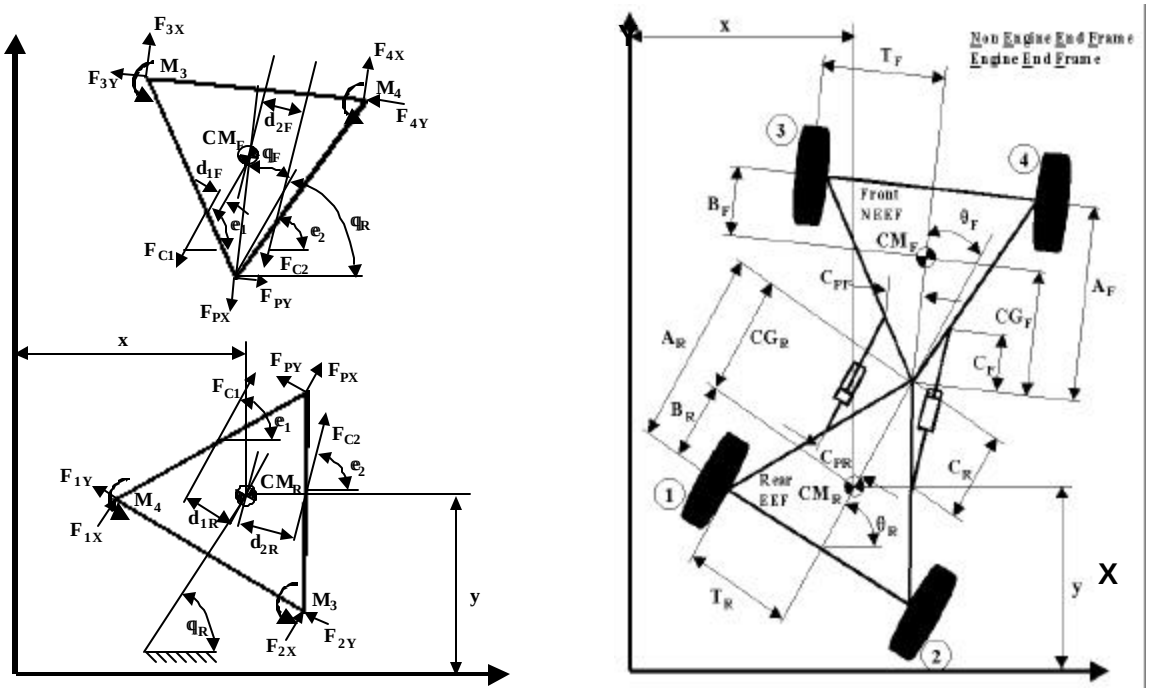

Figure 3: External Force and Moment Free Body Diagrams of the EEF and the NEEF and a top view of the wheel loader in the Absolute Coordinate System.

Applying Newton's Laws of motion to the EEF as demonstrated in Figure 3:

$$m_R \ddot{x} = (F_{1X} + F_{2X} + F_{PX})\cos\theta_R - (F_{1Y} + F_{2Y} + F_{PY})\sin\theta_R + F_{C1}\cos\varepsilon_1 + F_{C2}\cos\varepsilon_2 \quad (1)$$

$$m_R \ddot{y} = (F_{1Y} + F_{2Y} + F_{PY})\cos\theta_R + (F_{1X} + F_{2X} + F_{PX})\sin\theta_R + F_{C1}\sin\varepsilon_1 + F_{C2}\sin\varepsilon_2 \quad (2)$$

$$I_{CMR}\ddot{\theta}_R = M_1 + M_2 + T_R(F_{2X} - F_{1X}) + CG_R F_{PY} - B_R(F_{1Y} + F_{2Y}) + d_{2R}F_{C2} - d_{1R}F_{C1} \quad (3)$$

Applying Newton's Laws of Motion to the NEEF:

$$m_F\ddot{x}_F = (F_{3X} + F_{4X})\cos(\theta_R + \theta_F) - F_{PX}\cos\theta_R - (F_{3Y} + F_{4Y})\sin(\theta_R + \theta_F) + F_{PY}\sin\theta_R - F_{C1}\cos\varepsilon_1 - F_{C2}\cos\varepsilon_2 \quad (4)$$

$$m_F\ddot{y}_F = (F_{3Y} + F_{4Y})\cos(\theta_R + \theta_F) - F_{PY}\cos\theta_R + (F_{3X} + F_{4X})\sin(\theta_R + \theta_F) - F_{PX}\sin\theta_R - F_{C1}\sin\varepsilon_1 - F_{C2}\sin\varepsilon_2 \quad (5)$$

$$I_{CMF}(\ddot{\theta}_R + \ddot{\theta}_F) = M_3 + M_4 + T_F(F_{4X} - F_{3X}) + CG_F F_{PY}\cos\theta_F - CG_F F_{PX}\sin\theta_F + B_F(F_{3Y} + F_{4Y}) + d_{1F}F_{C1} - d_{2F}F_{C2} \quad (6)$$

The constraint equations arising at the pin (articulation joint) were expressed as:

$$x_F = x + CG_R\cos\theta_R + CG_F\cos(\theta_R + \theta_F) \quad (7)$$

$$y_F = y + CG_R\sin\theta_R + CG_F\sin(\theta_R + \theta_F) \quad (8)$$

The electrohydraulic steering system of an articulated vehicle is different from one associated with a fixed-frame vehicle. The wheel loader steers using hydraulic cylinders, where the front and rear portions of the vehicle are rotated around an articulation joint, and the articulation angle provides the steering angle for the vehicle. The vehicle's articulation rate is controlled by the supplying flow rate to the steering cylinders. The steering signal, from a joystick, is translated to a positional control of the spool controlling the flow rate of hydraulic fluid.

From the Bernoulli equation and assuming there was no change in altitude, conservation of flow rate, and one-dimensional flow, the theoretical flow rate was written as the nonlinear orifice equation. Equation 9 uses the nonlinear orifice equation and relates the current from the operator's joystick position to the flow rate into the pistons of the vehicle's electrohydraulic system.

$$Q_{IN} = Q_M = Ki\sqrt{P_S - P_M} \quad (9)$$

P_S and P_M are the pressures of the supply and the pressure across the servomotor, respectively. Q_M is the flow rate to the hydraulic cylinders. The constant K consisted of the constants from the orifice equation, the spool gain based in the metering characteristics, the gain of the solenoid drive and the relationship between the spool area and the current i.

A control volume approach was used to derive the pressure values used to determine the cylinder forces. The respective control volumes were interchangeable depending on the turn direction. The governing piston motion equations were expressed as:

$$\dot{P}_1 = \left(\frac{B_M}{VOL_1 + VOL_{LINE}} \right) \left(Q_{IN} + \dot{x}_{C1} A_2 - \dot{x}_{C2} A_1 \right) \tag{10}$$

$$\dot{P}_2 = \left(\frac{B_M}{VOL_2} \right) \left(Q_{OUT} - \dot{x}_{C1} A_1 + \dot{x}_{C2} A_2 \right) \tag{11}$$

$$Q_{OUT} = C_d A_{OUT} \sqrt{\frac{2(P_2 - P_{TANK})}{\rho}} \tag{13}$$

A_1 is the piston head end area, and A_2 is the piston rod end area. Depending on the articulation direction, $P_{(1,2)}$ is the pressure developed in control volume (1,2). VOL_{LINE} is the volume of hydraulic fluid in the lines. $VOL_{(1,2)}$ is the volume of hydraulic fluid in control volume (1,2). B_M is the bulk modulus of the hydraulic fluid.$x_{(C1,C2)}$ is the position of cylinder (1,2). Q_{OUT} is the flow rate into the hydraulic tank, P_{TANK} is the pressure in the tank and A_{OUT} is the orifice area for the flow out of the system into the tank.

The tires were modeled using the Pacejka tire model, also known as the Magic Tire Formula. (Bakker, Nyborg, and Pacejka, 1987). The formula can be expressed in the following form:

$$Y = S_Y + D \sin\left[C \tan^{-1}\left\{ B(X - S_X)(1 - E) + E \tan^{-1} B(X - S_X) \right\} \right] \tag{14}$$

The six constants in the formula S_X, S_Y, B, C, D, and E are known as the Magic Formula (MF) constants. By varying these constants, a large number of functions with common and uncommon curve shapes can be generated representing the tire characteristics for the cornering forces and self-aligning torques.

Real time simulations of the dynamic model were performed by explicitizing the equations dealing with the vehicle dynamics and the tire model using Mathematica. The simulation model was programmed in C and implemented in user-code block form within MatrixX System Build. The electrohydraulic actuating system was implemented in block diagram form within MatrixX. The virtual simulation code was generated with a template (Norris et al., 2001b; Norris, 2001a) using the Autocode feature within MatrixX. It took 15 s to perform vehicle simulations for a 90 s operation with a 50 Hz sampling rate on an HP C180 workstation.

The Design

The design component in the virtual machine is the interface between the virtual designer, virtual operator and the virtual machine. It consists of the electromechanical interface between the electrical steering command signal input by the operator and the hydraulic spool displacement of the electrohydraulic steering control valve as illustrated in Figure 4. The intent of using the interface was, while maintaining a consistent dynamic system, that the vehicle performance could be optimized for a particular driver. Extending the concept further, the relationship between the steering command and the spool displacement could be optimized for a given operator based on environmental

conditions such as velocity, terrain conditions, and load. The electromechanical interface will be referred to as the *virtual modulation curve (VMC).* As a multi-dimensional mapping, the interface will be referred to as a *virtual modulation surface (VMS).*

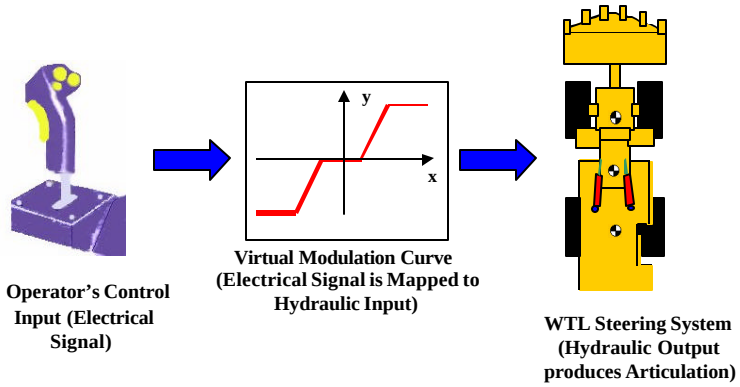

Figure 4: The Virtual Modulation Curve as the adaptable design.

The Virtual Operator

The virtual operator (or HOPM) was implemented into the vehicle-operator system as demonstrated in Figure 5. The virtual operator provided the signal input into the electro-hydraulic system, which provided the forces on the EEF and NEEF of the vehicle dynamic model. The objective was to perform real time adaptive steering design while retaining the dynamic behavior of the electrohydraulic steering control system. The other inputs into the wheel loader model were of a constant gain and converted to respective force magnitudes used in the dynamic vehicle model.

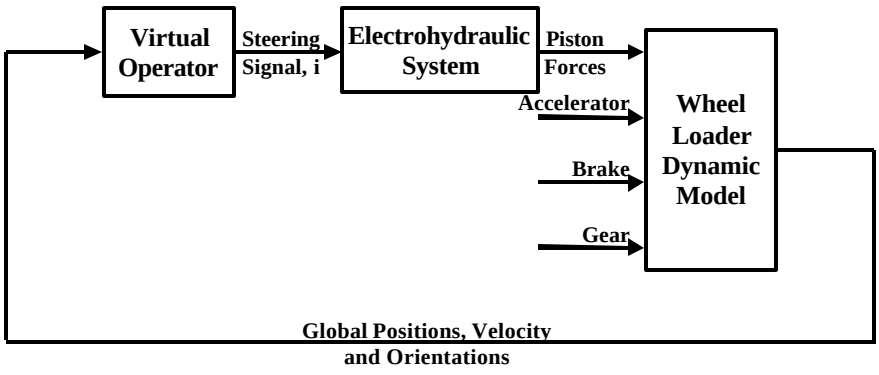

Figure 5: The virtual operator in the operator - vehicle system.

The virtual operator was formulated in three components, a trajectory planner, an error interpreter, and a human decision making model (HDMM), as shown in Figure 6, to adequately model consistent human performance characteristics. .

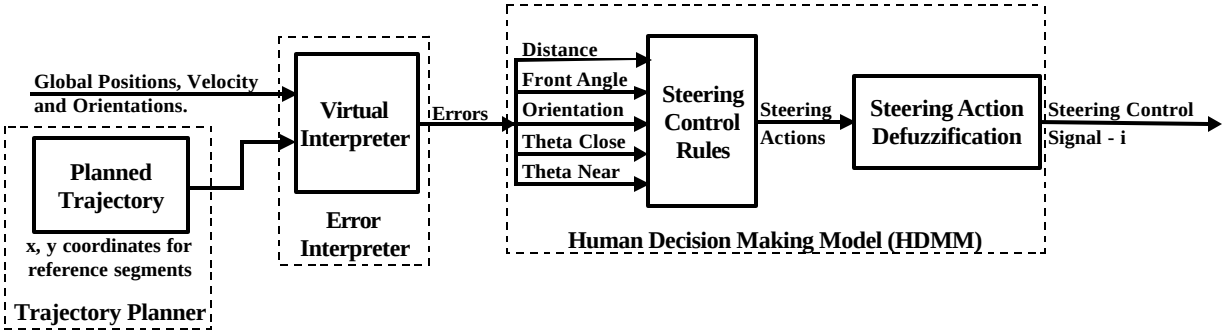

Figure 6: The three components of the virtual operator: Virtual Trajectory Planner, Virtual Interpreter, and HDMM.

The Trajectory Planner and Road System

The trajectory planner provided a path for the vehicle consisting of a series of equally segmented points as shown in Figures 7. The road system was generated by the trajectory planner and based on two options (Norris, 2001a; Norris et al., 2001c). The first option required a user-generated trajectory that could be input directly into the planner. The other option generated a trajectory based on several user inputs including the distance between the points, the radius of curvature, the arc radius and the direction. Figure 7 demonstrates the variables used in generating the road system by the trajectory planner.

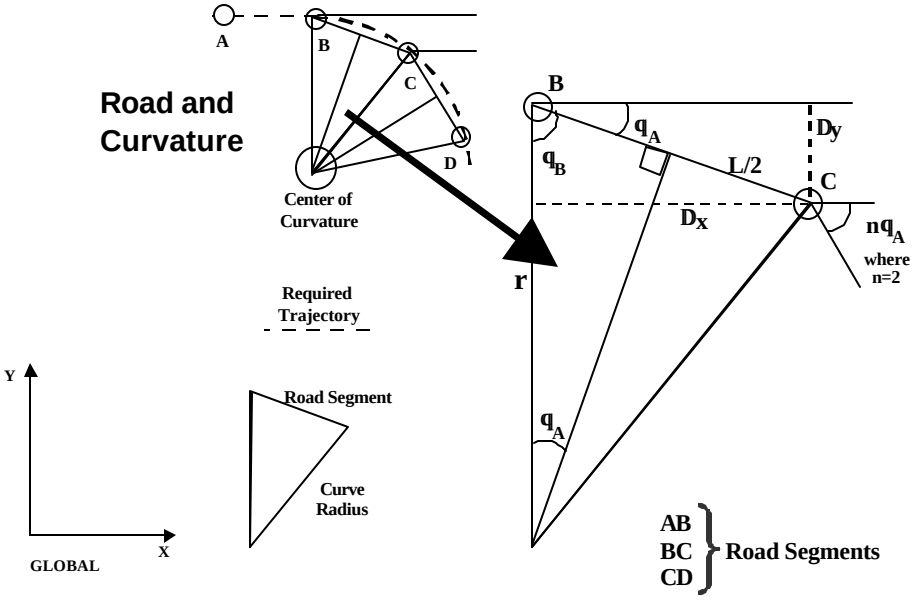

Figure 7: Variables and techniques used by the trajectory planner to generate the road system.

Equations 1 through 6 were used to generate the required trajectory.

$$\theta_A = a\sin\left(\frac{L}{2r}\right) \quad (15)$$

$$N = \frac{\theta_{ARC}}{\theta_A} \quad (16)$$

$$\Delta y(n) = L\sin\left(n\theta_A\right) \quad (17)$$

$$\Delta x(n) = L\cos\left(n\theta_A\right) \quad (18)$$

$$y(n) = y(n-1) \pm \Delta y(n) \quad (19)$$

$$x(n) = x(n-1) \pm \Delta x(n) \quad (20)$$

Where:

θ_A = Angle between the roadway and the angle of the segment along the road curvature.

θ_{ARC} = User provided required angle of curvature.

r = User provided radius of curvature for the trajectory.

L = User provided length of road segment.

N = Number of road segments required to obtain the required angle of curvature.

n = Current segment number.

The following variables were generated based on the criteria that n <= N.

x(n) / x(n-1) = x global position at the end of the latest / previously generated segment.

y(n) / y(n-1) = y global position at the end of the latest / previously generated segment.

$\pm\Delta x(n)$ = Change in the previous x position for the latest segment - the sign was dependent on the global quadrant.

$\pm\Delta y(n)$ = Change in the previous y position for the latest segment. - the sign was dependent on the global quadrant.

Error Interpreter

The error interpreter generated five errors based on the differences between the vehicle position, orientation and the generated trajectory from the planner. The errors were classified as either an immediate check or as a near check as shown in Figure 8. The near check provided the look ahead aspect of the HOPM. The errors were transferred to the HDMM and used to determine a corresponding control action. The error calculations are illustrated in Figure 9. Five errors were used as inputs to the HDMM. The errors were used due to the vehicle's four degrees of freedom and consideration of all the vehicle's possible relative orientations (Norris, 2001a; Norris et al., 2001c).

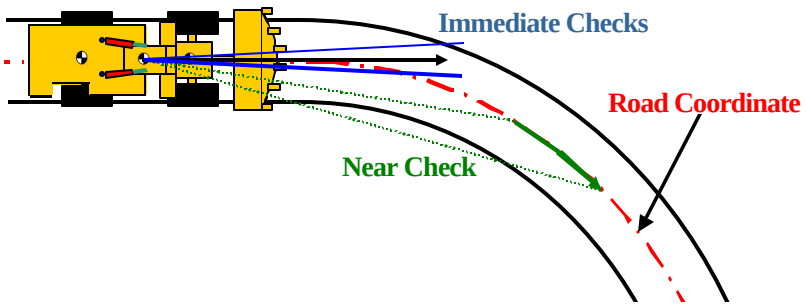

Figure 8: An illustration of the immediate checks and the near check for the error interpreter.

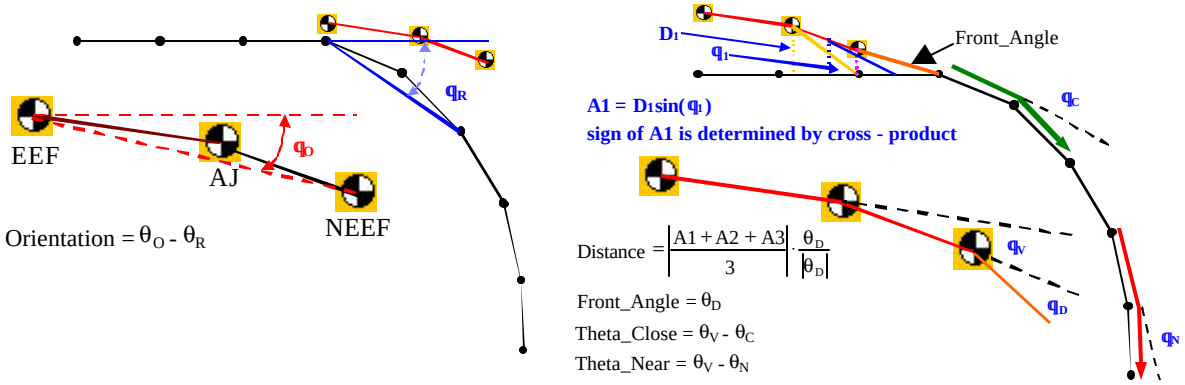

Figure 9: An illustration of the errors and the equations used by the error interpreter. The errors are presented as inputs to the fuzzy logic controller (HDMM).

$\theta_{1,2,3}$ = Angle from the (front endpoint, midpoint, rear endpoint respectively) of the referenced road segment to the (NEEF CG, NEEF to AJ segment midpoint, AJ respectively.)

A1 = Perpendicular distance to the AJ.

A2 = Perpendicular distance to the midpoint of the segment between the AJ and the NEEF CG.

A3 = Perpendicular distance to the NEEF CG.

θ_D = Angle from the NEEF CG to the front endpoint (Front Angle).

θ_V = Articulation angle, also the relative angle between the orientation of the EEF and the NEEF.

θ_O = Global angle the NEEF CG to the EEF CG.

θ_C = Relative angle between the referenced segment and the next segment in the trajectory.

θ_N = Relative angle between two contiguous road segments a distance away from the segments referenced by θ_C. The distance was based on the vehicle velocity.

θ_R = Global angle from the NEEF CG to the EEF CG.

The Distance error was the average distance from the vehicle's position to a reference segment along the required trajectory. The intent was to emulate driver's "aiming" behavior, which is usually observed among expert drivers of wheel loaders. The vehicle is represented by two

interconnected line segments with checker-boarded circles representing the locations of the centers of gravity for the EEF, Articulation Joint and NEEF respectively (from left to right).

The Front Angle error was the global angle from the front of the wheel loader to the end of the referenced road segment. The referenced road segment is the segment that shares the same global X or Y value based on the direction or quadrant of the road segment. At large distances from the trajectory, the use of this error prevented vehicular circular maneuvers. Another reason for including the Front Angle as an error was to eliminate a Bang - Bang controller effect as the vehicle approached the path from a large distance (Norris, 2001a).

The Orientation error was determined based on the global rotation of the vehicle relative to the trajectory. The orientation error prevented the wheel loader model from attaining equilibrium conditions unfavorable to the trajectory tracking control.

Another error, Theta Close was established as the difference between the vehicle's articulation angle and the angle between contiguous road segments. Avoiding calculations requiring a comparison of the calculated turning radius of the vehicle relative to the path's radius of curvature, the articulation angle of the vehicle was considered comparable to the angle between contiguous road segments on the roadway.

Under the same assumptions, Theta Near gauged the required change in articulation angle to match the curvature further along the trajectory. The distance between Theta Close and Theta Near was based on the vehicle velocity. Theta Near provided the look-ahead aspect of the controller, where major future changes in the roadway were detected that could affect future controller output.

The remaining immediate check was performed to determine when to shift the reference segment ahead as the vehicle followed the trajectory. All of the errors, with the exclusion of the movement check (Distance, Forward Angle, Orientation, Theta Close and Theta Near) were used as inputs to the HDMM or the virtual designer depending on the design phase.

Human Decision Making Model (HDMM)

Human decision-making for steering tasks is based primarily on judgment gained by experience. In terms of error correction, judgment is based on an inexact or "fuzzy" estimation. The design objective for the HDMM was to attain a controller with expert operator-like performance and a fixed output for a given set of input criteria. Consequently, the virtual operator incorporated a Mamdani type fuzzy controller (Norris, 2001a; Norris et al., 2001d). The desire was not to necessarily obtain an optimal controller but to approximate human operator performance characteristics. The controller minimized the error provided by the error interpreter with the objective of remaining parallel to the road at a minimum distance. The output from the controller was provided directly to the electrohydraulic steering system.

The inputs for many fuzzy logic based controllers include error rates (Norris, 2001a). Error rates were not input to the HDMM as the controller functioned as a velocity-based control. As opposed to a position controller, the velocity controller used the logic:

If error is large then increase rate of articulation.

If error is small then decrease rate of articulation

If error is minimal or zero - rate of articulation is zero

In the case of minimal error, the vehicle remained at the current articulation angle.

The lack of error rate inputs reduced the number of controller inputs and enhanced the system flexibility. The reference segments were shifted along the trajectory without the discontinuities

inherent with discrete changes in the measured errors. The result was the capacity to arbitrarily change the reference objective without retaining previous decision history within the fuzzy system inputs (Norris, 2001a; Norris et al., 2001d).

Fuzzification Interface

The fuzzification interface converted control inputs into linguistic information using membership functions. The membership functions were based on the outputs from the error interpreter. The input variables to the HDMM were Distance, Front Angle, Orientation, Theta Close, and Theta Near. To reduce computational effort, linear approximations were implemented. Fuzzy membership functions for the various linguistic variables were chosen to be Pi type (or trapezoidal) with Z and S shoulder functions (Lin and Lee, 1996). Intrinsic to the use of trapezoidal membership functions is a reduction in computational effort. The quantity of pertinent rules is significantly reduced when one or more inputs fall within the flat region of their respective trapezoidal membership functions. If the value falls in the flat region of the trapezoid, the degree of validity is one for that linguistic variable. Consequently, one linguistic variable is used in determining the rules corresponding to that input variable. A value falling in an intersecting sloped region of the membership function receives a degree of validity for the two adjacent linguistic variables, effectively doubling the amount of rules pertaining to the one input variable.

The regional approach to interpreting input information is an asset for modeling human performance. Human perception provides an inexact estimation of error. Exact error measurements are not possible; however, humans can readily determine if an error is "acceptable", “close” or “far” away from an objective based on experience. The boundaries between error classifications are where the uncertainty may occur. Trapezoidal membership functions incorporate the imprecise classification in their transitional sloped regions. The flat area atop the trapezoid represents a region of classification certainty.

The membership function parameters were tuned by hand to minimize the maximum distance variation from a given trajectory. The membership functions for the input variables are provided in Figures 10 and 11.

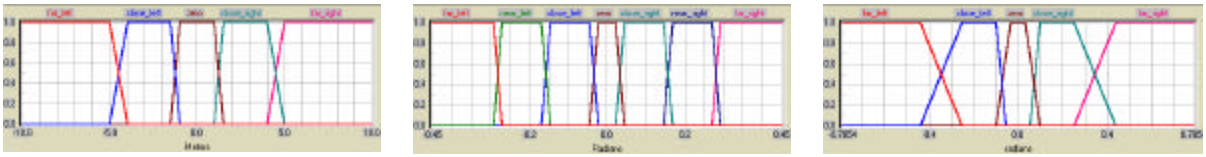

Figure 10: Left – The Distance membership functions. Middle - The Front Angle membership functions. Right – The Orientation membership functions.

Figure 11: Left – The Theta Close membership functions. Right – The Theta Near membership functions.

The Rule Base and Defuzzification Method

The rule base was derived based on heuristic knowledge (Lin and Lee, 1996). A hierarchical technique was used based on the importance of the inputs relative to their linguistic variable regions (Norris et al., 2001e; Norris, 2001a). The technique may be summarized as follows. The Distance input was used as the primary metric for the rule base. If the current distance of the vehicle was designated "far" away from the trajectory, the primary objective was to return to the path and the Front Angle was the only error used to determine the control effort. The Front Angle error was important in the region because it prevented vehicular oscillation between negative and positive maximum commands producing high frequency oscillation or a Bang-Bang control effect.

As the wheel loader reached the "close" Distance region, the Front Angle input was no longer important as the objective was not to take the minimum distance back to the trajectory. The effect of the Orientation error became important as the emphasis changed to gradually smoothing the trajectory of the vehicle as it approached the roadway.

Once the wheel loader reached the "zero" Distance region, Orientation replaced Distance as the metric. At that point, the interactions of Orientation and Theta Close determined the steering control signal. Once Orientation entered its linguistic variable zero region, Theta Close and Theta Near were used to make the control decision.

The hierarchical technique reduced the size of the rule base by 98.7%, from 6125 rules covering the entire region of possibility to 81, which substantially improved computation speeds. The rules as well as the logic behind the control strategy were incorporated within the rule-base in a technique resulting directly from the work presented in this paper. The technique is known as a fuzzy relational control strategy (FRCS) employing fuzzy relational control variables (FRCV) (Norris et al., 2001e).

The control laws employed in the rule base were a complete set of the following sample control rules:

> IF Distance is far left, AND Front Angle is near left, THEN steering action is Right3.
> IF Distance is close left, AND Orientation is close left, THEN steering action is Right2.
> IF Distance is zero, AND Orientation is far left, AND Theta Close is far left THEN steering action is Right3.

The steering actions were defuzzified using symmetric membership functions where Right3 corresponded to a controller output of 1.0 (Left3 = -1.0), Right2 = 0.6667 (Left2 = -0.6667), Right1 = 0.3333 (Left1 = 0.3333) and Zero = 0.0.

The defuzzification technique was a form of the Center of Area with singletons. The technique is also known as the Center of Maximum (CoM). The center of maximum technique served as a "best compromise" technique that greatly reduced the required computation effort, yet maintained smooth output functions.

The HDMM was designed using the FuzzyTech™ software package and generated as a C program. The error interpreter and trajectory planner were programmed in C. The code for the virtual operator was implemented in user-code block diagram form within MatrixX System Build. Vehicle simulations, including the virtual operator, designed for a length of 90 s with a 50 Hz sampling rate, required 18 sec. for completion on an HP C180 workstation.

The Virtual Trainer

The virtual trainer consisted of training procedures for the virtual operator. The purpose of the virtual trainer was to draw on training data from an expert human operator to train the virtual operator to perform in like fashion (Norris, 2001a). The training for the wheel loader steering design problem involved manually tuning the membership functions of the virtual operator to reduce the distance and orientation errors.

The Virtual Designer

The design selected for adaptation was termed the virtual modulation curve (VMC). The modulation curve served as a map between the electrical signal from the controller interface from the operator and the spool displacement controlling the flow to the hydraulic cylinders. The objective was to demonstrate that the modulation curve could be designed for multi-dimensional input-output mapping as required to meet additional qualitative objectives.

The physical structure, as presented in Figure 12, had several components. As noted in the figure, the VMC mapped the inputs from the X-axis to the corresponding outputs on the Y-axis. The input and output ranges were both within $\pm$ 1.0. The input to the VMC and the output from the VMC were bounded by saturation limits of $\pm$ 1.0.

Two notable regions were components of the VMC, a dead zone and a saturation region. A dead zone is a region where the input signal was mapped to a zero output. In Figure 12, the dead zone is represented by the region bounded by zero and X_{min} (or - X_{min} on the negative input axis.) The dead zone served the purpose of ensuring complete shut-off of the steering and filtering low magnitude noise from the steering command signals. The saturation region represented the region where steering command signals were mapped to a 1.0 output, irrespective of value. The saturation region was bounded from X_{max} to 1.0 in the positive input quadrant (mapped to -1.0 in the negative input quadrant and bounded by - X_{max} and -1.0). The purpose of the saturation region was to filter low magnitude noise from the upper regions of the steering command mapping.

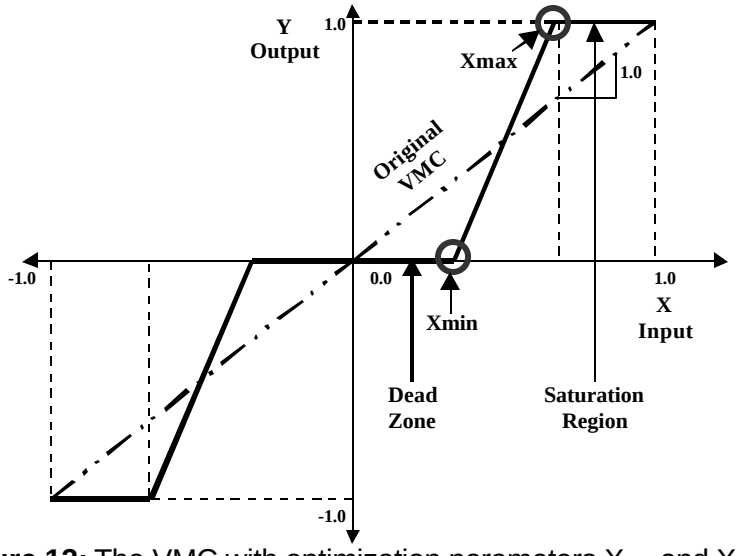

Figure 12: The VMC with optimization parameters X_{min} and X_{max}.

A linear approximation was used to map points that fell between the points identifying the saturation region and the dead zone (X_{max} and X_{min}). The slope of the linear region, with the removal of the dead zone (or $X_{min} = 0$), will be known as the Gain of the VMC.

The VMC was parameterized with the two terms X_{min} and X_{max}. Consequently, they were determined to be the adaptable components for off-line and on-line training. Retaining the anti-symmetric property for negative valued input signals, the values of $-X_{min}$ and $-X_{max}$ were used for the corresponding saturation region and dead zone boundaries (Norris, 2001a).

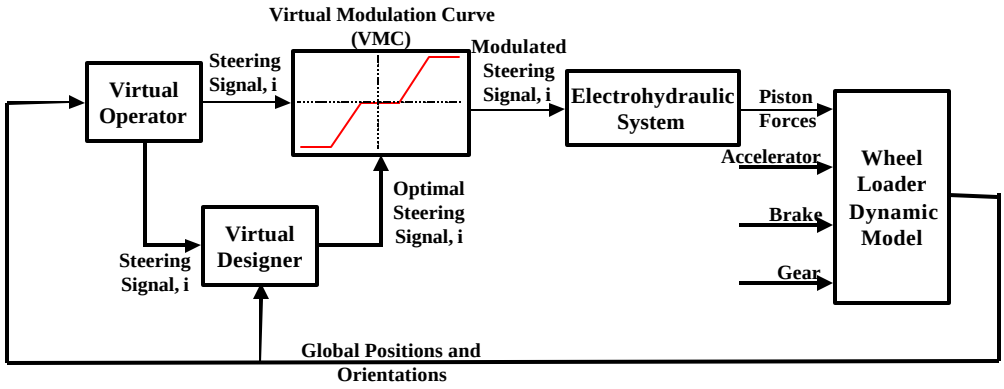

Figure 13: The virtual designer as applied to the two-dimensional modulation curve case.

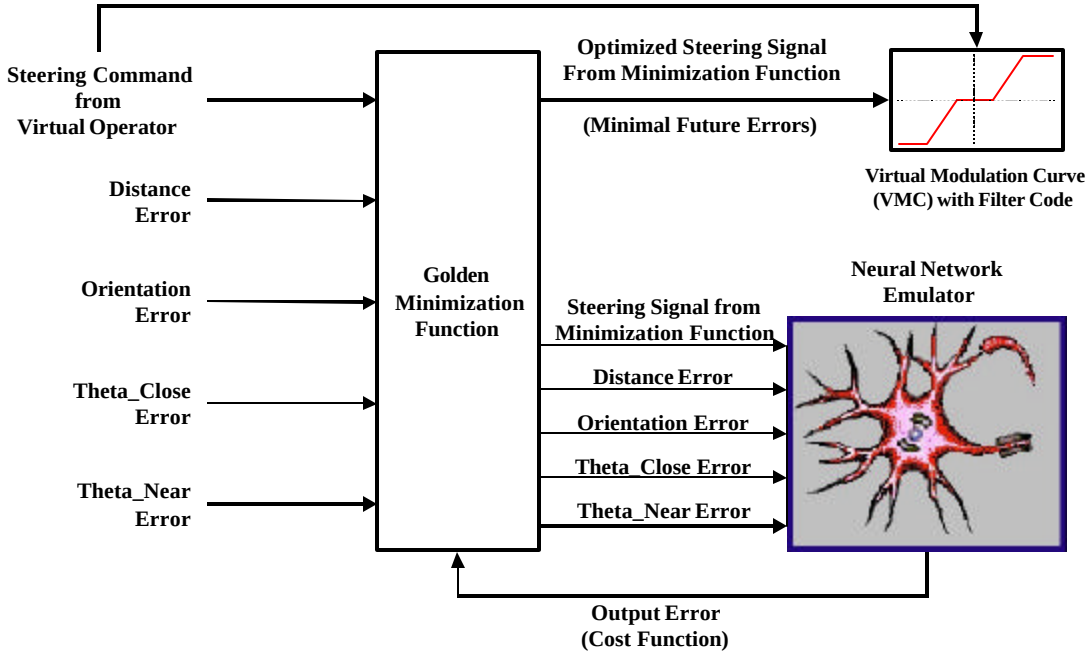

Figure 14: The structure of the virtual designer for on-line training including the emulator neural network, Golden minimization function and the augmented VMC.

A neural network emulator structure was developed to adequately model the wheel loader and virtual interpreter systems. The inputs to the neural network were the HDMM steering command, and the Distance, Orientation, Theta Close and Theta Near errors. The emulator

predicted the future Distance, Orientation, Theta Close and Theta Near errors for an interval of 0.1 sec. simulation time and was trained off-line with data from generated random trajectory and vehicle orientations from multiple simulations (Norris, 2001a).

As implemented on-line and demonstrated in Figure 14, the emulator was coupled with a global minimization function. As the simulation progressed, the steering and error signals were sent to the virtual designer. Using the signals as inputs to the neural network, the global minimization function determined the optimal steering signal that would produce minimal error. Both the virtual operator's steering signal and the steering signal from the virtual designer were sent to the filter algorithm for adaptation. Additional details are presented in (Norris, 2001a).

In the final phase, the virtual designer was removed while retaining the adapted modulation curve as shown in Figure 15. Figure 16 demonstrates the incorporation of the filter in the human operator - vehicle system. To meet the real-time constraints, the neural network and filter algorithms were encoded in C. The derived functions were implemented within user code blocks in the MatrixX simulation.

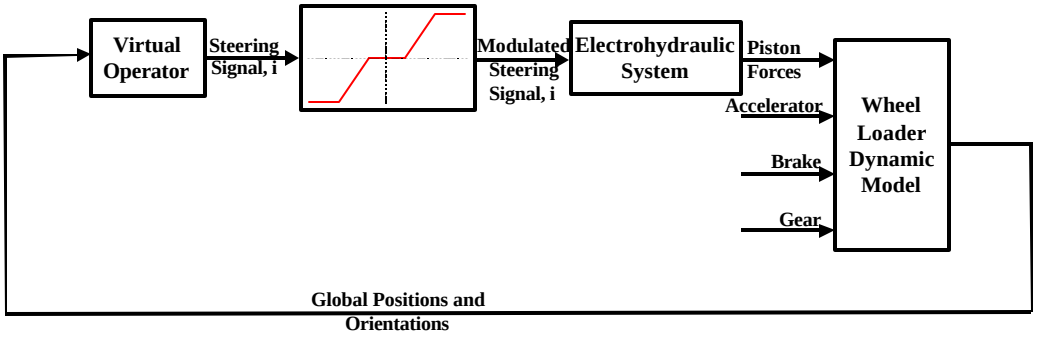

Figure 15: Application of the trained two-dimensional modulation curve with the virtual operator.

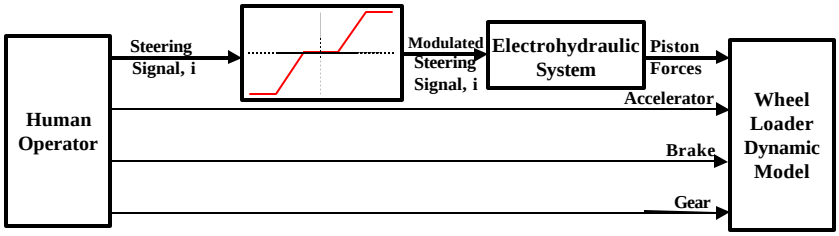

Figure 16: Implementation of the trained two-dimensional modulation curve with a human operator.

System Integration

The vehicle model was subject to validation. One of the validations included obtaining the step response of the vehicle's articulation angle as demonstrated in Figure 17. As demonstrated in the figure, the vehicle's response possessed second order characteristics. The overshoot and extended settling time resulted from the characteristics of the hydraulic steering system. In the hydraulic steering system, both the hydraulic fluid and the steering cylinders provided dampening characteristics. As a cylinder approached to its full extension, the hydraulic fluid on the opposing side of the piston head served as a spring-like component.

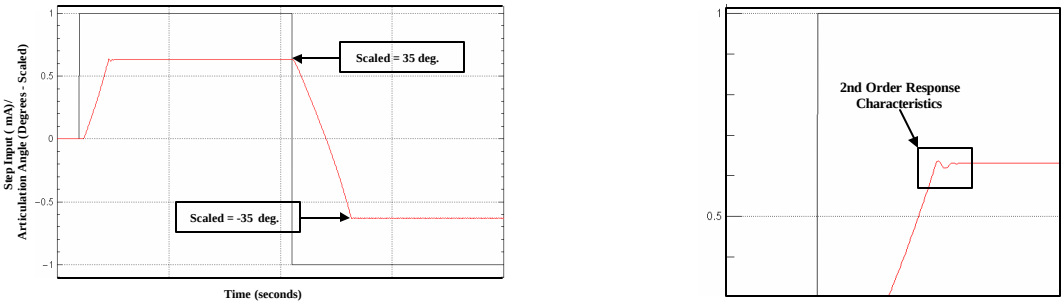

Figure 17: Wheel loader step response in terms of scaled articulation angle.

The vehicle-virtual operator system was tasked to pass the SAE standard steering evaluation course for off-road vehicles (SAE Standard J1511, 1994). The requirements were for the vehicle to remain within the course boundaries while the operator drove through the course at a constant 16 km/hr ± 2 km/hr. As shown in Figure 18, the virtual operator successfully completed the course. The articulation angles generated from the virtual operator's steering command are demonstrated next to the steering course. Changes were made in the Distance membership function, maintaining the same rule base, and the new operator was tested on the SAE course test was as demonstrated in Figure 19.

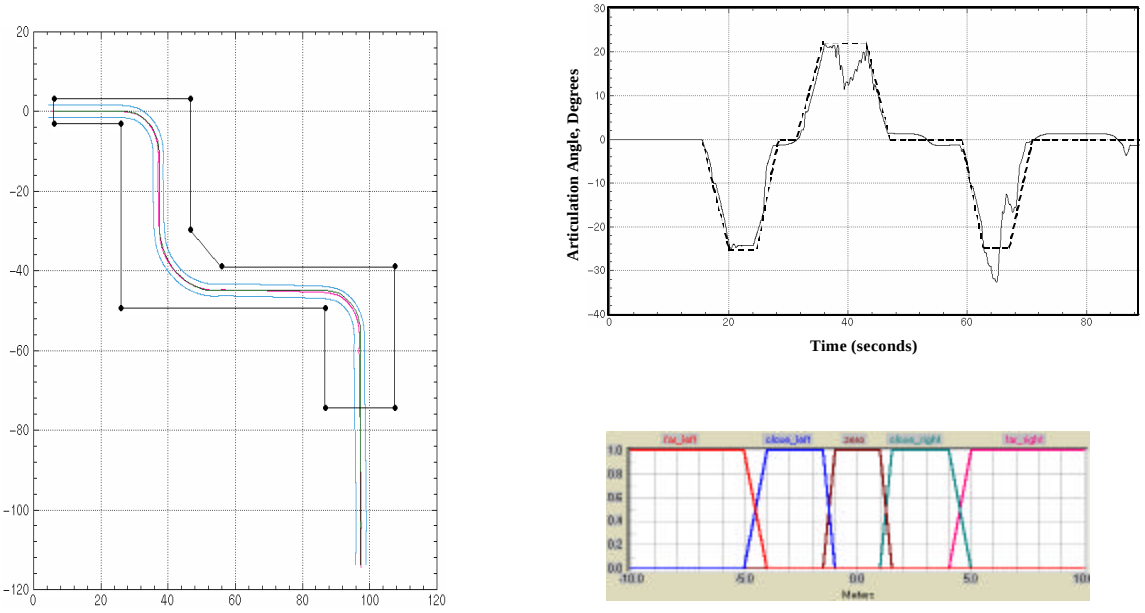

Figure 18: The virtual operator steering the vehicle through the SAE standard steering evaluation course (bounded with straight lines) with the corresponding plot of the wheel loader articulation angles. The centerline of the trajectory in the top figure consists of the coordinates of the NEEF Center of Gravity. The outer lines represent the location of the tire connection coordinates. The lower right hand corner of the figure contains the distance membership function.

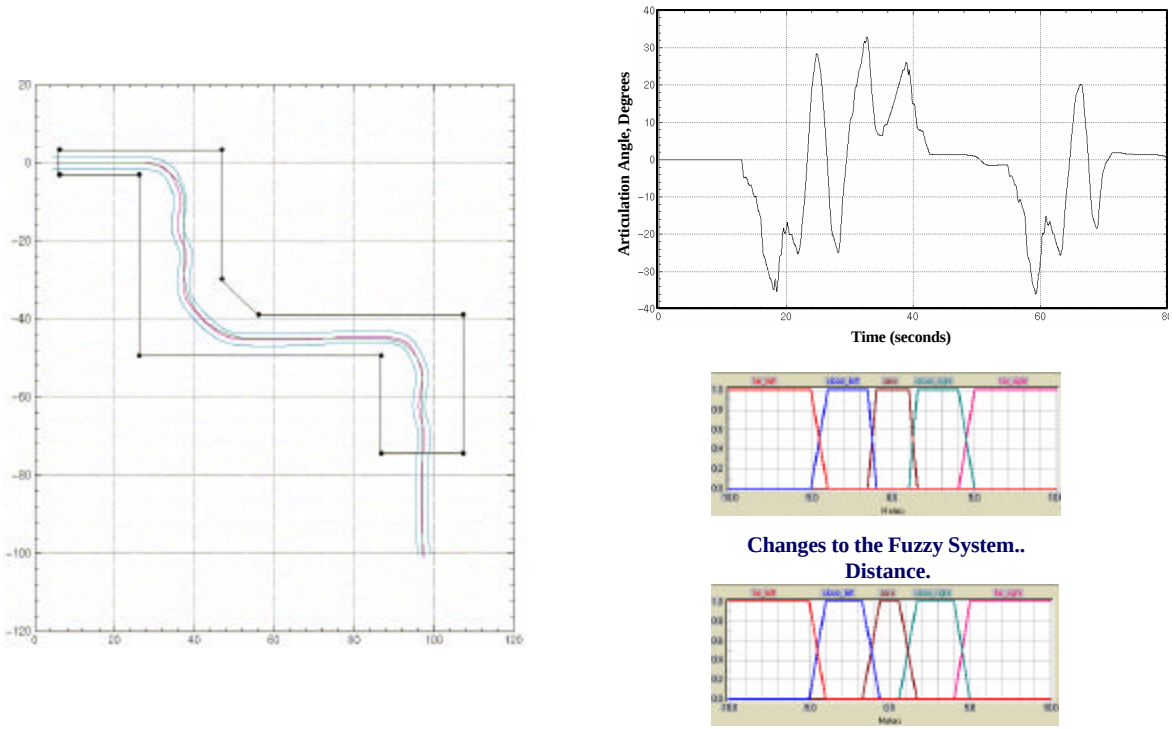

Figure 19: A modified virtual operator steering the vehicle through the SAE standard steering evaluation course (bounded with straight lines) with the corresponding plot of the wheel loader articulation angles. The centerline of the trajectory in the top figure consists of the coordinates of the NEEF Center of Gravity. The outer lines represent the location of the tire connection coordinates. The lower right hand corner of the figure contains the distance membership function as modified from the original.

Comparing the trajectories in Figure 18 and 19, changes to the membership functions, essentially the experience of the virtual operator, produced another operator that overcompensated at a small turning radius. This is a problem with some less experienced operators.

The proof of concept for virtual design tools was achieved in the following way. The HOPM was trained as an expert operator using a modulation curve with a slope of one and an intercept of zero (X_{min} = 0.0, X_{max} = 1.0). Once the operator was trained, X_{min} and X_{max} were varied to a value for x of 1.275 and an x intercept of 0.1175, effectively providing a dead zone. For the off-line case, a Lagrangian optimization function was used to adapt the parameters of the VMC. The function was a built-in function of the MatrixX Optimization module. The VMC parameters were adjusted to minimize the cost function in equation 21 and the operator's overall performance on the SAE road course. Distance represented the vector of Distance errors collected at a 50 Hz sampling rate throughout the simulation. The optimized modulation curve parameters are presented in Table 1.

$$\text{Cost} = \sqrt{(\max(\text{Distance}))^2 + (\min(\text{Distance}))^2} \tag{21}$$

The on-line training algorithm as discussed in the virtual designer section, was initialized in a similar manner. The results of the on-line training showed that the consistent operator's

performance was improved though the parameter values for X_{min} and X_{max} did not converge due to the differing controller structures. The results demonstrate that it is possible to optimize the performance of an operator with a two-dimensional VMC and that it will be feasible to apply a virtual modulation surface (VMS).

Table 1: Results of the optimization function applied to the VMC, the off-line virtual designer.

	X_{min}	X_{max}	Max Distance	Min Distance	Cost
Original	0.0	1.0	1.464	-0.693	1.619736
Parameter Initialization	0.1175	1.275	2.1434	-0.834	2.3
X_{min} Constant X_{max} Varied	0.0	1.00478	1.3613	-0.3298	1.40068
Optimization Results	0.010662	1.00478	1.28412	-0.3267	1.325027

Table 2: Simulation results from the on-line virtual designer.

	X_{min}	X_{max}	Max Distance	Min Distance	Cost
Original	0.0	1.0	1.464	-0.693	1.619736
Parameter Initialization	0.1175	1.275	2.1434	-0.834	2.3
Optimization Results	Variable	Variable	1.295	-0.3298	1.336336

Conclusion

The virtual design tool framework has been applied to an electrohydraulic steering system design for a wheel loader. The real time wheel loader model was complex, coupled, and all of the parameters were configuration dependent.

A real time human performance model, for steering a four-degree of freedom articulated vehicle along a trajectory, has been developed. Human like error interpretation and a common sense rule base have been used to successfully perform real-time control system decision-making. The rule base remained fixed while the perception of the model was altered to attain different performance characteristics. The techniques used to develop the performance model are viable for many other applications including situations where an operator model is required, for human performance studies, qualitative system design or in autonomous equipment algorithm development.

The solution demonstrates that the virtual design tool framework has the capacity to capitalize on the full potential of virtual environments. A few of the possible application areas including rapid prototyping of vehicle system designs, autonomous vehicle research, the study of human controllability issues and “operator-in-the-loop” design techniques. The virtual design tool framework, along with the flexibility inherent in mechatronic and electromechanical systems, will provide a better and safer control system design strategy, employing performance characteristics desired by a human operator.

References

Bakker, E., H. Nyborg, H. Pacejka. 1987. Tyre Modeling for Use in Vehicle Dynamic Studies. *SAE Vehicle Dynamics and Simulation*, paper no. 870421.

Connacher, H., S. Jarayam, and K. Lyons. 1996. Integration of Virtual Assembly with CAD. *Symposium on Virtual Reality in Manufacturing, Research and Education*, October.

Lin, C., and Lee, G. 1996. *Neural Fuzzy Systems: A Neural-Fuzzy Synergism in Intelligent Systems.* Princeton, New Jersey: Prentice Hall.

Norris, W. R. 2001a, A Design Framework for Qualitative Human-in-the-Loop System Development. Diss., University of Illinois Urbana-Champaign, Urbana, IL.

Norris, W. R., Z. Kantarcioglu, R. Sreenivas, and Q. Zhang 2001b. Technique for the Virtual Simulation of a Wheel Loader with an Electrohydraulic Steering System. Submitted to SAE Vehicle Dynamics and Simulation.

Norris, W. R., R. Sreenivas, and Q. Zhang. 2001c. An Articulated Wheel Loader Path Tracking Algorithm and Error Interpretation. Submitted to Journal for the IEEE Control Systems Society.

_____. 2001d. A Real -Time Human Operator Performance Model Using Fuzzy Logic. Submitted to IEEE Control Systems Society.

_____. 2001e. Hierarchical Rule-Base Reduction for a Fuzzy Logic Based Human Operator Performance Model." Submitted to ANNIE 2001.

_____. 2001f. Real Time Virtual Simulation of a Wheel Loader with an Electro-hydraulic Steering System. Submitted to SAE Vehicle Dynamics and Simulation.

Pauling, P., and C. S. Larson 1988. Simulation of an Articulated Wheel Loader Including Model for Eathmoving Tires. *SAE Earthmoving Conference*, paper no. 880777.

Purschke, F., M. Schulze, and P. Zimmermann. 1998. Virtual reality-new methods for improving and accelerating the development process in vehicle styling and design. *Computer Graphics International*, 789-797.

SAE Standard J1511, 1994. Steering for Off-Road, Rubber-Tired Machines. Revised February 1994.

Willhelm, R., and K. Nagalla. 1996. Computer Representation of Virtual Parts. *Symposium on Virtual Reality in Manufacturing, Research and Education*, October.