Fuzzy Control for Knowledge-Based Interpolation

Christian Moewes and Rudolf Kruse

Abstract. Fuzzy control accounts for the biggest industrial success of fuzzy logic. We review an interpretation of Mamdani's heuristic control approach. It can be seen as knowledge-based interpolation based on input-output points of a vaguely known function. We reexamine two real-world control problems that have been fortunately solved based on this interpretation.

1 Introduction

The biggest success of fuzzy logic in the field of industrial and commercial applications has been achieved with *fuzzy controllers*. It has been developed by Ebrahim "Abe" Mamdani and his student Sedrak Assilian in 1975 [12]. Fuzzy control is a way of defining a nonlinear table-based controller whereas its nonlinear transition function can be defined without specifying every single entry of the table individually. Many real-world problems have been tackled successfully by Mamdani's fuzzy control approach.

But what exactly is the justification of this heuristic approach? This question aroused the interests of many researchers [1, 2, 4, 5, 6, 7, 8, 14]. In our opinion, Mamdani's approach can be seen as knowledge-based interpolation. It is a kind of approximate reasoning using fuzzy set theory. Given some input-output points (*i.e.* some knowledge) of a vaguely known function, an approximate output for a new input point can be deduced by interpolation. Here, the gradual nature of fuzzy sets is very helpful to model similarity between given inputs points and unknown ones. This view has been justified during the 1990s.

Christian Moewes · Rudolf Kruse
Faculty of Computer Science, Otto-von-Guericke University of Magdeburg, Germany
e-mail: cmoewes@ovgu.de, kruse@iws.cs.uni-magdeburg.de

E. Trillas et al. (Eds.): Combining Experimentation and Theory, STUDFUZZ 271, pp. 91–101.
springerlink.com

Mamdani and Assilian developed their idea application-driven to control a steam engine based on human expert knowledge. Our justification has been formulated in a similar way with the industrial partner Volkswagen AG (VW) in Wolfsburg, Germany. In the beginning of the 1990s, VW engineers and managers were skeptical towards fuzzy control whether it is actually worth something.

Shortly after in 1993, the first workshop "Fuzzy Systems – Management of Uncertain Information" took place in Braunschweig, Germany. About 120 participants attended the workshop including VW engineers. The chairman was the second author — Abe Mamdani and Didier Dubois were invited speakers. The results of the workshop have been published in an English written monograph [10]. One year later in 1995, after the skepticism was gone the first fuzzy controller at VW went into production.

2 Fuzzy Control

Suppose we consider a technical system. For this system, we dictate a desired behavior. Generally a time-dependent *output variable* must reach a desired set value. The output is influenced by a *control variable* which we can manipulate. Finally, there exists a time-dependent *disturbance variable* that influences the output as well. The current control value is usually determined based on the current measurement values of the output variable ξ, the variation of the output $\Delta\xi = \frac{d\xi}{dt}$ and further variables.

Hereafter we will refer to input variables $\xi_1 \in X_1, \ldots, \xi_n \in X_n$ and one control variable $\eta \in Y$. The solution of a control problem is a suitable control function $\varphi : X_1 \times \ldots \times X_n \to Y$ that determines an appropriate control value $y = \varphi(\mathbf{x})$ for every input tuple $\mathbf{x} = (x^{(1)}, x^{(2)}, \ldots, x^{(n)}) \in X_1 \times \ldots \times X_n$. In classical control engineering, φ is commonly determined by solving a set of differential equations. It is very often out of the question to specify an exact set of differential equations. Note that human beings, however, are greatly able to control certain processes without knowing about higher mathematics.

Simulating the behavior of a human "controller" can be done by questioning the individual directly. An alternative would be extract essential information by observing the controlled process. The result of such *knowledge-based analysis* is a set of *linguistic rules* that control the process. Linguistic rules comprise a premise and a conclusion. The former relates to a fuzzy description of the crisp measured input, where the latter defines a suitable fuzzy output. Thus we need to formalize mathematical descriptions of the linguistic expressions used in the rules. Furthermore initialized rules need to be accumulated to result in one fuzzy output value. Finally, a crisp output value must be computed from the fuzzy one. The whole architecture for that knowledge-based model of a fuzzy controller is shown in Fig. 1.

The *fuzzification interface* operates on the current input value $\mathbf{x}_0$. If needed, $\mathbf{x}_0$ is mapped into a suitable domain, *e.g.* normalization to the unit interval. It also

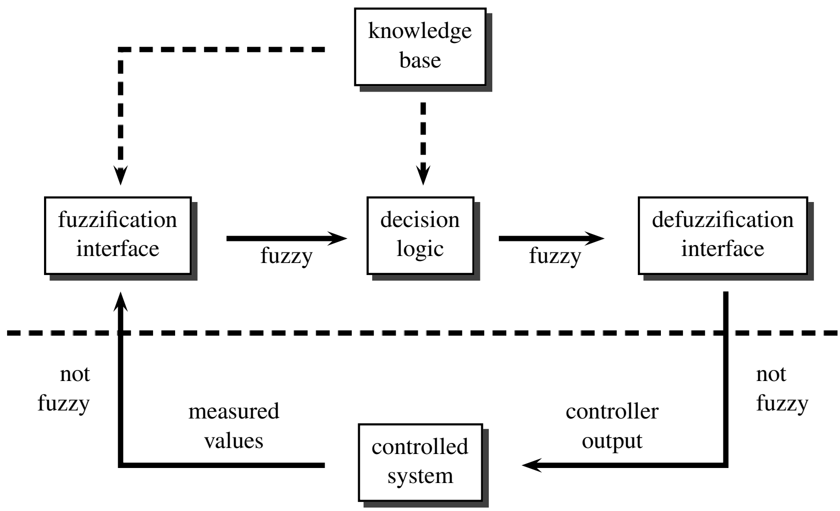

Fig. 1 Architecture of a fuzzy controller.

transforms $\mathbf{x}_0$ into a linguistic term or fuzzy set. The *knowledge base* comprises the *data base*, *i.e.* all pieces of information about variable ranges, domain transformations, and the fuzzy sets with their corresponding linguistic terms. Moreover, it also contains a *rule base* storing the linguistic rules for controlling. The *decision logic* determines the output value of the corresponding measured input using the knowledge base. The *defuzzification interface* produces the crisp output value given the fuzzy output.

3 What Exactly Is Mamdani Control?

In 1975, the first model of a fuzzy controller was created by Mamdani and Assilian [12]. Here, the knowledge of an expert must be expressed by linguistic rules. First, for the set X_1, p_1 fuzzy sets $\mu_1^{(1)}, \ldots, \mu_{p_1}^{(1)} \in \mathscr{F}(X_1)$ must be defined. Accordingly, each fuzzy set is named with a suitable linguistic term. Second, X_1 is partitioned by its fuzzy sets. To be able to interpret each fuzzy set as fuzzy value or fuzzy interval, it is favorable to only use unimodal membership functions. Also, fuzzy sets of one partition should be disjoint, *i.e.* they satisfy

$$i \neq j \Rightarrow \sup_{x \in X_1} \left\{ \min \left\{ \mu_i^{(1)}(x), \mu_j^{(1)}(x) \right\} \right\} \leq 0.5.$$

Having divided X_1 into p_1 fuzzy sets $\mu_1^{(1)}, \ldots, \mu_{p_1}^{(1)}$, we partition the remaining sets $X_2, \ldots, X_n$ and Y in the same manner. Finally, these fuzzy partitions and the linguistic terms associated with the fuzzy sets correspond to the data base in our knowledge base.

The rule base is specified by rules of the form

$$\textbf{if } \xi_1 \textbf{ is } A^{(1)} \textbf{ and } \ldots \textbf{ and } \xi_n \textbf{ is } A^{(n)} \textbf{ then } \eta \textbf{ is } B \tag{1}$$

whereas $A^{(1)},\ldots,A^{(n)}$ and B represent linguistic terms corresponding to fuzzy sets $\mu^{(1)},\ldots,\mu^{(n)}$ and μ, respectively, according to fuzzy partitions of $X_1 \times \ldots \times X_n$ and Y. Hence the rule base comprises k control rules

$$R_r : \textbf{if } \xi_1 \textbf{ is } A^{(1)}_{i_{1,r}} \textbf{ and } \ldots \textbf{ and } \xi_n \textbf{ is } A^{(n)}_{i_{n,r}} \textbf{ then } \eta \textbf{ is } B_{i_r}, \quad r = 1,\ldots,k.$$

Remark that these rules are not regarded as logical implications. They rather define $\eta = \varphi(\xi_1,\ldots,\xi_n)$ piecewise where

$$\eta \approx \begin{cases} B_{i_1} & \text{if } \xi_1 \approx A^{(1)}_{i_{1,1}} \text{ and } \ldots \text{ and } \xi_n \approx A^{(n)}_{i_{n,1}}, \\ \vdots & \vdots \\ B_{i_k} & \text{if } \xi_1 \approx A^{(1)}_{i_{1,k}} \text{ and } \ldots \text{ and } \xi_n \approx A^{(n)}_{i_{n,k}}. \end{cases}$$

Since the rules are treated as *disjunctive*, we can say that the control function φ is obtained by knowledge-based interpolation.

Observing a measurement $\mathbf{x} \in X_1 \times \ldots \times X_n$ the decision logic applies each R_r separately. It computes the degree to which $\mathbf{x}$ fulfills the premise of R_r, *i.e.* the degree of applicability

$$\alpha_r \stackrel{\text{def}}{=} \min\left\{\mu^{(1)}_{i_{1,r}}(x^{(1)}),\ldots,\mu^{(n)}_{i_{n,r}}(x^{(n)})\right\}. \tag{2}$$

“Cutting off” the output fuzzy set μ_{i_r} of rule R_r at α_r leads to the rule’s output fuzzy set:

$$\mu_{\mathbf{x}}^{\mathrm{o}(R_r)}(y) = \min\{\alpha_r, \mu_{i_r}(y)\}. \tag{3}$$

Having computed all α_r for $r = 1,\ldots,k$, the decision logic combines all $\mu_{\mathbf{x}}^{\mathrm{o}(R_r)}$ applying the t-conorm maximum in order to get the overall output fuzzy set

$$\mu_{\mathbf{x}}^{\mathrm{o}}(y) = \max_{r=1,\ldots,k}\{\min\{\alpha_r, \mu_{i_r}(y)\}\}. \tag{4}$$

In control engineering, a crisp control value is needed. Therefore $\mu_{\mathbf{x}}^{\mathrm{o}}$ is forwarded to the defuzzification interface. Here, it depends on the kind of method that is implemented to defuzzify $\mu_{\mathbf{x}}^{\mathrm{o}}$. The most well-known approaches are the max criterion method, the mean of maxima (MOM) method and the center of gravity (COG) method. Using the first approach, simply an arbitrary value $y \in Y$ is chosen for which $\mu_{\mathbf{x}}^{\mathrm{o}}(y)$ reaches a maximum membership degree. Picking a random value leads to a nondeterministic control behavior which is usually undesired. The MOM method chooses the mean value of the set of elements $y \in Y$ resulting in maximal membership degrees. The defuzzified control value η might not even be in the set which can lead to unexpected control actions. The COG method defines the value located under the center of gravity of the area $\mu_{\mathbf{x}}^{\mathrm{o}}$ as control value η, *i.e.*

$$\eta = \left(\int_{y \in Y} \mu_{\mathbf{x}}^{\mathrm{o}}(y) \cdot y \, \mathrm{d}y \right) \Big/ \left(\int_{y \in Y} \mu_{\mathbf{x}}^{\mathrm{o}}(y) \, \mathrm{d}y \right). \tag{5}$$

In most control applications, this method shows smooth control behaviors. However, it might even lead to counterintuitive results as well. For a more profound discussion about defuzzification, see *e.g.* [9].

Regarding (3), it is clear that the minimum is used as fuzzy implication. Obviously this does not coincide with its crisp counterpart. Just consider $p \to q$ knowing that p is false. Then $p \to q$ is true regardless of the truth value of q in classical propositional logic. However, $\min\{0,q\}$ is always 0. One way to justify the heuristic of Mamdani and Assilian is to replace the concept of implication by the one of *association* [2]. We say that for a rule R_r an output fuzzy set B_{i_r} is associated with n input fuzzy sets $A_{i_{j,r}}^{(j)}$ for $j = 1,\ldots,n$. This association is modeled by a fuzzy conjunction, *e.g.* the t-norm min.

We retrieve Mamdani's heuristics by extensionality assumptions [5, 6]. If the fuzzy relation R relating the $x^{(j)}$ and y satisfies some extensionality properties, then Mamdani's approach is derived in the same way. Let E and E' be two similarity relations defined on the domains X and Y of x and y, respectively. The extensionality of R on $X \times Y$ thus means

$$\begin{aligned} &\forall x \in X : \forall y,y' \in Y : \top(R(x,y),E'(y,y')) \leq R(x,y'), \\ &\forall x,x' \in X : \forall y \in Y : \top(R(x,y),E(x,x')) \leq R(x',y). \end{aligned} \tag{6}$$

So, if $(x,y) \in R$, then x will be related to the neighborhood y. The same shall hold for y in relation to x. Then $A_r^{(j)}(x) = E(x,a_r^{(j)})$ and $B_r(x) = E'(y,b_r)$ can be seen as fuzzy sets of values that are close to $a_r^{(j)}$ and b_r, respectively. Naturally, $\forall r = 1,\ldots,k : R\left(a_r^{(1)},\ldots,a_r^{(p)}),b_r\right) = 1$. The user thus only needs to define reasonable similarity relations E_j and E' for each input ξ_j and the output η, respectively. Then, using the extensionality properties of R, one gets

$$R(x^{(1)},\ldots,x^{(p)},y) \geq \max_{r=1,\ldots,k} \top\left(A_r^{(1)}(x^{(1)}),\ldots,A_r^{(p)}(x^{(p)}),A_r(y)\right).$$

If we use the t-norm $\top = \min$, then Mamdani's approach to compute the fuzzy output is obtained. In [1, 4] indistinguishability or similarity is expressed as link between the extensionality property and fuzzy equivalence relations. Fuzzy interpolation can be also seen as logical inference given fuzzy information coming from an vaguely known function [8]. Likewise, in [14] fuzzy rules are obtained from set of pairs (a_i,b_i) and similarity relations on X and Y.

4 Success of Mamdani Control in Automobile Industry

In the 1990s many real-world control applications have been greatly solved using Mamdani's approach. Among them are many control problems in the industrial automobile field. The number of publications, however, is really low. Two control applications at Volkswagen AG successfully use Mamdani's approach, *i.e.* the engine idle speed control and the shift-point determination of an automatic transmission [13]. The idle speed controller is based on similarity relations (see Section 3). This helps to view the control function as interpolation of a point-wise known function. The shift-point determination continuously adapts the gearshift schedule between two extremes, *i.e.* economic and sporting. A sport factor is computed to individually adapt the gearshift movements of a driver.

4.1 Engine Idle Speed Control

The task is to control the idle speed of a spark ignition engine. One way is a volumetric control where an auxiliary air regulator alters the cross-section of a bypass to the throttle. This is depicted in Fig. 2.

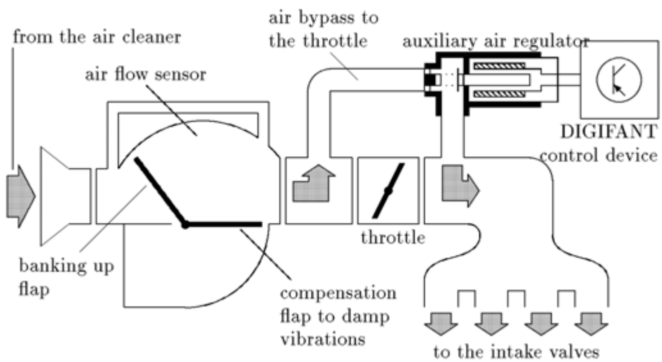

Fig. 2 Principle of the engine idle speed control.

The pulse width of the auxiliary air regulator is changed by the controller. If there is a drop in the number of revolutions, then the controller forces the auxiliary air regulator to increase the bypass cross-section. The air flow sensor measures the increased air flow rate and thus notifies the controller. The new quantity for the fuel injection must be computed. Due to a higher air flow rate, the engine yields more torque. This again results in a higher number of revolutions which could be reduced analogously by decreasing the bypass cross-section.

Both fuel consumption and pollutant emissions should be ultimately reduced. This can be reached by slowing down the idle speed. However, a switching on of certain automobile facilities, *e.g.* air-conditioning system, forces the number of revolutions to drop. Hence the controller must be very flexible. More problems involved in this control application can be found in [13].

Due to this motivating problem, VW and our working group cooperated in developing a Mamdani fuzzy controller based on similarity relations. The resulting fuzzy controller was easier to design and showed an improved control behavior compared to classical control approaches. Similarity relations to represent indistinguishability or similarity of points within a certain vicinity seems to be a natural modeling way for engineers.

In fact, indistinguishability is not produced by measurement errors or deviations. It just expresses that arbitrary precision is not necessary to control a system. A control expert must thus specify a set of k input-output tuples $\left(\left(x_r^{(1)},\ldots,x_r^{(p)}\right),y_r\right)$. For each $r=1,\ldots,k$, the output value y_r seems appropriate for the input $\left(x_r^{(1)},\ldots,x_r^{(p)}\right)$. So, the human expert defines the partial control function φ_0.

In the 1990s the question to be answered was to compute a suitable output value for an arbitrary input given specified similarity relations and φ_0 [13]. Using the extensionality properties defined in (6), one obtains Mamdani's fuzzy output directly by computing the extensional hull of φ_0 given the similarity relations. The partial control function φ_0 can thus be reinterpreted as k control rules of the form:

$$R_r: \textbf{if } \xi_1 \textbf{ is } \textit{approximately } x_r^{(1)} \textbf{ and} \ldots \textbf{ and } \xi_p \textbf{ is } \textit{approximately } x_r^{(p)} \\ \textbf{then } \eta \textbf{ is } \textit{approximately } y_r.$$

A more profound theoretical analysis of this approach can be found in [5].

To control the engine idle speed controller, two input variables are needed:

1. the deviation dREV [rpm] of the number of revolutions to the set value, and
2. the gradient gREV [rpm] of the number of revolutions between two ignitions.

The only output variable is the change of current dAARCUR for the auxiliary air regulator. The controller is shown in Fig. 3.

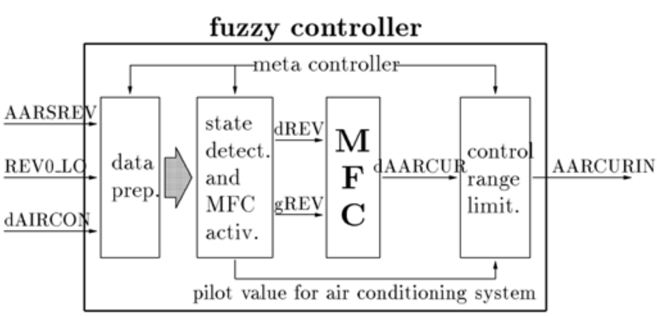

Fig. 3 Structure of the fuzzy controller.

The knowledge to control the engine idle speed controller was extracted by measurement data obtained from idle speed experiments. The partial control mapping $\varphi_0 : X_{(\mathrm{dREV})} \times X_{(\mathrm{gREV})} \to Y_{(\mathrm{dAARCUR})}$ has been specified as in Table 1 (left-hand side).

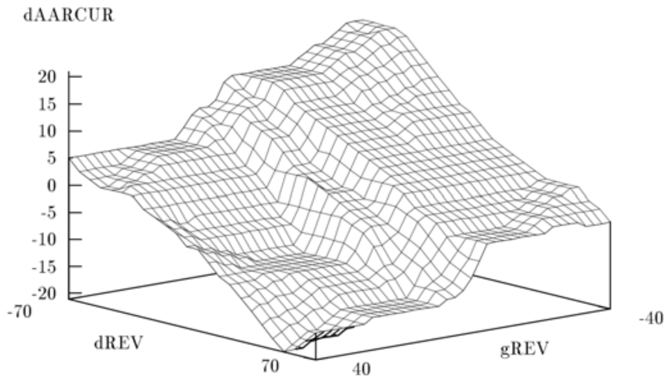

Fig. 4 Performance characteristics.

Using a similarity relation and φ_0, the fuzzy controller was defined. Its induced control surface is shown in Fig. 4 as a grid of supporting points. The center of area (COA) method has been used for defuzzification. To obtain the corresponding Mamdani fuzzy controller, each point of φ_0 was associated with a linguistic term, *e.g.* negative big (nb), negative medium (nm), negative small (ns), approximately zero (az), and so on. The obtained fuzzy partitions of all three variables are shown in Fig. 5–7, respectively. The partial mapping φ_0 was translated into linguistic rules of the form

if dREV **is** *A* **and** gREV **is** *B* **then** dAARCUR **is** *C*.

The complete set of rules is given on the right-hand side of Table 1.

Table 1 The partial control mapping φ_0 (left-hand side) and its corresponding fuzzy rule base (right-hand side).

		gREV						
		-40	-6	-3	0	3	6	40
	-70	20	15	15	10	10	5	5
	-50	20	15	10	10	5	5	0
	-30	15	10	5	5	5	0	0
dREV	0	5	5	0	0	0	-10	-5
	30	0	0	0	-5	-5	-10	-15
	50	0	-5	-5	-10	-15	-15	-20
	70	-5	-5	-10	-15	15	15	15

		gREV						
		nb	nm	ns	az	ps	pm	pb
	nb	ph	pb	pb	pm	pm	ps	ps
	nm	ph	pb	pm	pm	ps	ps	az
	ns	pb	pm	ps	ps	az	az	az
dREV	az	ps	ps	az	az	az	nm	ns
	ps	az	az	az	ns	ns	nm	nb
	pm	az	ns	ns	ns	nb	nb	nh
	pb	ns	ns	nm	nb	nb	nb	nh

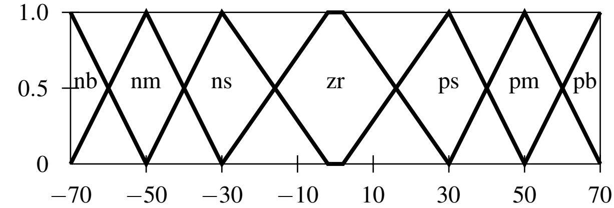

Fig. 5 Deviation dREV of the number of revolutions.

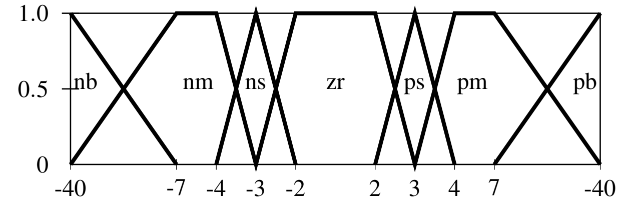

Fig. 6 Gradient gREV of the number of revolutions.

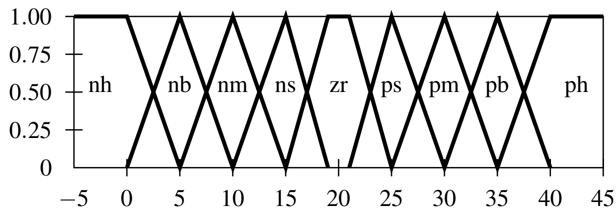

Fig. 7 Change of current dAARCUR for the auxiliary air regulator.

In [5, 13] the Mamdani fuzzy controller shows a very smooth control behavior compared to its serial counterpart. Furthermore the fuzzy controller reaches the desired set point precisely and fast. Its behavior is robust even with slowly increasing load. Thus the number of revolutions does not lead to any vibration even after extreme changes of load occur.

4.2 Flowing Shift-Point Determination

Conventional automatic transmissions select gears based on so-called gearshift diagrams. Here, the gearshift simply depends on the accelerator position and the velocity. A lagging between up and down shift avoids oscillating gearshift when the velocity varies slightly, *e.g.* during stop-and-go traffic. For a standardized behavior, a fixed diagram works well. Until 1994, the VW gear box had two different types of gearshift diagrams, *i.e.* economic “ECO” and sporting “SPORT”. An economic gearshift diagram switches gears at a low number of revolutions to reduce the fuel consumption. A sporting one leads to gearshifts at a higher number of revolutions. Since 1991 it was a research issue at VW to develop an individual adaption of shift-points. No additional sensors should be used to observe the driver.

The idea was that the car "observes" the driver [13] and classifies him or her into calm, normal, sportive (assigning a sport factor $\in [0,1]$), or nervous (to calm down the driver). A test car at VW was operated by many different drivers. These people were classified by a human expert (passenger). Simultaneously, 14 attributes were continuously measured during test drives. Among them were variables like the velocity of the car, the position of the acceleration pedal, the speed of the acceleration pedal, the kick down, or the steering wheel angle.

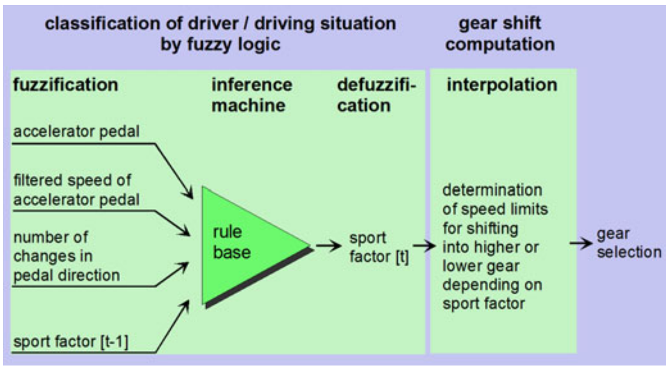

Fig. 8 Flowing shift-point determination with fuzzy logic.

The final Mamdani controller was based on 4 input variables and one output. The basic structure of the controller is shown in Fig. 8. In total, 7 rules could be identified at which the antecedent consists of up to 4 clauses. The program was highly optimized: It used 24 Byte RAM and 702 Byte ROM, *i.e.* less than 1 KB. The runtime was 80 ms which means that 12 times per second a new sport factor was assigned. The controller is in series since January 1995. It shows an excellent performance.

5 Conclusions

We reviewed the fuzzy control approach of Abe Mamdani. We gave a possible interpretation to justify this heuristic method, *i.e.* knowledge-based interpolation based on input-output points of a vaguely known control function. This view has been developed in corporation with Volkswagen AG, Wolfsburg during the nineties of the last millennium. We reviewed two real-world control applications that have been successfully handled based on this interpretation.

This paper clearly demonstrates that Abe Mamdani was a man of vision. In the ESPRIT Basic Research Action 3085, entitled *Defeasible Reasoning and Uncertainty Management Systems (DRUMS)* [3, 11], all participants have been impressed by his broad knowledge in different scientific disciplines. Everybody was delighted by his sense of humor and his modesty. In the meetings of the Scientific Committee of the European Centre for Soft Computing[1] in Mieres, Asturias, Spain the second author regularly experienced the pleasure of friendly exchanges with Abe. The fuzzy community will always remain grateful to Abe for having been the first to show the road to practical applications.

References

1. Boixader, D., Jacas, J.: Extensionality based approximate reasoning. International Journal of Approximate Reasoning 19(3-4), 221–230 (1998); doi:10.1016/S0888-613X(98)00018-8
2. Cordón, O., del Jesus, M.J., Herrera, F.: A proposal on reasoning methods in fuzzy rule-based classification systems. International Journal of Approximate Reasoning 20(1), 21–45 (1999); doi:10.1016/S0888-613X(00)88942-2
3. Gabbay, D.M., Kruse, R. (eds.): Abductive Reasoning and Uncertainty Management Systems, Handbook of Defeasible Reasoning and Uncertainty Management Systems, vol. 4. Kluwer Academic Publishers, Dordrecht (2000)
4. Klawonn, F., Castro, J.L.: Similarity in fuzzy reasoning. Mathware & Soft Computing 2(3), 197–228 (1995)
5. Klawonn, F., Gebhardt, J., Kruse, R.: Fuzzy control on the basis of equality relations with an example from idle speed control. IEEE Transactions on Fuzzy Systems 3(3), 336–350 (1995), doi:10.1109/91.413237
6. Klawonn, F., Kruse, R.: Equality relations as a basis for fuzzy control. Fuzzy Sets and Systems 54(2), 147–156 (1993), doi:10.1016/0165-0114(93)90272-J
7. Klawonn, F., Kruse, R.: The Inherent Indistinguishability in Fuzzy Systems. In: Lenski, W. (ed.) Logic versus Approximation. LNCS, vol. 3075, pp. 6–17. Springer, Heidelberg (2004)
8. Klawonn, F., Novák, V.: The relation between inference and interpolation in the framework of fuzzy systems. Fuzzy Sets and Systems 81(3), 331–354 (1996), doi:10.1016/0165-0114(96)83710-9
9. Kruse, R., Gebhardt, J., Klawonn, F.: Foundations of Fuzzy Systems. John Wiley & Sons Ltd., Chichester (1994)
10. Kruse, R., Gebhardt, J., Palm, R. (eds.): Fuzzy Systems in Computer Science. Vieweg-Verlag, Braunschweig/Wiesbaden, Germany (1994)
11. Kruse, R., Siegel, P. (eds.): ECSQAU 1991 and ECSQARU 1991. LNCS, vol. 548. Springer, Heidelberg (1991)
12. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. International Journal of Man-Machine Studies 7(1), 1–13 (1975), doi:10.1016/S0020-7373(75)80002-2
13. Schröder, M., Petersen, R., Klawonn, F., Kruse, R.: Two Paradigms of Automotive Fuzzy Logic Applications. In: Jamshidi, M., Titli, A., Zadeh, L., Boverie, S. (eds.) Applications of Fuzzy Logic: Towards High Machine Intelligence Quotient Systems. Environmental and Intelligent Manufacturing Systems Series, vol. 9, pp. 153–174. Prentice-Hall Inc., Upper Saddle River (1997)
14. Sudkamp, T.: Similarity, interpolation, and fuzzy rule construction. Fuzzy Sets and Systems 58(1), 73–86 (1993), doi:10.1016/0165-0114(93)90323-A

[1] http://www.softcomputing.es/