Fifth IFSA World Congress (1993), 1354-1357

Optimization of Fuzzy Systems by Switching Reasoning Methods Dynamically

Michael H. SMITH[†] *and Hideyuki TAKAGI*[‡]

[†‡] Computer Science Division, University of California, Berkeley, CA 95720
([‡] Central Research Laboratories, Matsushita Electric Industrial Co., Ltd., Moriguchi, 570 Japan)
mhs@robotics.berkeley.edu, takagi@diva.berkeley.edu, [‡]FAX (510)642-5775

Abstract

This paper proposes that the best reasoning (i.e. rule evaluation) method which should be used in a fuzzy system significantly depends on the reasoning environment. It is shown that allowing for dynamic switching of reasoning methods leads to better performance, even when only two different reasoning methods are considered. This paper discusses DSFS (Dynamic Switching Fuzzy System) which dynamically switches and finds the best reasoning method (from among 80 different possible reasoning methods) to use depending on the reasoning situation. To overcome the reasoning speed and memory problem of DSFS due to its computational requirements, the DSFS Switching Reasoning Table method is proposed and its higher performance as compared to a conventional fuzzy system is shown. Finally, efforts to obtain general relationships between the characteristics of different reasoning methods and the actual control surface/environment is discussed.

1 Introduction

Usually only one reasoning (i.e. rule evaluation) method such as the Min Fuzzy Intersection operator and Height defuzzification method is used to perform the necessary rule-base inference in fuzzy systems. These fuzzy systems are often then tuned by changing the membership functions and/or rules by various methods such as genetic algorithms [1].

In [6, 7], tuning fuzzy systems by dynamically switching and using different reasoning methods as the reasoning environment changes was introduced. This methodology was implemented and evaluated by the design and use of DSFS (Dynamic Switching Fuzzy System). DSFS showed that the optimal reasoning method mainly depends on the reasoning environment. DSFS also showed that switching between different reasoning methods provided better performance and fault tolerance than conventional Fuzzy Logic Controllers (FLCs) using only one reasoning method. DSFS was also found to be highly adaptive to changing reasoning environments and situations.

While DSFS provides a powerful tool to optimize a fuzzy system, it is slower and requires more memory than conventional FLCs using only one reasoning method. This is because DSFS compares and switches between 80 different reasoning methods at each sampling point in time to find the best performing method.

The purpose of this paper is to address this problem. First, in section 2 various reasoning methods and their differences in performance in balancing an inverted pendulum are discussed. In section 3 the advantages of switching between different reasoning methods is shown and DSFS is briefly discussed. In section 4, DSFS created Switching Tables which allow for dynamic switching of reasoning methods by table lookup rather than by using parallel fuzzy systems are introduced. These tables show that in the case of the inverted pendulum, the relationship rules defining which reasoning method to use in various reasoning situations are not complex and are easily determined. Finally, in section 5, some conclusions are discussed.

Table 1: Several reasoning methods

	FUZZY INTERSECTION OPERATORS (I)
1)	MIN
2)	HAMACHER PRODUCT
3)	ALGEBRAIC PRODUCT
4)	EINSTEIN PRODUCT
5)	BOUNDED DIFFERENCE
6)	DRASTIC PRODUCT
7)	HAMACHER INTERSECTION
8)	YAGER INTERSECTION
9)	DUBOIS INTERSECTION
10)	FUZZY AND
	DEFUZZIFICATION METHODS (D)
Defuzzify-then combine:	
1)	HEIGHT METHOD
2)	AREA METHOD
3)	BEST NO. of RULES METHOD
4)	WINNING RULE CENTROID
Combine-then-defuzzify:	
5)	PRODUCT*SUM of GRAVITY
6)	α-CUT SUM of GRAVITY
7)	CENTER of GRAVITY
8)	α-CUT MEAN of MAXIMUM

2 Various Reasoning Methods and Differences in Performance

Table 1 lists 10 Fuzzy Intersection operators[10, 3] and 8 Defuzzification methods [3, 4, 5] commonly used. They can be combined to create 80 different reasoning methods. Each reasoning method may sometimes perform better than others under different circumstances at different times and each will often give different results. One example is the variation in performance and sensitivity of the Algebraic Product versus the Min Fuzzy Intersection operator.

Figure 1 shows the performance of various reasoning methods in attempting to balance an inverted pendulum starting from rest at 34 degrees. The knowledge base used was a TSK model of four rules (with maximum allowable force of 38 N). The number of rules, shape of the membership functions, and the parameters of the consequents were tuned by using a Genetic Algorithm [1].

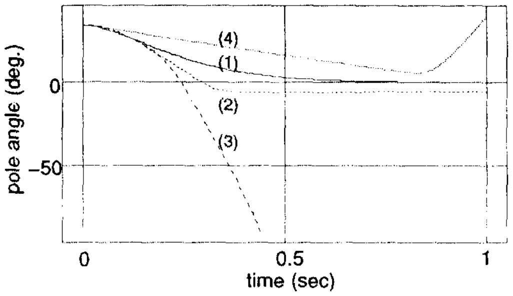


Figure 1: Performance of several reasoning methods: (1) Min + Height, (2) Fuzzy AND + Winning Rule Centroid, (3) Fuzzy AND + α-cut Sum of Gravity, (4) Min + α-cut Mean of Maximum

The four rules are ($x_1 = x_1(t)$ and $x_2 = x_2(t)$):
IF x_1 is A_1 and x_2 is B_1 THEN $y = 0.44x_1 + 1.02x_2 - 31.65$
IF x_1 is A_2 and x_2 is B_1 THEN $y = 1.54x_1 - 0.61x_2 - 30.14$
IF x_1 is A_1 and x_2 is B_2 THEN $y = 1.54x_1 - 0.61x_2 + 30.14$
IF x_1 is A_2 and x_2 is B_2 THEN $y = 0.44x_1 + 1.02x_2 + 31.65$
The membership functions are triangular shaped:
$A_1 = \{-119.65, -62.12, 4.59\}$, $A_2 = \{-4.59, 62.12, 119.65\}$,
B_1
$= \{-219, -1.99, 238.56\}$, and $B_2 = \{-238.56, 1.99, 219.64\}$.

The reasoning methods Min + α-cut Mean of Maximum, Fuzzy AND + α-cut Sum of Gravity, and Fuzzy AND + Winning Rule Centroid were all unsuccessful in balancing the pendulum. The reasoning method Min + Height Method, which is one of the most frequently used reasoning methods, was successful.

The failed Fuzzy AND + α-cut Sum of Gravity reasoning method was able to accelerate the pole much faster starting from rest at 34° than the Min+ Height Method. However, the Fuzzy AND + α-cut Sum of Gravity failed by not being able to slow the pendulum down due to over-acceleration. The Min+Height Method was able to balance the pendulum by accelerating the pendulum much slower and then de-accelerating.

Since the performance of a fuzzy system varies depending on which reasoning method is used, fuzzy systems have been tuned by trying to find the best overall single reasoning method and/or by tuning parametric Fuzzy Intersection operators and/or Defuzzification methods whose characteristics can be modified by parameters [2, 8, 9].

3 Dynamically Switching Reasoning Methods

Figure 2 shows the increased performance when the FLC, in balancing an inverted pendulum, is allowed to switch between the two reasoning methods Min + Height and Fuzzy AND + α-cut Sum of Gravity by using the following relationship rule:

IF $\theta > 18$ THEN use Fuzzy AND + α-cut Sum of Gravity
ELSE use Min + Height Method

Switching between these two reasoning methods allows the fuzzy system to use the powerful "acceleration" of the Fuzzy AND + α-cut Sum of Gravity method when far away from the set-point while using the good damping effect of the Min + Height Method when close to the set-point. Note that the performance of all of these experiments was obtained with using the above TSK knowledge base where the rules, shape of the membership functions, and the parameters of the consequents were already tuned by a Genetic Algorithm. Allowing two reasoning methods to be used instead of one clearly achieves significantly better performance.

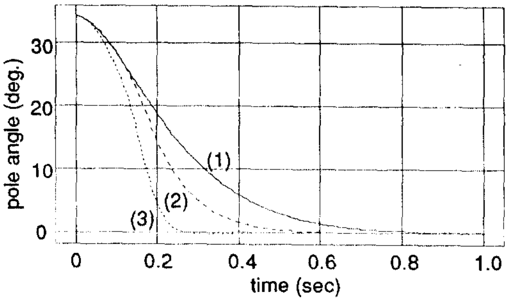


Figure 2: Performance when switching between different reasoning methods is allowed: (1) Min + Height only, (2) Min + Height and Fuzzy AND + α-cut Sum of Gravity, (3) DSFS (80 different reasoning methods)

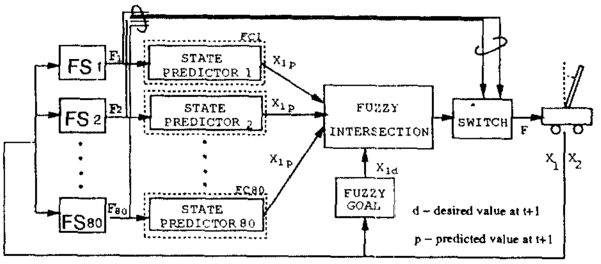
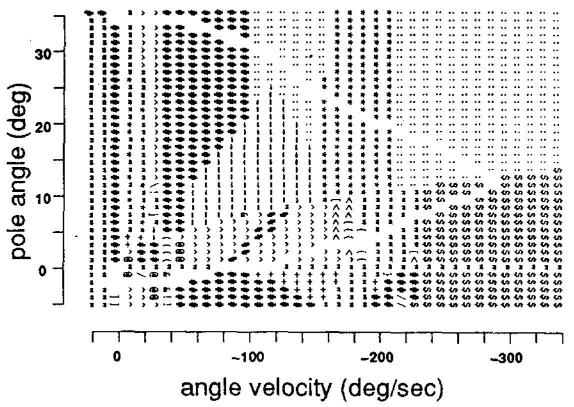


Figure 3: System Diagram

DSFS, figure 3, was designed so that as the reasoning situation changes, it dynamically compares and chooses the best performing reasoning method (out of the 80 possible combinations of the Fuzzy Intersection operators and Defuzzification methods shown in Table 1) [6, 7]. DSFS dynamically switches between the different reasoning methods and finds the best one to use. For example, at time t DSFS might use Min + Height method whereas at time $t+1$ DSFS might use Algebraic Product + α-Cut Sum of Gravity. Figure 2 shows the performance of DSFS in balancing an pendulum while switching between the 80 different reasoning methods using the above TSK model. DSFS's performance is better than the two reasoning methods switching fuzzy system described above or the single Min + Height fuzzy system.

4 DSFS Switching Tables

While DSFS provides a powerful tool to optimize a fuzzy system, it is slower and requires more memory than conventional FLCs using only one reasoning method. This is because DSFS compares and switches between 80 different reasoning methods at each sampling point in time to find the best performing one. To solve this problem, DSFS Switching Tables can be created by DSFS. These tables also allow for dynamic switching of reasoning methods as DSFS does, but by table lookup rather than by using parallel fuzzy systems. Furthermore, by examining these tables, the relationship rules defining which reasoning method to use in various reasoning situations may not be complex and



	I	D		I	D		I	D		I	D		I	D
:	10	6	*	1	6	~	1	8	#	10	4	∧	2	7
"	1	1	@	3	7	$	10	1	!	4	6	+	1	4
&	1	7	}	2	1	\	3	6	]	5	6	(	2	8
{	2	1	<	3	1	%	8	1	[	4	1	/	5	1

Figure 4: Switching Reasoning Table and control path of controller balancing the pendulum starting from rest at 34° overlaid onto the switching reasoning table. I and D in table are Intersection Operators and Defuzzification methods defined in Table 1.

may be easily determined. An example of a simple relationship rule is the above IF-THEN rule used for switching between two reasoning methods.

A Switching Table, Figure 4, was created by using DSFS to generate over 28,000 sample data points (θ, $\delta\theta/\delta t$, Fuzzy Intersection Operator, and Defuzzification Method). This switching table shows which reasoning method should be used depending on the reasoning situation (θ,$\delta\theta/\delta t$ – which were only calculated for a partial range of possible values). Due to lack of complete data which led to the approximating of switching boundaries, slight improvement of performance was obtained by manual tuning.

The DSFS Switching Table partitions the reasoning situations (values of θ, $\delta\theta/\delta t$) into different regions where different reasoning methods perform better than others. The control path of one experiment of balancing the pendulum upright (where the pendulum was started at rest at position $\theta = 34$ degrees) is overlaid on top of the switching table. One can see which regions are crossed and which reasoning methods are invoked. Figure 5 shows the performance of DSFS, the original Switching Table generated by DSFS, the manually-tuned DSFS Switching Table, and the single Min + Height method for this experiment. The DSFS Switching Tables perform almost as well as DSFS and much better then the FLC using only the Min + Height reasoning method. However, it should be noted that if the pendulum is started from rest close to the set-point of 0°, the DSFS Switching Tables perform poorly since $\delta\theta/\delta t$ is only sampled every ten degrees/second. Hence the tables are not sensitive enough for that control region. More detailed tables can be created if necessary or regions close to the set-point can be set to use the Min + Height method.

Figure 6 shows the surface plot of the Switching Table with the corresponding output forces generated by using the recommended reasoning methods. When θ and $\delta\theta/\delta t$ are both large, then the appropriate maximum force ($38N$) is applied (similar to saturation of actuators in a conventional controller). As θ or $\delta\theta/\delta t$ become smaller, the switching table switches to other reasoning methods which, depending on the reasoning situation, either generate smaller forces to start coasting or, by applying maximum negative force ($-38N$), to start braking towards the goal of 0 degrees, 0 degrees/second. When θ and $\delta\theta/\delta t$ are close to the goal state, the switching table changes the force more slowly and provides a smoother transition between states. (The surface plot is coarse and less precise when θ and/or $\delta\theta/\delta t$ are close to 0 as $\delta\theta/\delta t$ is only plotted every 10 degrees/second). Finally, if the DSFS Switching Table in Figure 4 is compared with it's control surface plot (Figure 6, and with the contour map of the control surface, Figure 7), the contours of the changes in the output forces of the surface plot can be mapped to the changes in switching regions shown in the DSFS Switching Table. This allows one to easily identify general relationships between different reasoning methods and the reasoning situation/environment.

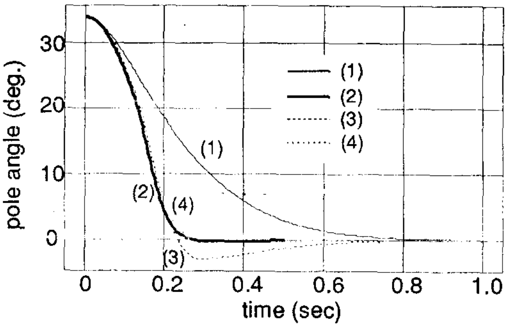


Figure 5: Performance of a switching reasoning table: (1) Min + Height, (2) DSFS, (3) switching reasoning table method obtained from DSFS, (4) switching reasoning table method manually tuned

An analogy of the difference between a conventional fuzzy controller using only one reasoning method and a FLC using DSFS Switching Tables is that of a driver who sees an traffic light far ahead when stopped at a light. He can accelerate briefly and then begin to take his foot partially off the gas petal. He can continue to slowly reduce the speed of the car by feeding it minimal gas, adjusting for bumps and dips in the road, until he comes to a stop (actions similar to using the Min + Height method in the above TSK model). Or he can accelerate for a longer time, and when he is much closer to the traffic light, he can then step hard on the brakes, maintain that braking force (saturation of the actuators) until he is closer to the light, and then he can slowly reduce the braking force until he comes to a smooth stop, adjusting for bumps and dips in the road. Using brakes will get the driver to the light faster while still providing a smooth, comfortable ride (actions similar to DSFS Switching Tables).

5 Conclusions

The best reasoning method to be used in a fuzzy system depends significantly on the reasoning environment. Allowing for dynamic switching between different reasoning methods as the reasoning situation changes leads to better performance, even when only two different reasoning methods are considered.

Using DSFS Switching Tables in FLCs has led to higher performance than conventional fuzzy systems using only a single reasoning method. The best reasoning methods chosen by the DSFS Switching Tables are mostly decided by the static in-

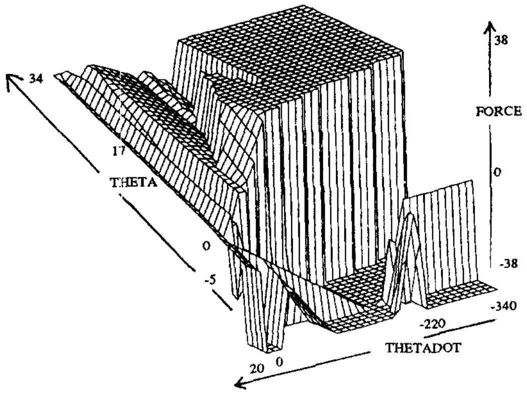


Figure 6: Control surface

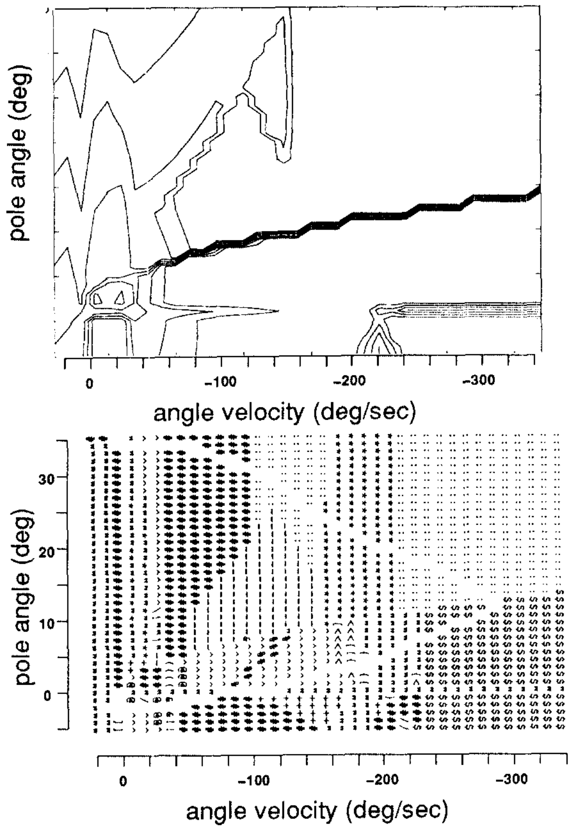


Figure 7: Control surface and switchin reasoning table shown as Figure 4.

formation of the reasoning environment (which are inputs of a controller). It cannot be concluded whether the static reasoning environment alone completely decides the best reasoning method because of the lack of precision of our current experiments.

As Figure 4 shows, the relationship rules defining which reasoning method should be used for various reasoning situations are not complex and are easily determined. The results of this paper allows one to implement a practical DSFS through the use of switching tables. DSFS can be realized without requiring the large memory and computation time needed for multi-parallel fuzzy systems.

Acknowledgment

We would like to thank Prof. R. Fearing of UC Berkeley for his discussion. This research is supported in part by NASA Grant NCC-2-275, MICRO State Program Award No.90-191, and EPRI Agreement RP8010-34.

References

[1] M.A. Lee and H. Takagi, "Integrating Design Stages of Fuzzy Systems using Genetic Algorithms," IEEE 2nd Int'l Conf. on Fuzzy Systems, 1993, pp.612-617.

[2] T. Miyoshi, S.Tano, Y. Kato, and T. Arnould, "Operator Tuning in Fuzzy Production Rules Using Neural Networks," Proceedings of the Second IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'93), 1993, pp.641-646.

[3] M. Mizumoto, "Pictorial representations of fuzzy connectives, Part I: cases of T-norms,T-conorms, and averaging operators,"Fuzzy Sets and Systems, vol.31, 1989, pp.217-242.

[4] M. Mizumoto, "Comparison of various fuzzy reasoning methods," 2nd IFSA Congress, Tokyo, 1987, pp.2-7.

[5] M. Mizumoto, "Fuzzy controls under various defuzzifier methods,"Int Workshop on Fuzzy Systems Applications (IIZUKA'88), 1988, pp.143-146.

[6] M.H. Smith, "Evaluation of performance and robustness of a parallel dynamic switching fuzzy system," 2nd Int'l Workshop on Industrial Fuzzy Control and Intelligent Systems (IFIS'92), 1992, pp.163-172.

[7] M.H. Smith, "Parallel Dynamic Switching of Reasoning Methods in a Fuzzy System," Proceedings of the Second IEEE International Conference on Fuzzy Systems(FUZZ-IEEE'93),pp.968-973.

[8] R.R. Yager, "On a general class of fuzzy connectives," Fuzzy Sets and Systems, vol.4, 1980, pp.235-242.

[9] R.R. Yager and D.P. Filev, "Adaptive defuzzification for fuzzy system modelling' North American Fuzzy Logic Processing Society, 1992, pp.135-142.

[10] H.J. Zimmermann, Fuzzy Set Theory and Its Applications. Boston,Massachusetts: Kulwer Academic Publishers, 1991, chapter 3.