Fuzzy Logic in Control: Truck Backer-Upper Problem Revisited

Andri Riid and Ennu Rüstern
Department of Computer Control
Tallinn Technical University
Ehitajate tee 5, Tallinn, 19086, Estonia , e-mail: andri@dcc.ttu.ee

***Abstract*--Truck backer-upper problem, considered an acknowledged benchmark in nonlinear system identification, is an excellent test-bed for fuzzy control systems. Fuzzy controller, formulated on the basis of human understanding of the process or identified from measured control actions, can be regarded as an emulator of human operator. Controller design, however, may become difficult, especially if the number of state variables is large. In this paper, supervisory control system is proposed that reduces the complexity of the control problem and enhances control. This is demonstrated with backing simulations in comparison with other fuzzy control techniques.**

I. Introduction

Truck backer-upper problem, famous from the work of Nguyen and Widrow [1] has been investigated by many researchers. Being a highly nonlinear control problem, it has raised interest among the practitioners of computational intelligence. The advantage of the original Nguyen-Widrow approach is that the controller consisting of two neural networks is able to tune itself through a number of training epochs. In recent works, several authors e.g. [2] have complemented neural network with genetic algorithms. The basic shortcoming of all these data-driven methods is the computational cost.

On the other hand, it is hard not to notice that nearly anyone is able to drive the truck to the desired position given some time to adjust himself to the controls. It would be sort-sighted to ignore such knowledge and the possibility of fuzzy logic for utilizing it. Fuzzy controller that replaces human operator has been formulated on the basis of expert knowledge [3] or identified from control data [4]. Although the high computational load is avoided, the controller design procedure is ill-defined, plagued with the curse of dimensionality and may result in deteriorated performance.

In this paper fuzzy supervisory truck backer-upper control system is proposed. The proposed controller architecture allows the decomposition of the control task, thus relieving the problem with the curse of dimensionality. For our selection of state variables we also found the controller design task much easier than with the traditional selection. Last, but not least - our controller shows superior performance when compared with several other fuzzy controllers designed for truck backer-upper problem.

An overview of the control problem is given in section II. A selection of fuzzy logic based controllers, described in section IV, are tested using the MATLAB truck backer-upper demo. The comparison of controller performance (section V) is based on two estimates given in section III.

II. System definition

The truck as in [3], [4] corresponds to the cab part of the Nguyen-Widrow's truck and trailer, referred to as simplified Nguyen-Widrow problem. The truck position is determined by the three state variables x = [-20,20], y = [0, 25], and, Φ = [-90°, 270°] - the angle between truck's onward direction and the x-axis (Fig. 1). The width and length of the truck are 4 and 2 meters, respectively.

Truck must arrive from the initial position (x_0, y_0, Φ_0) to the loading dock (x_f = 0, y_f = 0) at a right angle (Φ = 90°). Truck only moves backward with the fixed speed. To control the truck at every stage appropriate steering angle θ = [-45°, 45°] must be provided. Thus controller is a function of state variables

$$\theta = f(x, y, \Phi) . \qquad (1)$$

Typically it is assumed that enough clearance between the truck and the loading dock exists so that the truck y-position coordinate y can be ignored, simplifying the controller function to:

$$\theta = f(x, \Phi) . \qquad (2)$$

For obvious reasons such controller does not perform very well if the distance between the truck position and the loading dock is small.

III. Evaluation of control performance

The primary control goal as stated in section 2, is the final state (0, 0, 90°). In practice, however, some tolerance (Fig. 2) should be allowed. Backing of the truck is considered successful if the following criterion (3) is met:

$$\varepsilon_c = \varepsilon_x + 0.0267\varepsilon_\Phi - 0.4 \le 0 , \qquad (3)$$

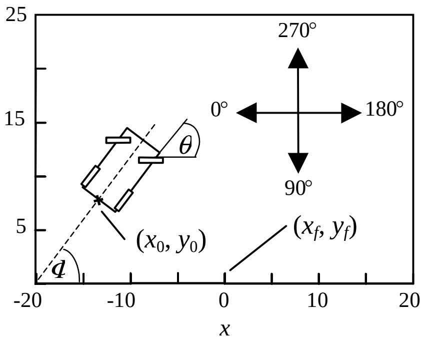

Fig. 1. Truck backer-upper system.

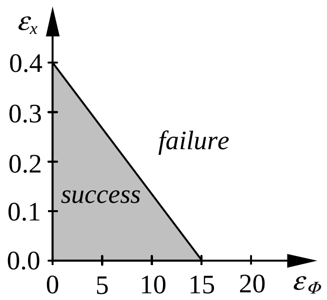

Fig. 2. Success/failure criterion.

where

$$\begin{cases} \varepsilon_x = \mathrm{abs}(x_f - x(T_f)) \\ \varepsilon_\Phi = \mathrm{abs}(\Phi_f - \Phi(T_f)) \end{cases}, \tag{4}$$

and where T_f is the duration of the backing.

Moreover, length and smoothness of the backing trajectory can be considered as a secondary criterion, expressed here by indirect estimate η

$$\eta = \frac{\int_0^{T_f} \sqrt{(dx/dt)^2 + (dy/dt)^2}\, dt}{\sqrt{(x_0 - x_f)^2 + (y_0 - y_f)^2}}, \tag{5}$$

where the distance the truck actually covers during the backing is divided by the shortest distance between the initial position and the position of the loading dock. Note that magnitude of η depends on the difference between Φ_0 and Φ_f and is always bigger than 1.

IV. Truck Backer-Upper Controllers

The basic fuzzy logic controllers are considered: controller based on expert knowledge and controller identified from human operator control actions, selection of state variables may vary. In addition, MATLAB state feedback controller and supervisory fuzzy controller are described.

A.. Expert-defined controller

Selection of the controller function (1) implies that the following rule base format should be used

IF x is A and y is B and Φ is C THEN θ is D, (6)

where A, B, C and D are the linguistic labels of the system variables associated with the corresponding fuzzy sets.

Design of the fuzzy controller includes the definition of input-output domains, partitions and fuzzy sets, and the contents of the rule base. The only source of that information in present case is human understanding of the driving process. The biggest problem with (6) is so-called curse of dimensionality. Employing the input partition {5 3 7}, for example, results in 105 rules that all must be determined manually. Although we can drive the car to the loading dock manually from almost any position, design of the fuzzy controller that would achieve the same goal is not a trivial task. Though fuzzy logic is so far the best of available artificial interfaces between humanlike reasoning and nonlinear world, it is certainly not the optimal one. Typically, the controller must be retuned and tested several times, sometimes it is necessary to add fuzzy sets if controller performance is poor, etc. In summary, whole design procedure is time-consuming and frustrating when the number of tuning parameters is large. We found the design task of (6) extremely difficult and therefore state variable selection (1) was replaced with (2). The resulting controller is quite similar to the one designed by Kong and Kosko [3]. Some readjustment of the controller parameters was necessary though, because the truck backing systems in [3] and in present case are not identical.

B. Data-driven modeling of human operator

The crucial problem with data-driven techniques is what kind of data must be prepared, how much data is required and how to collect it. In theory, we need a sufficient amount of data that would give good representation of operator actions. In practice, resources are always limited. The immediate problem in case of multidimensional systems is that some rules created in the initialization phase remain uncovered by data, implying that the rule base of the controller will be sparse. This may result in unexpected behavior. Also, the more data we have the longer is the learning process. Another data-driven modeling issue is that modeling algorithms available are not perfect, there always exists modeling error.

These days the favored choice seems to be ANFIS [5], although good properties of Gustafson-Kessel clustering in combination with least squares (GK/LS) procedure have been shown [6]. The basic difference between these approaches is that ANFIS is generally superior in what concerns approximation error, GK/LS is faster. The model obtained with ANFIS is essentially a black box, while the outcome of GK/LS modeling is transparent [7] (if properly configured) and allows interpretation. Because the latter difference is fundamental, we have applied both algorithms.

Data used in modeling was collected from 31 truck backing experiments with 8 upward, 6 leftward, 6 rightward and 11 downward initial angles. Starting positions were chosen so that different backing trajectories would be present (Fig. 3a). To reduce the computational load, most of data was filtered out so that the final data set, consisting of 642 input-output pairings, corresponds to the situation as if information had been available every third second only.

The number of parameters that influence the approximation error and must be determined prior to training is quite large and all of them cannot be determined automatically. Therefore, the determination of training parameters was based on trial and error until the configuration by what "reasonably low" approximation error was achieved was established. First, the necessity for modeling the control law (1) was confirmed because with (2), results of any acceptable accuracy could not be achieved. ANFIS was then applied to 1st order Takagi-Sugeno system with input partition of {7 3 9} and GK/LS model was initialized as a 0th order Takagi-Sugeno system with the same partition. Final modeling root-mean-squareerrors for ANFIS 2500 epochs, error 0.2129) and GK/LS (error 0.2048) are quite similar, though.

C. MATLAB state feedback controller

The state feedback controller that is included in MATLAB truck backer-upper demo employs fuzzy logic only to improve conventional control and was used here for comparison purposes only. The controller consists of two state feedback controllers (8,9) that operate on two different sets of state variables. The fuzzy logic block is then used as a blender to combine these two controllers smoothly, based on the distance between the truck and the loading dock.

D. Fuzzy supervisory controller

The control block in this case consists of fuzzy supervisor and PID controller (Fig. 4). The task of the supervisor is to provide setpoint Φ_r for the given state, appropriate steering angle is then determined by PID controller. Thus, in contrast to previous approaches, negative feedback is introduced. Although extra effort is required to determine the parameters of PID controller, it can be considered a bargain price for the exclusion of one state variable from the input of the fuzzy block. The rule base of the supervisor is easily configured, e.g. gray region in Fig. 5. that reads as

$$\text{IF } x \text{ is } mf4 \text{ AND } y \text{ is } mf3 \text{ THEN } \Phi_r \text{ is } 90°, \qquad (7)$$

is in good accordance with the idea what angle the truck in this particular area should maintain. The rest of the rules are derived using similar reasoning. Even simpler, single-input supervisor can be obtained by using only 7 rules that correspond to the subset of rules where "y is $mf3$", making the controller block equal to (2).

V. RESULTS

The comparison of controller performance was based on backing the truck from a number of randomly chosen initial positions. The feasibility of control was confirmed by manual backing (included in Figs. 5-6). The positions with what hitting the wall could not be avoided were filtered out, resulting in the final set of ten initial states N_e (Table I). The corresponding ε_c and η are depicted in Figs 5-6 and 7-8, respectively.

If y_0 is large enough (N_e = 3, 9, 10), all controllers enjoy success. In contrast, there are instances like N_e = 8, where all or N_e = 4, where most of them fail. Some controllers are clearly more accurate than others, their success rate ranging

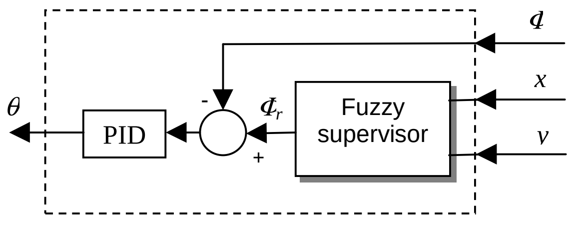

Fig. 3. Block diagram of fuzzy supervisory controller

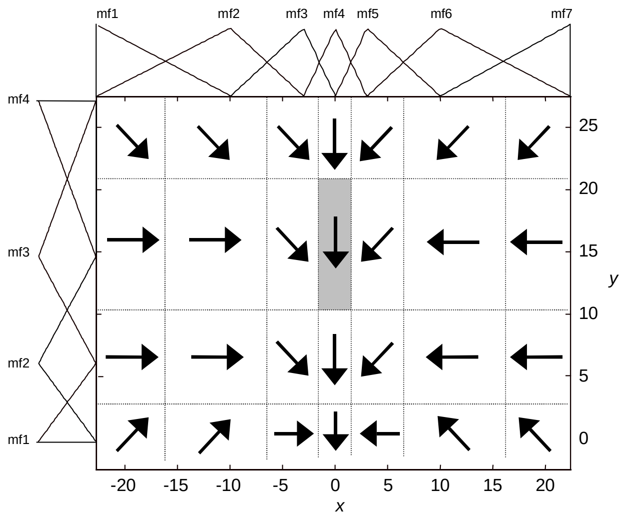

Fig. 4. Rule base of the two-input fuzzy supervisor.

from 30% (expert defined controller) to 90% (two-input supervisory controller). As could be expected, controllers that follow the control law (1), perform better than those that do not (expert defined controller, single-input supervisory controller) if y_0 is small.

TABLE I
INITIAL CONDITIONS

N_e	1	2	3	4	5	6	7	8	9	10
x_0	-3.7	-14.3	18.4	5.2	-14.9	12.1	-10.1	-0.9	-4.4	-11.9
y_0	18.0	8.5	17.7	7.3	5.0	9.1	7.0	9.1	16.2	18.6
Φ_0	-12.5	154.7	92.1	76.4	252.5	206.9	158.0	162.8	265.4	253.6

Of two data-driven controllers, the one obtained by GK/LS clearly stands out. Its performance is quite comparable with two-input supervisory controller. ANFIS-based controller, interestingly, shows unexpected behavior on some occasions as the truck "goes wandering" before returning to the loading dock (clearly expressed by η in Fig. 7 when N_e = 1 and N_e = 6). Because the approximation error of both algorithms was of similar magnitude, it can be assumed that the erratic behavior of ANFIS-based controller is at least in some degree caused by its non-transparent nature.

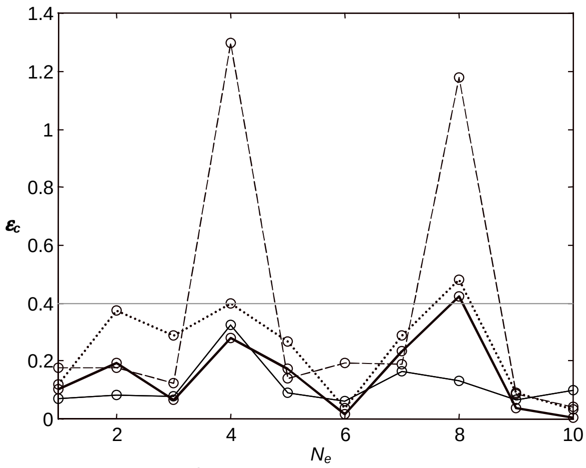

Fig. 5. ε_c : state feedback controller (dashed line), manual control (normal line), two-input supervisory controller (bold line), GK/LS-based controller (dotted line).

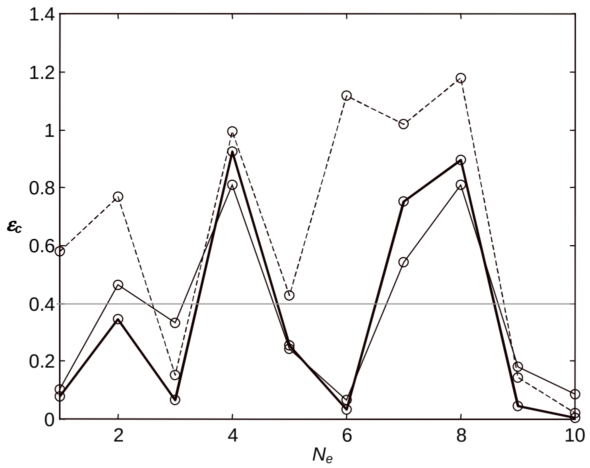

Fig. 6. ε_c : expert defined controller (dashed line), ANFIS-based controller (normal line), single-input supervisory controller (bold line).

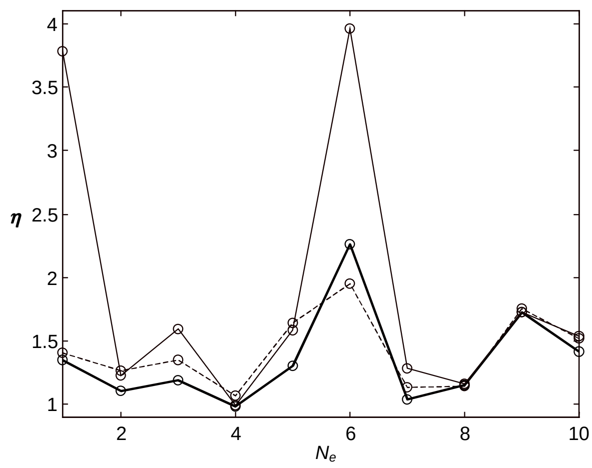

Fig. 8. η expert defined controller (dashed line), ANFIS-based controller (normal line), single-input supervisory controller (bold line).

We do not claim to have obtained optimal controllers in any of the cases. In fact, the universal approximation property of fuzzy systems allows us to assume that given enough time and/or computational power, controllers that easily outperform the current candidates, could be designed. Given the current state of things, however, time and complexity of controller design are of utmost importance and in these conditions, our proposed supervisory control has shown the best performance.

VI. Conclusions

The attempts to solve the truck backer-upper problem, rooted in computational intelligence, can be divided into two groups. The first group of methods seeks the solution through self tuning using neural networks, genetic algorithms or a combination of both.

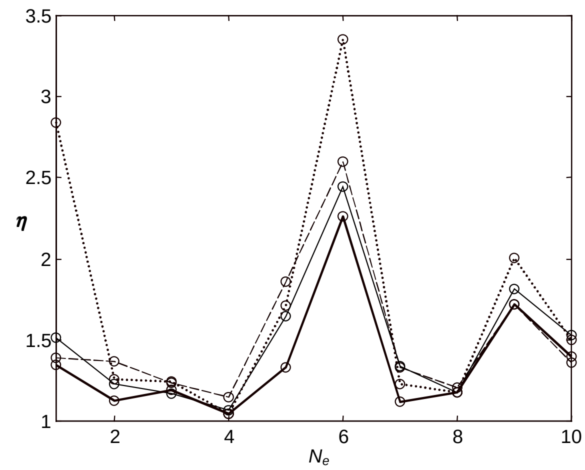

Fig. 7. η state feedback controller (dashed line), manual control (normal line), two-input supervisory controller (bold line), GK/LS-based controller (dotted line)

The second group of solutions, based on fuzzy logic, regards the controller as an emulator of human operator. The problems that plague these approaches are the computational complexity of the former and the poorly defined design procedure of the latter. Synthesis of conventional (PID) and fuzzy control has been proposed and applied in current paper. Compared to fuzzy controllers of same functionality, number of rules (and thus the number of parameters that has to be determined) is smaller by an order of magnitude, additionally, such "stripped" rule base has better compatibility with human driving experience. The PID controller in the control loop introduces negative feedback and stabilizes the control. In comparison with other fuzzy control systems the proposed fuzzy supervisory control system performs with satisfying control accuracy.

References

1. D. Nguyen and B. Widrow, "The truck backer-upper: An example of self-learning in neural network," *IEEE Contr. Syst. Mag.*, vol. 10, no. 2, pp. 18-23, 1990.
2. M. Schoenauer and E. Ronald, "Neuro-genetic truck backer-upper controller," *Proc. IEEE Conf. on Computational Intelligence*, pp. 720-723, 1994.
3. S.-G. Kong, B. Kosko, "Adaptive Fuzzy Systems for Backing up a Truck-and-Trailer," *IEEE Trans. on Neural Networks*, vol. 3, no. 5, pp. 211-223, 1992.
4. L.-X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Trans. on System, Man, and Cybernetics*, vol. 22, no. 6, pp. 1414-1427, 1992.
5. J.-S. R. Jang, *Neuro-Fuzzy Modeling: Architectures, Analyses, and Applications*, Ph.D. Dissertation, EECS Department, Univ. of California at Berkeley, 1992.
6. R. Babuska, *Fuzzy Modeling and Identification*, Ph.D. dissertation, Technical University of Delft, 1997.
7. A. Riid and E. Rüstern, "Transparent fuzzy systems and modeling with transparency protection," *Proc. IFAC Symp. on Artificial Intelligence in Real Time Control*, Budapest, pp. 229-234, 2000.