w

Tatra Mt. Math. Publ. **16** (1999), 325–338

TATRA MOUNTAINS
Mathematical Publications

APPLICATION OF AN APPROXIMATE FUZZY LOGIC CONTROLLER IN AN AGV STEERING SYSTEM, PATH TRACKING AND COLLISION AVOIDANCE STRATEGY

Szilveszter Kovács — László T. Kóczy

ABSTRACT. The application of the fuzzy interpolation-based approximate fuzzy reasoning methods in direct fuzzy logic control systems gives a simplified way for constructing the fuzzy rule base. The rule base of a fuzzy interpolation-based controller is not necessarily complete, it could contain the most significant fuzzy rules only without risking the chance of having no conclusion for some of the observations. In other words, during the construction of the fuzzy rule base, it is enough to concentrate on the main control actions, we do not have to bother with building a complete fuzzy rule base by adding "filling" rules (rules could be deduced from the others).

In this paper we would like to introduce an approximate fuzzy reasoning method based on K-H interpolation in the vague environment of the fuzzy rule base [2–4], which could be implemented to be simple enough for practical direct fuzzy logic control applications. For demonstrating the efficiency of the proposed approximate fuzzy reasoning method in direct fuzzy control, as a simulated complex practical application, a steering control of an automated guided vehicle (AGV) is also introduced. The main goal of the steering control is path tracking [6] (to follow a guide path) and to make the task more complex, the second one is a restricted (limited) collision avoidance. In our case, restricted collision avoidance means "avoiding obstacles without the risk of loosing the guide path". In this paper we would also like to introduce an approximate obstacle detection strategy based on measurements of three ultrasonic sensors and the move of the AGV; the fuzzy rule bases and their vague environments realising the path tracking and restricted collision avoidance strategy of the AGV (only 12 rules for control steering and 5 for the speed); and the simulated results of the system on a test path and obstacle configuration.

2000 Mathematics Subject Classification: 03B52.
Keywords: interpolate fuzzy reasoning, vague environment, path tracking and collision avoidance strategy.

This research was partly supported by the Hungarian National Scientific Research Fund grant no: F029904 and T019671, and the Hungarian Ministry of Culture and Education grant no 0422/1997.

1. Approximate fuzzy reasoning method based on K-H interpolation in the vague environment of the fuzzy rule base

Using the concept of vague environment described by scaling functions [1] instead of the linguistic terms of the fuzzy partition gives a simple way for fuzzy approximate reasoning.

The concept of vague environment is based on the similarity or indistinguishability of the elements. Two values in the vague environment are ε-distinguishable if their distance is grater than ε. The distances in vague environment are weighted distances. The weighting factor or function is called *scaling function* (factor) [1]. Two values in the vague environment X are ε-distinguishable if

$$\varepsilon < \delta_s(x_1, x_2) = \left| \int_{x_2}^{x_1} s(x)\, dx \right| ,$$

where $\delta_s(x_1, x_2)$ is the vague distance of the values x_1, x_2 and $s(x)$ is the scaling function on X.

For finding connections between fuzzy sets and a vague environment we can introduce the membership function $\mu_A(x)$ as a level of similarity $\boldsymbol{a}$ to x, as the degree to which x is indistinguishable from $\boldsymbol{a}$ [1]. The α-cuts of the fuzzy set $\mu_A(x)$ is the set which contains the elements that are $(1-\alpha)$-indistinguishable from $\boldsymbol{a}$ (see Fig. 1):

$$\delta_s(a,b) \le (1-\alpha)\,,\ \ \mu_A(x) = 1 - \min\{\delta_s(a,b), 1\} = 1 - \min\left\{ \left| \int_a^b s(x)\, dx \right|, 1 \right\}$$

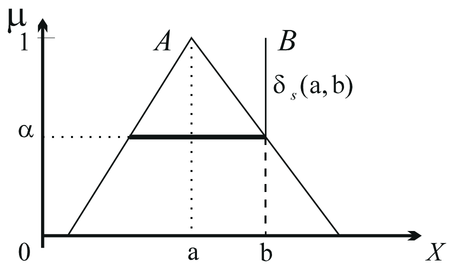

FIGURE 1. The α-cuts of $\mu_A(x)$ contains the elements that are $(1-\alpha)$-indistinguishable from $\boldsymbol{a}$

It is very easy to realise (see Fig. 1.), that in this case the vague distance of points a and b ($\delta_s(\boldsymbol{a}, \boldsymbol{b})$) is basically the *Disconsistency Measure* (S_D) of the fuzzy sets A and B (where B is a singleton):

$$S_D = 1 - \sup_{x \in X} \mu_{A \cap B}(x) = \delta_s(\boldsymbol{a}, \boldsymbol{b}) \qquad \text{if} \quad \delta_s(\boldsymbol{a}, \boldsymbol{b}) \in [0, 1]\,,$$

where $A \cap B$ is the min t-norm, $\mu_{A \cap B}(x) = \min\big[\mu_A(x)\,,\, \mu_B(x)\big] \;\; \forall x \in X$.

It means that we can calculate the disconsistency measures between member fuzzy sets of a fuzzy partition and a singleton, as vague distances of points in the vague environment of the fuzzy partition. The main difference between the disconsistency measure and the vague distance is that the vague distance is a crisp value in range of $[0, \infty]$, while the disconsistency measure is limited to $[0, 1]$. We could use the vague distance as an "extended" disconsistency measure, that is why they are useful in interpolate reasoning with insufficient evidence.

So if it is possible to describe all the fuzzy partitions of the primary fuzzy sets (the antecedent and consequent universes) of our fuzzy rule base by vague environments, and the observation is a singleton, we can calculate the "extended" disconsistency measures of the antecedent primary fuzzy sets of the rule base and the observation, and the "extended" disconsistency measures of the consequent primary fuzzy sets and the consequence (we are looking for) as vague distances of points in the antecedent and consequent vague environments.

The vague environment is described by its scaling function. For generating a vague environment of a fuzzy partition we have to find an appropriate scaling function, which describes the shapes of all terms in the fuzzy partition. A fuzzy partition can be characterized by a single vague environment if and only if the membership functions of the terms fulfils the following requirement [1]:

$$s(x) = \big|\, \mu(x) \,\big| = \left| \frac{d\mu}{dx} \right| \;\; \text{exists iff} \;\; \min\big\{\mu_i(x)\,,\, \mu_j(x)\big\} > 0 \Rightarrow$$
$$\big|\, \mu_i(x) \,\big| = \big|\, \mu_j(x) \,\big| \quad \forall i, j \in I\,,$$

where $s(x)$ is the vague environment we are looking for.

Generally the above condition is not fulfilled, so the question is how to describe all fuzzy sets of the fuzzy partition with one "universal" scaling function. For this reason we propose to use the *approximate scaling function* [2], [3], [4]. The approximate scaling function is an approximation of the scaling functions describing the terms of the fuzzy partition separately.

If the vague environment of a fuzzy partition (the scaling function or the approximate scaling function) exists, the member fuzzy sets of the fuzzy partition can be characterized by points in the vague environment. (In our case the points are characterizing the cores of the terms, while the shapes of the membership functions are described by the scaling function.) If all the vague environments of the antecedent and consequent universes of the fuzzy rule base exist, all the primary fuzzy sets (linguistic terms) used in the fuzzy rule base can be characterized by points in their vague environment. So the fuzzy rules (build on the primary fuzzy sets) can be characterized by points in the vague environment of the fuzzy rule base, too. In this case the approximate fuzzy reasoning can be handled as a classical interpolation task. Applying the concept of vague environment (the distances of points are weighted distances), any interpolation, extrapolation or regression methods can be adapted very simply for approximate fuzzy reasoning [2], [3], [4].

We suggest to adapt the *Kóczy-Hirota interpolation* [5]. This method generates the conclusion as a weighted sum of the vague consequent values, where the weighting factors are inversely proportional to the vague distances of the observation and the corresponding rule antecedents:

$$\operatorname{dist}(y_0, y) = \frac{\sum_{k=1}^{r} w_k \cdot \operatorname{dist}(y_0, b_k)}{\sum_{k=1}^{r} w_k}, \qquad w_k = \frac{1}{\left(\operatorname{dist}(x, a_k)\right)^p}, \tag{1.1}$$

where w_k is a weighting factor inversely proportional to the vague distance of the observation and the kth rule antecedent,

$$\operatorname{dist}(a_k, x) = \operatorname{dist}(x, a_k) = \sqrt{\sum_{i=1}^{m} \left(\int_{a_{k,i}}^{x_i} s_{X_i}(x_i)\, dx_i \right)^2},$$

$$\operatorname{dist}(y_0, b_k) = \int_{y_0}^{b_k} s_Y(y)\, dy\,,$$

where s_{X_i} is the ith scaling function of the m dimensional antecedent universe, s_Y is the scaling function of the one dimensional consequent universe, x is the multidimensional crisp observation, a_k are the cores of the multidimensional fuzzy rule antecedents A_k, b_k are the cores of the one dimensional fuzzy rule consequents B_k, $R_i = A_i \to B_i$ are the fuzzy rules, p is the sensitivity of the weighting factor for distant rules, y_0 is the first element of the one dimensional universe ($Y\colon y_0 \le y,\ \forall y \in Y$), y is the one dimensional conclusion we are looking for.

A simple one dimensional example of the approximate scaling function and the Kóczy-Hirota interpolation (1.1) is introduced in Figure 2. For the comparison the crisp conclusions of the K-H interpolation and the classical methods the conclusions generated by the max-min CRI and the center of gravity defuzzification for the same rulebase are also demonstrated in Figure 2. A more detailed description of the proposed approximate fuzzy reasoning method can be found in [2], [3], [4].

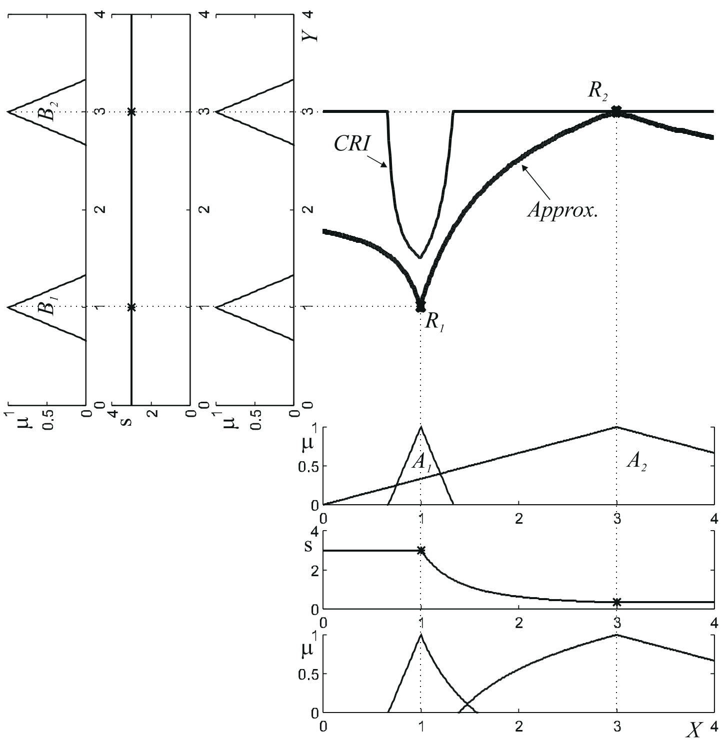

FIGURE 2. Interpolation of two fuzzy rules (R_i: $A_i \to B_i$) in the approximated vague environment of the fuzzy rulebase, using the K-H interpolation ($p = 1$) (Approx.) and the min-max. CRI with the centre of gravity defuzzification (CRI). μ is the membership grade, and s is the scaling factor.

As an example of the practical application of the proposed approximate fuzzy reasoning method, a real path tracking and restricted collision avoidance control strategy for differential steered AGVs (Automated Guided Vehicle) [6] is introduced.

2. The guide path controlled AGV

The Automatically Guided Vehicle (AGV) is a typical element of the group of materials handling equipment. A popular way of AGV guidance is based on the guide path method. The guide path is usually a painted marking or a passive or active wire (guidewire) glued onto or built into the floor. The goal of the steering part of the guidance system of the AGV is to follow the marking of the guide path. The guiding system senses the position of the guide path by special sensors (*guide zone*) tuned for the guide path. The *guide zone* is a section of the AGV determined by the guide path sensor (or raw of sensors). The goal of the steering control is to follow the guide path by the guide zone with a minimal path tracking error on the whole path (e.g., Fig. 3).

The kinematic of the AGV is determined by its wheel configuration. The AGVs without fixed directional wheels in their wheel configuration can be moved to arbitrary direction. While the AGVs with at least one fixed directional wheel can run only on a path curve has its momentary centre on the line, and fits the axe of the fixed directional wheel. In the further part of this article we would like to concentrate on the path tracking strategy of a differential steered AGV which has fixed directional wheel (e.g., Fig. 3).

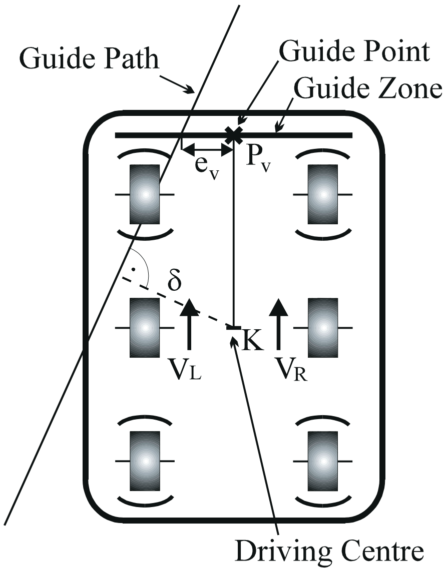

FIGURE 3. Differential steered AGV with guide zone, δ is the path tracking error, e_v is the distance of the guide path and the guide point, P_v is the guide point and K is the driving centre. V_L and V_R are the contour speeds of the left and right wheels (differential steering).

3 The Path tracking and the restricted collision avoidance strategy

The main goal of the steering control is path tracking (to follow a guide path) [6]. To make the example task more complex, we added a second goal as a restricted (limited) collision avoidance. In our case, restricted collision avoidance means "avoiding obstacles without the risk of loosing the guide path".

The simplest way of defining these strategies is based on describing the operator's control actions. These control actions could form the fuzzy rule base.

In our case — using the previously introduced approximate reasoning method for direct fuzzy control — constructing the fuzzy rule base is very simple. We do not have to bother with building a complete fuzzy rule base, it is enough to concentrate on the main control actions, by simply adding rules piece by piece. Having the simulated model of the controlled system, we can check the performance of the controller after each step.

This kind of design (modify and test) could be very useful in case of controlling unknown, or partly known systems.

The control of a differential steered AGV is very similar to the control of a car. The base idea of the path tracking strategy is very simple: keep the driving centre of the AGV as close as possible to the guide path, then if the driving centre is close enough to the guide path, simply turn the AGV into the docking direction. For defining this part of the strategy, we have to examine the observations we need for the guidance system. The above simple strategy needs only two observations: The distance between the guide path and the driving centre (path tracking error), and the distance between the guide path and the guide point. Using the guide zone, we can determine the distance of the guide path and the guide point, but we have no information on the path tracking error. We suggest to calculate the estimated momentary path tracking error (δ) from the previous (e_{vo}) and the current value (e_v) of the distance between the guide path and the guide point (measured by the guide path) and from the move of the AGV [6]. The movement of the AGV can be measured by incremental move sensors linked to the driven wheels. This method calculates the path tracking error as if the guide path was straight (see Fig. 4). So for arbitrary shaped guide path it is only an estimated value. The lay-by docking points can be placed only on straight path (we would like to minimize the minimal docking distance), so the estimation we are using seems to be good enough in our situation.

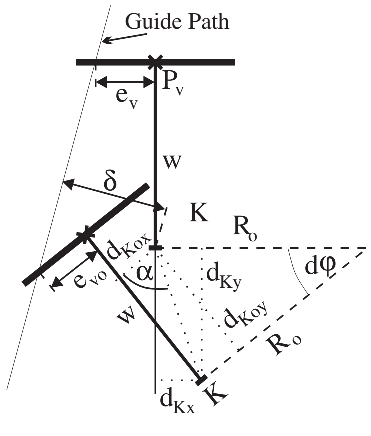

FIGURE 4. Calculation of the estimated momentary path tracking error,

The estimated momentary path tracking error can be calculated by the following forms (see Fig. 4):

$$
\begin{aligned}
&d\varphi = \frac{s_L - s_R}{d}\,,\\
&R_o = d \cdot \frac{s_L + s_R}{2\cdot(s_L - s_R)}\,,\\
&dK_{ox} = R_o \cdot \bigl(1-\cos(d\varphi)\bigr)\,,\\
&dK_{oy} = R_o \cdot \sin(d\varphi)\,,\\
&\alpha = \arctan\left(\frac{dK_{oy}}{dK_{ox}}\right),\\
&dK = \sqrt{dK_{ox}^2 + dK_{oy}^2}\,,\\
&dK_x = -dK \cdot \cos(\alpha + d\varphi)\,,\\
&dK_y = -dK \cdot \sin(\alpha + d\varphi)\,,\\
&e_{vox} = -dK_x + w\cdot\sin(d\varphi) + e_{vo}\cdot\cos(d\varphi)\,,\\
&e_{voy} = -dK_y + w\cdot\cos(d\varphi) - e_{vo}\cdot\sin(d\varphi)\,,\\
&\delta_m = e_v - w\cdot\left(\frac{e_v - e_{vox}}{w - e_{voy}}\right),\\
&\xi = \arctan\left(\frac{e_v - e_{vox}}{w - e_{voy}}\right),\\
&\delta = \delta_m \cdot \cos(\xi)\,,
\end{aligned}
$$

where e_v is the distance between the guide path and the guide point (from the guide zone), e_{vo} is the previous value of e_v; s_R, s_L are the move of the AGV measured on the right and the left wheel, w is the distance of the guide point and the driving centre, d is distance of the two wheels, d_s is the width of the guide zone, and δ is the estimated momentary path tracking error we are looking for.

For defining the restricted collision avoidance strategy we have to study the types of the possible collision situations. There are two different collision situations, the frontal and the side collision. We need the simplest obstacle sensor configuration giving enough information for both the avoidable situations. Having the preconditions of motionless and avoidable obstacles, it is sufficient to have three ultrasonic distance sensors on the front of the AGV, one in the middle (U_M) and one-one on both sides (U_L, U_R) (see Fig. 5).

The three distances (R_L, R_R, R_M), measured by the three obstacle sensors (U_L, U_R, U_M) give a sufficient information for finding a strategy to be able to avoid the frontal collision situations.

The sufficiency of the measurements of these sensors for generating observations for avoiding the side collisions is not so trivial. Having the preconditions of

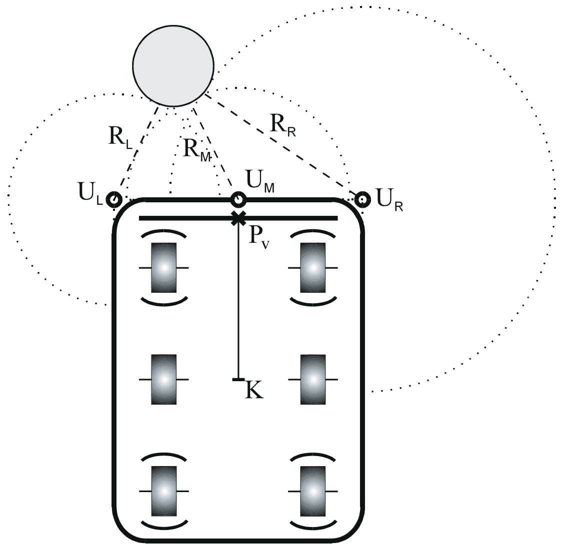

FIGURE 5. R_L, R_R, R_M are the distances measured by the left, right and middle ultrasonic distance sensors U_L, U_R and U_M.

motionless and avoidable obstacles, we have a chance to use the obstacle distance measurements of the near past for scanning the boundaries of the obstacles. Collecting the previous measurements of the left and right obstacle sensors and the corresponding positions of the AGV (measured by the motion sensors on the wheels), we can approximate the boundaries of the obstacles by discrete points. We call these points *unsafe*, or *risky points*. The distance measured by an obstacle sensor means the existence of a potential obstacle outside the circle defined by the position of the sensor and the measured value (see, e.g., Fig. 6). Having more measurements and more positions we can approximate the boundaries of the obstacles by the pair by pair point of intersection of these circles. We are simply collecting the point of intersection of the previous and the actual circles. The intersections of the two circles are two points. We can choose the real one by checking whether one of them is covered by the body of the AGV (see, e.g., on Fig. 6). Were both points situated on uncovered positions, we can choose the point situated farther from the longitudinal axe of the AGV to be the real one (based on the precondition of avoidable obstacles).

The main idea of the side collision avoidance strategy is to avoid side collisions to obstacles by avoiding side collisions to unsafe points. For having observations easier to handle than unsafe points, we calculate the actual maximal left and right turning angle without side collision (α_{ML}, α_{MR}) (see, e.g., on Fig. 7).

The consequences (output variables) of the rules we are searching for are the value of the speed (V_a) and the steering (V_d). The AGV we studied has a differential steering, so the speed (V_a) and the steering (V_d) can be calculated

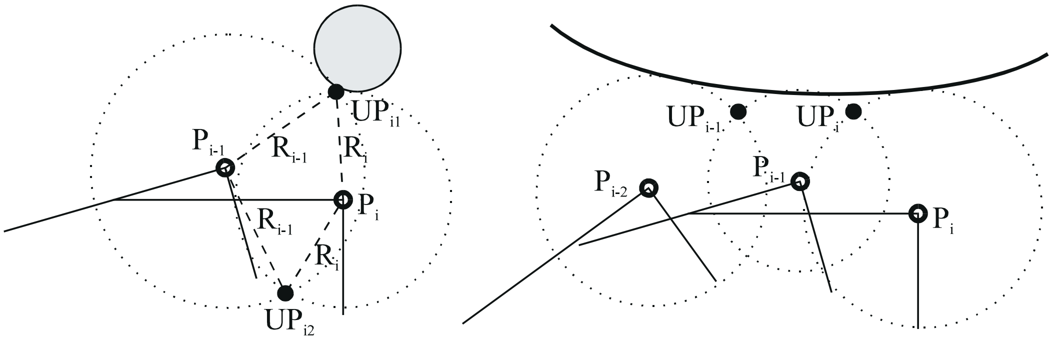

FIGURE 6. The obstacles boundaries approximated by discrete unsafe points, where R is the distance measured by the ultrasonic sensor P and UP is the unsafe (risky) point.

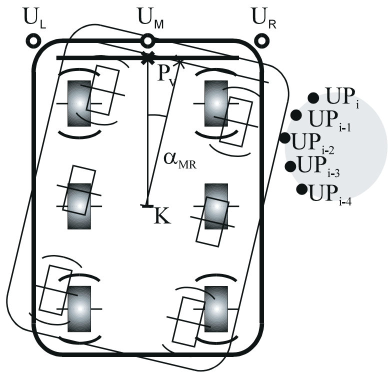

FIGURE 7. α_{MR} is the maximal right turning angle without side collision

as follows:

$$V_d = V_L - V_R\,,$$
$$V_a = \frac{V_L + V_R}{2}\,,$$

where V_L, V_R are the contour speeds of the left and right wheels, respectively.

Let us collect the rules characterizing our path tracking and restricted (limited) collision avoidance strategy by describing the momentary manoeuvres (*speed* (V_a), *steering* (V_d)) needed for path tracking, frontal and side collision avoidance in some significant positions of the AGV. These positions are characterized by the observations: by the *distance of the guide path and the guide point* (e_v), the *estimated path tracking error* (δ), the *distances measured by the left middle and right ultrasonic sensors* (R_L, R_R, R_M) and the *approximated maximal left and right turning angle without side collision* (α_{ML}, α_{MR}).

Having two conclusions, the *speed* (V_a) and the *steering* (V_d), we have two rule bases. One for the *steering* R_{Vd} and one for the *speed* R_{va} of the AGV.

The ith rules of these rule bases have the following forms:

$R_{Vd,i}$:

$$\textbf{if } e_v = A_{1,i} \textbf{ and } \delta = A_{2,i} \textbf{ and } R_L = A_{3,i} \textbf{ and} R_R = A_{4,i} \textbf{ and } R_M = A_{5,i} \textbf{ and } \alpha_{ML} = A_{6,i} \textbf{ and } \alpha_{MR} = A_{7,i} \textbf{ then } V_d = B_i .$$

$R_{Va,i}$:

$$\textbf{if } e_v = A_{1,i} \textbf{ and } \delta = A_{2,i} \textbf{ and } R_L = A_{3,i} \textbf{ and } R_R = A_{4,i} \textbf{ and } R_M = A_{5,i} \textbf{ then } V_a = B_i .$$

Having a simulated model of the AGV and a trial guide path, we have got only 12 rules for controlling the steering and 5 for the speed:

R_{Vd}:	e_v	δ	R_L	R_R	R_M	M_L	M_R	V_d
1.	NL							PL
2.	PL							NL
3.	NM	Z					L	PL
4.	PM	Z				L		NL
5.	NM	PM	L		L	L		Z
6.	PM	NM		L	L		L	Z
7.	Z	PM	L		L	L		NS
8.	Z	NM		L	L		L	PS
9.	Z	PM	S		S			PL
10.	Z	NM		S	S			NL
11.	Z	Z	L	S	S			NL
12.	Z	Z	S	L	S			PL

R_{Va}:	e_v	δ	R_L	R_R	R_M	V_a
1.	Z	Z	L	L	L	L
2.	NL	PL				Z
3.	PL	NL				Z
4.	NL	Z				Z
5.	PL	Z				Z

where N: negative, P: positive, L: large, M: middle, Z: zero.

Having the rule base, the next step of building the fuzzy controller based on interpolation in the vague environment of the fuzzy rule base is to generate the vague environments of the antecedent and consequent universes (Fig. 8). We have generated these vague environments (scaling functions) by a tuning process based on values fetched from expert's knowledge. The tuning process was optimized for the core positions and the scaling factor values of the linguistic terms for getting the shortest docking distance on the trial guide path (Fig. 9).

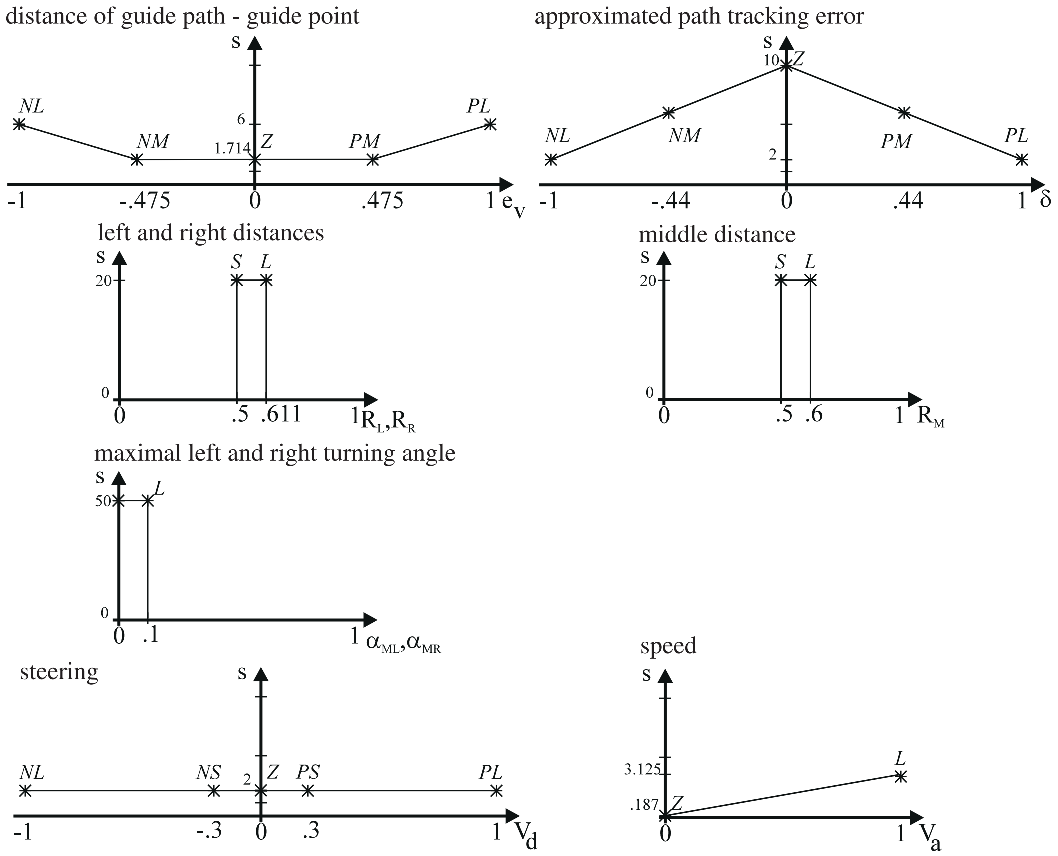

FIGURE 8. Vague environments (scaling functions) of the antecedent and consequent universes.

4. Conclusions

The approximate fuzzy reasoning method based on K-H interpolation in the vague environment of the fuzzy rule base gives an efficient way for designing direct fuzzy logic control applications.

The example introduced in this paper, the simulated implementation of the path tracking and restricted collision avoidance strategy of an automated guided

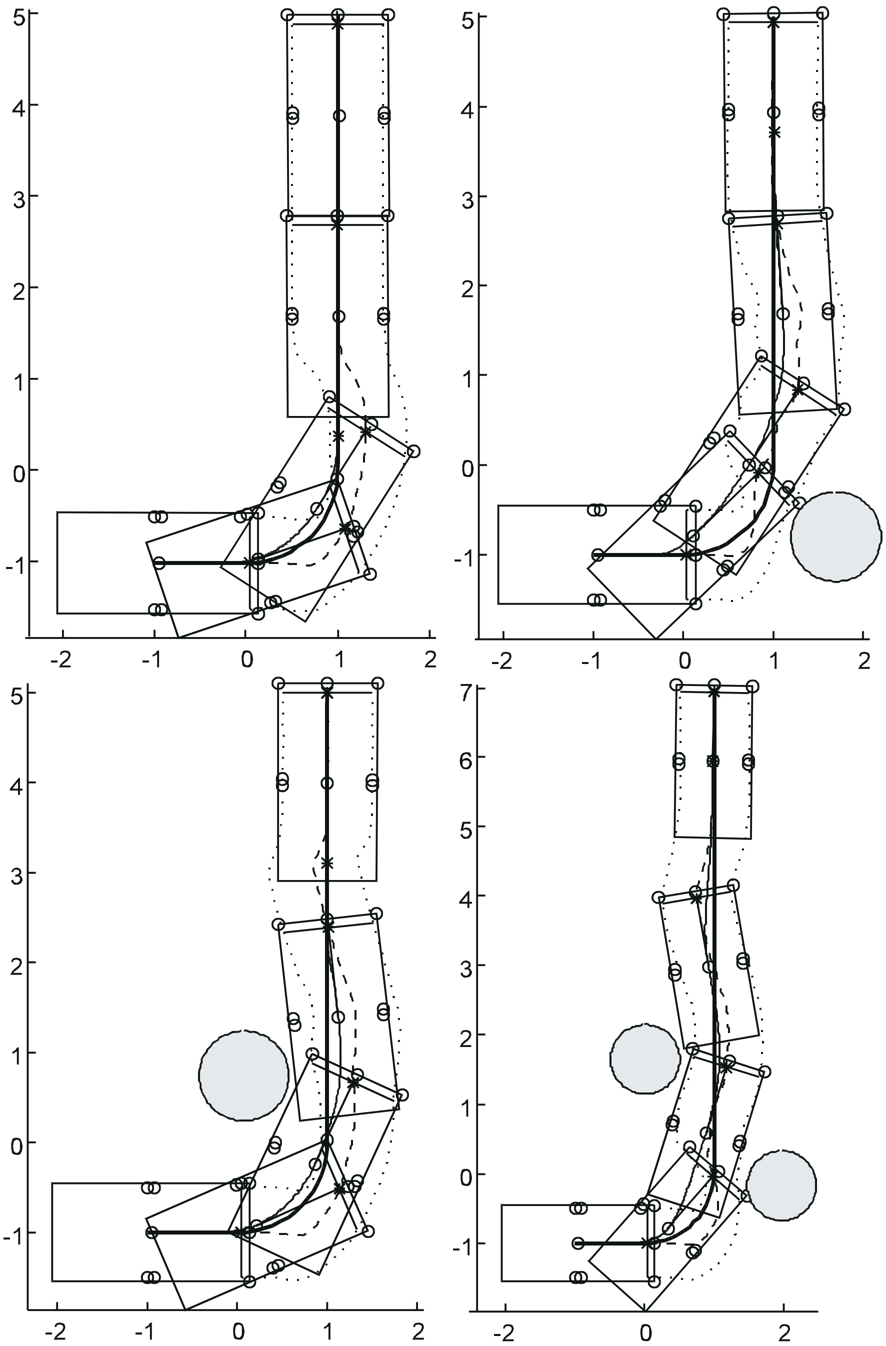

FIGURE 9. Some simulated results of the AGV using the interpolate fuzzy reasoning based FLC path tracking and restricted collision avoidance strategy on a trial guide path

vehicle (AGV), demonstrates the simplicity of collecting the fuzzy rules in a rather complex application (in spite of having 7 observations, the rule base of the steering contains only 12 rules). We do not have to bother with building a complete fuzzy rule base, it is enough to concentrate on the main control actions, by simply adding rules piece by piece. Having the simulated model of the controlled system, we can check the performance of the controller after each step.

This kind of design (modify and test) could be very useful in case of controlling unknown or partly known systems.

REFERENCES

[1] KLAWONN, F.: *Fuzzy sets and vague environments*, Fuzzy Sets and Systems **66** (1994), 207–221.

[2] KOVÁCS, SZ.: *New aspects of interpolative reasoning*, in: Proceedings of the 6th. International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Granada, Spain, 1996, pp. 477–482.

[3] KOVÁCS, SZ.—KÓCZY, L. T.: *Approximate fuzzy reasoning based on interpolation in the vague environment of the fuzzy rule base as a practical alternative of the classical CRI*, in: Proceedings of the 7th International Fuzzy Systems Association World Congress, Prague, 1997, pp. 144–149.

[4] KOVÁCS, SZ.—KÓCZY, L. T.: *The use of the concept of vague environment in approximate fuzzy reasoning, Fuzzy Set Theory and Applications*, Tatra Mt. Math. Publ. **12** (1997), 169–181.

[5] JOÓ, I.—KÓCZY, L. T.—TIKK, D.—VÁRLAKI, P.: *On a stable interpolation method*, in: Proceedings of the 7th International Fuzzy Systems Association World Congress, Prague, 1997, pp. 133–137.

[6] KOVÁCS, SZ.—KÓCZY, L. T.: *Application of the approximate fuzzy reasoning based on interpolation in the vague environment of the fuzzy rule base in the fuzzy logic controlled path tracking strategy of differential steered* AGVs*, Proceedings of the International Conference, 5th Fuzzy Days, Dortmund*, in: Lecture Notes in Comput. Sci., Vol. 1226, Springer-Verlag, Berlin, 1997, pp. 456–467.

Received May 6, 1998

Department of Information Technology
University of Miskolc
Miskolc-Egyetemváros
H–3515, Miskolc
HUNGARY

E-mail: szkszilv@gold.uni-miskolc.hu

Department of Telecommunication
and Telematics
Technical University of Budapest
Sztoczek u. 2
H–1111, Budapest
HUNGARY

E-mail: koczy@ttt-202.ttt.bme.hu