

A Fuzzy Controller For Vehicle Rendezvous and Docking

Graham J. Eatherley and Emil M Petriu, *Senior Member, IEEE*

Abstract—This paper discusses a microprocessor-based fuzzy controller and an on-board two-dimensional relative position measurement system for the rendezvous and docking of an experimental truck and trailer.

I. INTRODUCTION

BACKING a truck and trailer to a loading dock is a difficult exercise for all but skilled truck drivers. Normal driving instincts typically lead to erroneous movements, and a great deal of practice is required to develop the requisite skills. This severely nonlinear problem has received considerable attention and interest.

In Nguyen and Widrow's simulation [1], the truck only moves backward with a goal to dock the trailer parallel to the loading dock. A two-stage neural network controller requiring about 20000 backups for training performed well in backing up the truck from difficult initial starting configurations. Kosko's solution [2] uses both neural networks and fuzzy logic. The fuzzy rulebase contains 105 "hand-crafted" fuzzy rules which successfully control the truck and trailer even in jackknifed positions. Plumer's solution [3] consists of a feed-forward neural network to control the steering of the vehicle using local potential field information. Koza [4] applies a genetic programming technique to evolve superior backer-uppers.

This paper approaches the theoretical "truck and trailer backing" problem from a more realistic "truck and trailer docking" perspective in which both forward and reverse motion is allowed. It is shown that a symmetrical analysis of the problem in polar reference frames leads to a smaller, more robust set of control rules for steering of the truck.

In addition, the paper presents experimental results using a scale model truck and trailer and an original on-board optical sensor which measures the 2D relative position (docking angle and distance) of the vehicle in polar coordinates.

II. PROBLEM ANALYSIS

Problem analysis is simplified by using polar frames attached to the dock, trailer, and truck instead of the usual Cartesian coordinate system. The 2D position of the trailer relative to the dock is completely determined, as illustrated in Fig. 1, by three state variables: α , β and d .

Manuscript received May 9, 1994; revised January 23, 1995. This work was partially supported by the Natural Sciences and Engineering Research Council of Canada and by Bell Canada.

The authors are with the Department of Electrical Engineering, University of Ottawa, Ontario, K1N 6N5, Canada.
IEEE Log Number 9411494.

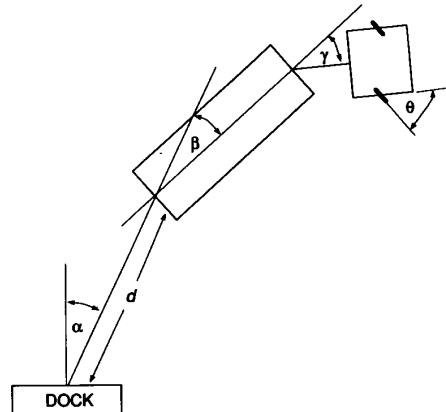


Fig. 1. Polar analysis.

α is the angle between the normal from the dock and the line connecting the dock and the trailer, β is the angle between the trailer axis and the line connecting the dock and the trailer, and d is the distance from the dock to the trailer. The control variable is the angle γ between the trailer and truck axes which is completely determined by the steering wheel angle θ . The problem of backing up the trailer to the dock is to reduce the state variables to zero so that $\alpha = \beta = d = 0$.

How we diminish α and β will affect the trajectory the trailer follows in backing up to the dock. If β goes to zero more quickly than α , the trailer will approach the dock more directly, whereas if α goes to zero more quickly than β , then the trailer will follow a highly curved trajectory. A compromise solution and one that is easy to implement is to control γ to maintain $\alpha = \beta$. In this fashion, both α and β approach zero at the same rate so that the trailer follows more or less a circular trajectory in approaching the dock.

Referring to Fig. 1 and assuming that the truck remains in reverse, if the controller is acting to maintain $\alpha = \beta$ it can essentially ignore d for steering control since the trailer must approach the dock under these conditions. Distance d is only used to control the vehicle's speed and stop it upon reaching the dock. The resulting control strategy is somewhat subjective but intuitive. If $\alpha > \beta$ the control strategy when moving in reverse is to increase β by moving the cab to the left (by steering right) to correct, and vice versa. Notice that this strategy is also appropriate in the other quadrant (left of the vertical from the dock).

Forward motion is used only when the trailer is in a position from which it would be difficult to approach the dock with

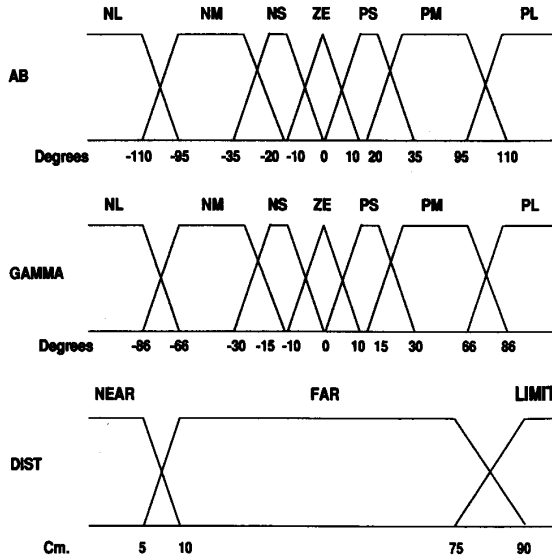


Fig. 2. Input membership functions.

proper alignment; i.e., if it is much different from β or if γ is beyond the "jackknife" angle to the left or right. We exploit the symmetry by using the above rules in reverse; i.e., we will still maintain $\alpha = \beta$, but we now move the cab to the right to increase β . Notice that we now steer right to move the cab right.

III. FUZZY CONTROL SYSTEM

By using polar coordinates and exploiting natural left-right and forward-reverse symmetries, the developed fuzzy system uses three input fuzzy variables: $AB = \alpha - \beta$, $GAMMA = \gamma$, and $DIST = d$ and three output variables: $STEER = \theta$, $SPEED$ and $DIRN$. Both AB and $GAMMA$ have seven values each, requiring 49 rules for all combinations of the steering function. The truck speed is determined only by its distance d from the dock and is controlled by three simple rules. As a result the size of the rule base is reduced by an order of magnitude relative to the classical solution which would lead to 343 rules ($7\alpha * 7\beta * 7\gamma$).

A. Membership Functions

The input variables AB and $GAMMA$ were defined to have seven membership functions each, namely: Negative Large, Negative Medium, Negative Small, Zero, Positive Small, Positive Medium and Positive Large. The $DIST$ variable has been defined to have only three membership functions: Near, Far, and Limit.

Some experimentation in the design of these values was done to "fine-tune" the control performance, although the actual values used are not critical for successful docking of the trailer (Fig. 2). Note the nonlinearity inherent in the width of the input membership functions.

All three output variables $STEER$, $SPEED$, and $DIRN$ are singletons defined as shown in Fig. 3. The vehicle steering variable, $STEER$, defined to have seven singleton points: Left

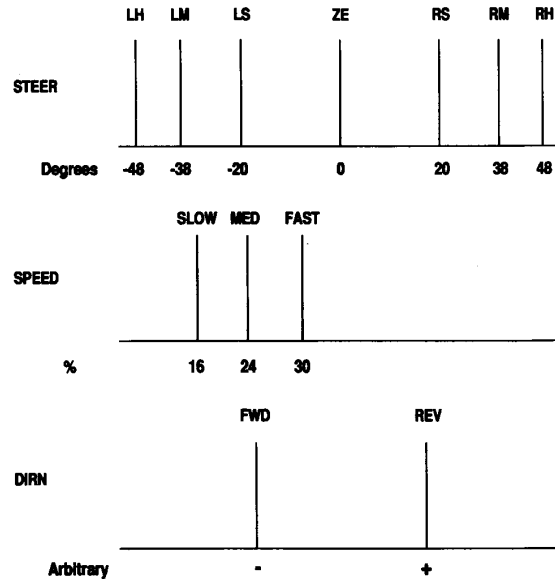


Fig. 3. Output membership functions.

High, Left Medium, Left Small, Zero, Right Small, Right Medium, and Right High, is the output parameter of most consequence, and hence its membership function points are critical. The distance between the singleton points is used to establish finer control near the center-point. The outermost singleton points were established by experimentation at the practical limit of the steering servo, both hard-left and hard-right.

The $SPEED$ variable, defined to have three singleton points: Slow, Medium, and Fast, is used primarily to slow down the vehicle as it approaches the dock.

$DIRN$ is a binary output variable which is used by the motor control interrupt routine to drive the truck in the appropriate direction. Its two singleton points, FWD and REV, are not critical and were simply set to arbitrary positive and negative values. In fact, there is nothing "fuzzy" at all about this variable—we either go forward or reverse.

B. Rule Base

As represented here, the problem has been converted from a "truck backer-upper" to a "truck docking problem." To some degree, allowing both forward and reverse motion simplifies the backing-up task; however, it does introduce another interesting control problem, that of how and when to switch between forward mode and backup mode.

The rule base is organized as a matrix. The rule definition begins by filling in the steady-state zero in the center where no control action is required as the trailer approaches the dock with perfect alignment (AB and $GAMMA$ are zero).

C. Rules to Control Direction of Travel

As long as the truck is in a position from which it can successfully dock, it will continue in reverse. The controller need only switch to forward in two situations. If γ is greater than about 75° left or right, the truck and trailer are effectively

STEER/DIRN RULES		GAMMA							
		NL	NM	NS	ZE	PS	PM	PL	
AB	NL	LH/F	LH/F	LH/F	LM/F	LS/F	RS/F	RM/F	
	NM	LH/F						RH/F	
	NS	LH/F		LM/R	LS/R	RS/R		RH/F	
	ZE	LH/F		LS/R	ZE/R	RS/R		RH/F	
	PS	LH/F		LS/R	RS/R	RM/R		RH/F	
	PM	LH/F						RH/F	
	PL	LM/F	LS/F	RS/F	RM/F	RH/F	RH/F	RH/F	

Fig. 4. Steering/direction rule base.

“jackknifed” and cannot recover in reverse at any steering angle. In addition, if α is very much different than β , then the truck cannot align properly with the dock by moving in reverse. The *outer* ring of the rule matrix corresponds to these extreme situations and therefore, we will switch to forward motion whenever the system enters this area.

The *inner* ring of the rule matrix corresponds to alignment positions where the angle of the truck and trailer, γ is small and α is very close to β . Under these conditions, the trailer is definitely under control and the backing-up task completes successfully. For the in-between case (shaded area of the matrix) where the aforementioned angles are moderate, the controller does not switch to either forward or reverse. That is, the truck *maintains* its current direction until it enters either the inner or outer ring. In this way, we allow the truck to go forward until alignment is considerably improved before switching to reverse.

D. Steering Rules

Because of the symmetry of the problem, only the upper or lower diagonal of this matrix needs to be designed. It is only necessary to remember that our goal is to maintain α equal to β .

If $\alpha > \beta$, then we must have the truck to the left of center when backing up. Whether we need to *steer* left or right depends on where the truck presently is; i.e., if the truck is already to the left we may not have to alter the steering at all. On the other hand, if the truck is to the right and we are in reverse, then we will have to steer right. Using these simple heuristic notions, we fill in the rest of the table. By careful examination of the system's behavior, it is possible to fine-tune the rule base to smooth the control process and eliminate unnecessary switching of direction. The final rule base giving both the fuzzy steering and direction outputs is shown in Fig. 4.

E. Defuzzification

The *defuzzification* process is used to convert fuzzy outputs to crisp numbers. In our case the *STEER* variable is defuzzified using the modified “centroidal” defuzzification method [5, ch.

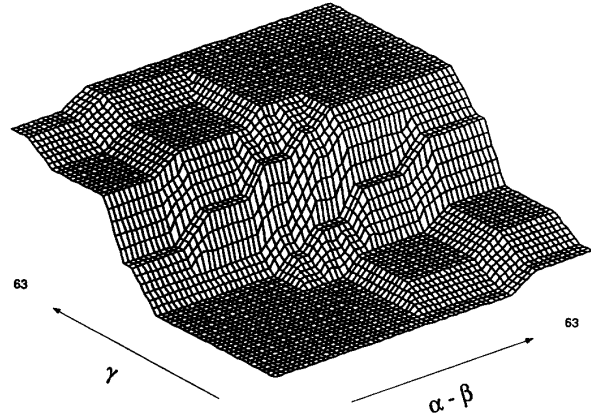


Fig. 5. Fuzzy truck I/O characteristic.

4]. The crisp value for the steering angle is calculated using the relation:

$$\theta = \frac{(LH \cdot \theta_{LH} + LM \cdot \theta_{LM} + LS \cdot \theta_{LS} + ZE \cdot \theta_{ZE} + RS \cdot \theta_{RS} + RM \cdot \theta_{RM} + RH \cdot \theta_{RH})}{(LH + LM + LS + ZE + RS + RM + RH)}. \quad (1)$$

The variables XX , for $XX \in \{LH, LM, LS, ZE, RS, RM, RH\}$ are current membership degree values of the fuzzy output *STEER* calculated for a specific instance of the analog inputs α - β and γ using the classical “max-min” compositional mode of inference [5, ch. 4]:

$$XX = (AB \times GAMMA) \cdot C_RULE_XX. \quad (2)$$

Although the “*DIRN*” and “*SPEED*” outputs do not warrant this analysis, the same defuzzification process is used in order to simplify the coding of the fuzzy engine.

After the fine-tuning of the membership functions, the I/O characteristic of the fuzzy controller (Fig. 5) exhibits less pronounced nonlinearities around the central region. Such a characteristic allows for smoother docking behavior. The nonlinearities observed in the medium region of the I/O characteristic are due to the extra wide medium regions (PM and NM) of the input variables which have been designed to discourage repeated switching of direction.

IV. EXPERIMENTAL DESIGN

An experimental scale model truck and trailer has been implemented as a test bed for the fuzzy controller. All truck control functions are implemented using a Motorola MC68HC11E9 eight-bit microcontroller. A Motorola-designed knowledge base generator (KBG) and Fuzzy Engine are used to develop and run the fuzzy system. Each rule as implemented in the “fuzzy truck” requires no more than four bytes of memory. A centroid defuzzification is performed in under 50 μ s.

Angle γ is measured by an on-board shaft encoder. The docking distance d and angles α and β are measured using an on-board azimuth and range measurement system and a pseudorandom bar-coded cylindrical dock.

The use of pseudorandom encoding for position measurement has been previously reported [6]. The method requires only a one-bit wide code track and is based on the window property of the pseudorandom binary sequences $\{S(p)|p = 0, 1, \dots, 2n - 2\}$.

The azimuth and range measurement system consists of a laser barcode reader and scanning servo mounted on the back of the trailer. A local controller finds the dock and keeps the scanner pointed at the center of the coded cylinder, resolving β . It also filters the scanner's readings and detects and translates the pseudorandom code. Finally, it calculates the current values of the docking distance d and angle α using (3) and (4) developed further and transmits these values to the master microprocessor which runs the fuzzy controller.

The geometry of the docking angle and distance measurement problem is illustrated in Fig. 6 where B is the location of the optical scanner, O is the center of the encoded cylinder, and A is the desired docking point. AB is the docking distance d to be measured. Angle ζ is obtained from the optical reader, and β is obtained from the pseudorandom codes read on the encoded cylinder. Angles ϵ and η will result from the pseudorandom codes read on the encoded cylinder.

Elementary trigonometry gives the docking distance:

$$d = r \cdot \frac{\sqrt{[\sin^2(\epsilon + \zeta) + \sin^2 \zeta - 2 \cdot \cos \eta \cdot \sin(\epsilon + \zeta) \cdot \sin \zeta]}}{\sin \zeta} \quad (3)$$

and the docking angle:

$$\alpha = \arcsin \frac{\sin \eta \cdot \sin(\epsilon + \zeta)}{r \cdot \sqrt{[\sin^2(\epsilon + \zeta) + \sin^2 \zeta - 2 \cdot \cos \eta \cdot \sin(\epsilon + \zeta) \cdot \sin \zeta]}} \quad (4)$$

V. CONCLUSIONS

The developed fuzzy controller has been tested from various initial configurations and exhibits appropriate docking behavior at speeds of approximately 20 cm/sec.

Fine-tuning of the membership functions was required to produce good docking behavior. Notice how the nonlinearity of the control problem is reflected in the membership functions. In particular, the membership functions of ZE , NS , and PS for variables $GAMMA$ and AB are much narrower than those for the extreme positions. This tightening up of the functions near the steady-state zero is necessary to produce adequate control behavior as the truck makes final corrections near the end of the docking task. The extra wide medium regions of the input membership functions (NM and PM) are necessary to discourage repeated switching of direction.

The experimental positioning system based on the miniature barcode reader is capable of tracking the center of the dock at ranges up to 2 m while moving at moderate speeds. We are currently working to extend this range using simple optics and digital filtering.

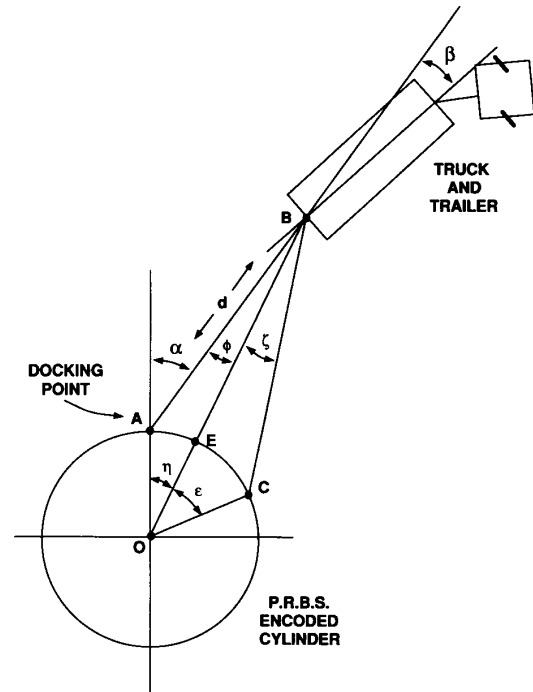


Fig. 6. Optical position sensing system.

ACKNOWLEDGMENT

The authors wish to thank L. Korba from NRC Ottawa for his support and readiness to discuss relevant problems. The authors also thank Symbol Technologies Canada for providing the laser barcode reader used in the experiments.

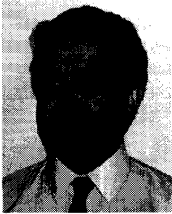
REFERENCES

- [1] D. Nguyen and B. Widrow, "The truck backer-upper: An example of self-learning in neural networks," in *Proc. IJCNN-89 Int. Joint Conf. Neural Networks*, June 1989, vol. II, pp. 357-363.
- [2] B. Kosko, "Comparison of fuzzy and neural control systems," *Neural Networks Fuzzy Systems*, pp. 339-361, 1992.
- [3] E. S. Plumer, "Neural network structure for navigation using potential fields," in *Proc. IJCNN-92 Int. Joint Conf. Neural Networks*, July 1992, vol. I, pp. 327-332.
- [4] J. R. Koza, "A genetic approach to the truck backer upper problem and the inter-twined spiral problem," in *Proc. IJCNN-92 Int. Joint Conf. Neural Networks*, July 1992, vol. IV, pp. 310-318.
- [5] W. Pedrycz, *Fuzzy Control and Fuzzy Systems*, 2nd ed. Toronto: Wiley, 1993.
- [6] E. M. Petriu and J. S. Basran, "On the position measurement of automated guided vehicles using pseudorandom encoding," *IEEE Trans. Instrum. Meas.*, vol. 38, no. 3, pp. 799-803, 1989.



Graham J. Eatherley received the B.Sc. degree in mathematics and physics from Mount Allison University in 1967. Recently, he completed a master of applied science in engineering at Ottawa University.

He has worked in various research positions in the Canadian Forces and Bell Canada during the past 25 years. He is currently on leave of absence from Bell Canada working toward the Ph.D. degree in electrical engineering at Ottawa University, Ottawa, Canada. His current interests are in AI applications to robotics.



Emil M. Petriu (M'86-SM'88) received the Dipl.Eng. and Dr.Eng. degrees in electrical engineering from the Polytechnic Institute "Traian Vuia," Timisoara, Romania, in 1969 and 1978, respectively.

Dr. Petriu is currently a Professor in the Department of Electrical Engineering at the University of Ottawa, Canada. From 1990 to 1993 he was the Director of the Ottawa-Carleton Institute for Electrical Engineering.

Dr. Petriu has co-authored three books, received two patents, and published over 50 scientific papers. His current research interests include robot sensors, multisensor fusion, and intelligent autonomous robot systems.

Dr. Petriu is Vice Chair of the Publications Committee of the IEEE Instrumentation and Measurement Society and a member of the Editorial Review Committee of IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT. He is a registered Professional Engineer in the Province of Ontario, Canada.