Int. Computer Symposium, Dec. 15-17, 2004, Taipei, Taiwan.

Generalized Fuzzy Automata for Fuzzy Feedback Control with Words

Chao-Lieh Chen

Department of Electronic Engineering
Kun-Shan University of Technology
No. 949, Da-Wan Rd. Yung-Kang, Tainan, TAIWAN
frederic@mail.ksut.edu.tw

Abstract - *A model of fuzzy control based on Generalized Fuzzy Automata (GFA) is proposed. The fuzzy controllers based on the GFA theory control plants with words since system states, inputs, and outputs are encoded in linguistic terms when modeling with GFAs. Therefore, the behavior of a control system is described with words. Mathematically we prove that GFA realize general fuzzy feedback controls. This novelty provides implementation, no matter in hardware or software, of fuzzy feedback controllers in a systematic, unified, effective, and objective paradigm. The properties of the rule base including fuzzy rules and membership functions of the controller are objectively defined without much subjective domain expertise involved. The proofs also demonstrate how fuzzy feedback control with words is performed.*

Keywords: Fuzzy Automata, Fuzzy Feedback Control, Computing With words

1. Introduction

Fuzzy feedback controls appear in many areas. Most of the environments of the control problems are with uncertainties. For example, a simple heating problem where the temperature of the plant is required to be consistent with specified variations. In classical control, to identify a system one need to know exactly all parameters such as mass of plant, environment temperature, and material properties exist in the system. However, in many cases this is not possible. Even though a system is identified, non-linear characteristics make the design of the controller difficult. Therefore, many articles present fuzzy feedback controllers as the solutions for both uncertainties and non-linearities [1, 6]. In these articles, a fuzzy feedback controller is equivalent to a fuzzy rule base where linguistic terms are characterized by membership functions. Domain expert gives the membership functions and the fuzzy rules. However, there is still a gap to control with words. The gap exists in where there is no *unified* behavioral description about the whole control systems including the environment and controller itself. Without the unified behavioral description model when suffering another control problem, the fuzzy control system need to be re-designed by consulting the domain expert. In this paper we adopt the generalized fuzzy automata (GFA) theory [2] to deal with behavioral descriptions in words of control systems.

Automata theory is essential in computer science. Behavior of an automaton presents behavior of a computation such as that of a controller. Mathematically, we said that a Moore typed automaton realize a controller iff the behavior of the automaton is an input-state homomorphism of the behavior of the controller. Behaviors of a specific class of automata are usually studied categorically [3, 4, 5]. Instead of categorically studying properties of the GFA automata, in this paper we show that for a general feedback control system there is a GFA automaton behaves the same. That is, for each feedback control problem, there exists an automaton in the GFA category realizes it.

This paper is organized as follows. In Section II, the words as controls and the class of generalized fuzzy automata are defined. For the generic realization of fuzzy control, in Section III, we prove algebraically that there exists an isomorphism, that is, a GFA automaton, of the control. In Section IV, examples and the design flow of a fuzzy feedback control is demonstrated. The conclusions are given in Section V.

2. Generalized Fuzzy Automata

In this section we define the objects and modules in generalized fuzzy automata (GFA).

2.1. Algebraic Structure of Words

We now study the algebraic structure of the words used in the GFA automata. The words are used in control systems. We first define the groupoid of the words. Since for each word there is a corresponding fuzzy set that interprets it, a set of words is regarded as level 2 fuzzy set.

Definition 1 (*Groupoid of words*) Let $L_n(X)$ be a level-n fuzzy set with universe $X^{(n-1)}$. A word w of Σ is a linguistic term associated with a fuzzy subset and a membership function $\mu_w(x)$ of Σ. A groupoid $L_2(\Sigma)^*$ of words is a Kleen's closure over *level-2* fuzzy set $L_2(\Sigma)$ of words with universe Σ. That is,

$L_2(\Sigma) = \{w \mid w \text{ is a word of } \Sigma\}$,

$(\forall w_1 \in L_2(\Sigma))(\forall w_2 \in L_2(\Sigma))[w_1 \cdot w_2 = w_1 w_2]$,

$w^0 = \varepsilon$ and $(\forall w_1 \in L_2(\Sigma))[w_1\, \varepsilon = \varepsilon\, w_1 = w_1]$, and

$L_2(\Sigma)^* = \{w \mid w \text{ is a word of } \Sigma\}^* = \bigcup_{i=0}^{\infty} (L_2(\Sigma))^i$

□

Note that the power set $2^\Sigma \subseteq L_2(\Sigma)$ since $\forall 2^x \in 2^\Sigma$ is a special fuzzy subset of Σ such that the membership grade of each element is 1. That is,

$$(\forall p \in 2^\Sigma)(x \in p)[\mu_p(x) = 1]$$

Moreover, the closure $\Sigma^* \subseteq L_2(\Sigma)^*$ since for classically defined strings s_1 and s_2 we have $(\forall s_1 \in \Sigma^*)(\forall s_2 \in \Sigma^*)[s_1 \cdot s_2 = s_1 s_2]$. This is based on the fact that if $(\exists n \in \mathbb{N})[s_1 = s_{11}s_{12}s_{13},\ldots,s_{1n}]$ and $(\exists m \in \mathbb{N})[s_2 = s_{21}s_{22}s_{23},\ldots,s_{2m}]$, then s_{1i} and s_{2j} are special fuzzy sets, called singletons, in $L_2(\Sigma)$. In Definition 1, extension principle for the concatenation operator "·" is not applied because it is not required in GFA automata. The abstraction in Definition 1 is very important such that we can show that fuzzy computing based on GFA is massively parallel and any fuzzy computation is a super composition using *t*-norms and *s*-norms (*t*-co-norm). A sequence of fuzzy input $X(t)$ along time t is a member of $L_2(\Sigma)^*$.

2.2. Definition of the Generalized Fuzzy Automata

In this section, with groupoid of words we define the GFA in Definition 2. Following that we then define the transition caused by an input sequence in Definition 3. In the following we use height-bounded observations (HBOs)[2], each of which is extended from an LR fuzzy set $(\alpha, m, \beta)_{LR}$ -- an normalized and convex observation operations $\boldsymbol{O}$ on a fuzzy set [1] with limited height h and is re-denoted (α, m, β, h). An observed input $X(t)$ and an observed state $S(t)$ at time t are then denoted $(\alpha_X(t), m_X(t), \beta_X(t), h_X(t))$ and $(\alpha_S(t), m_S(t), \beta_S(t), h_S(t))$ *respectively.*

Definition 2. (*Generalized Fuzzy Automata*): A generalized fuzzy automaton is a quintuple GFA $\boldsymbol{M}(\delta, \Sigma, U, S_0, F, T)$, where each of the state transitions $\{\delta(X_r(t), S_r(t), dT_r) \mid r \in \mathbb{N}\}$ is a set of transition fuzzy rules defined as

Rule r:
"IF, at time t, the observed input is
$X_r(t) = (\alpha_{Xr}(t), m_{Xr}(t), \beta_{Xr}(t), h_{Xr}(t))$
AND the observed state is
$S_r(t) = (\alpha_{Sr}(t), m_{Sr}(t), \beta_{Sr}(t), h_{Sr}(t))$
THEN about dT_r later, the state will become
$S_r^*(t + dT_r)$
$= (\alpha_{Sr^*}(t + dT_r), m_{Sr^*}(t + dT_r), \beta_{Sr^*}(t + dT_r), h_{Sr^*}(t + dT_r))$"

(1)

The element Σ is the set of fuzzy inputs; U is the universe of fuzzy states and is a crisp set. By denoting the support of a fuzzy set M with $\sigma(M)$, we have U as follows:

$$U \subseteq \sigma(S_0) \cup \left\{ \bigcup_{\hat{X} \in \Sigma^*} \sigma\left(\hat{\delta}\left(S_0, \hat{X}\right)\right) \right\} \quad (2)$$

In addition, S_0 is the initial fuzzy state, F denoting the set of finals is a crisp subset of U, and T is the universe of fuzzy time ticks, which are denoted as dTs and specified in the transition rules, that is

$$T \subseteq \{dT \mid (\exists\, S(t) \subseteq U)(\exists\, S(t+dT) \subseteq U)[S(t+dT) = \delta(X(t), S(t))]\}. \quad (3)$$

One can easily verify that any fuzzy state S is a subset of U since $\sigma(S) \subseteq U$. The inputs, states, and fuzzy time ticks are also called *L*-fuzzy sets [9] with co-domain a complete lattice.

□

Definition 3. ($\hat{\delta}$): Let $\hat{X} = \hat{X}'X \in L_2(\Sigma)^*$ is an input sequence whose last input fuzzy set is X, then $\hat{\delta}$ is defined recursively by

$$\hat{\delta} = \delta\left(X, \hat{\delta}\left(\hat{X}', S\right)\right) \quad (4)$$

where δ is the transition defined in (1).

□

By Definition 2, a GFA is a *variable structured* (time varying) automaton, which performs computation in parallel. At each time stamp with observed state $S(t)$ and input $X(t)$, there would be more than one transition rules have nonzero firing levels, the matching degrees of premises of transition rules. When *t*-norm and its co-norm are expressed as multiplication and addition in $\mathbb{R}$, one can easily show that δ is linear and such that a GFA becomes linear automaton.

3. Fuzzy Feedback Control Realization

We develop from the fuzzy feedback control model introduced in [1] for realization. The controller for this model was proven making the fuzzy feedback control system definitely attainable [1] even in a noisy environment producing uncertain observations. The control set used in [1] is a crisp set {u}={-1, 0, 1}. We generalize the control sequences into groupoid $L_2(\Sigma)^*$ and the model is then further generalized using GFA theory.

Definition 4. (*One-dimensional Fuzzy Feedback Control System*[1]) An one-dimensional Fuzzy Feedback control system (1-D FFCS) has uncertain fuzzy state with support $[\xi_1, \xi_2]$, the observer $\boldsymbol{O}$, and the control set $L_2(\Sigma) = \{u\} = \{N, Z, P\}$. The control is also uncertain and we only know that

$u = N$, $-r < d\xi_1/dt < d\xi_2/dt < -l < 0$
$u = Z$, $d\xi_1/dt = d\xi_2/dt = 0$
$u = P$, $0 < l < d\xi_1/dt < d\xi_2/dt < r$

The goal state, which is also called the reference, is $[g_1, g_2]$.

□

Lemma 1. (*Realization of 1-D FFCS*) Given any one-dimensional fuzzy feedback control system with groupoid $L_2(\Sigma)^*$ of control words and e the required precision constraints, that is, e is the range that the observed final state is contained in $[g_1, g_2]$, there is a GFA automaton realizes it.

Proof:

Let at time t the support of the observed state is $[k_1(t), k_2(t)]$. We prove the lemma by constructing a GFA. Construct a GFA $\boldsymbol{M}(\delta, \Sigma, U, S_0, F, T)$ where

$U = \cup\ [\xi_1(t), \xi_2(t)\ ], \forall t$

$S_0=(\alpha(0), s(0), \beta(0), 1)$

The transition function δ is defined as follows. Let $k_1(t) = s(t) - \alpha(t)$, $k_2(t) = s(t) + \beta(t)$, and $\alpha(t)$, $\beta(t)$ are real crisp numbers such that $s(t) \in [k_1(t), k_2(t)] \subseteq U$. Without lost of generality, let $s(t) = \lambda k_1(t)+(1-\lambda)k_2(t)$ for some $\lambda\in(0, 1)$, $\tau_1 = (k_1-g_1)/r$, and $\tau_2 = (g_2-k_2)/r$. We have the transition function δ as the following fuzzy rules:

Rule 1:

IF $u = N$ AND $\boldsymbol{O}(S(t)) = (\alpha(t), s(t), \beta(t), 1)$ THEN τ_1 later $S(t+\tau_1) = (\alpha(t) - le/r, s(t)+\lambda le/r, \beta(t), 1)$.

Rule 2:

IF $u = P$ AND $\boldsymbol{O}(S(t)) = (\alpha(t), s(t), \beta(t), 1)$ THEN τ_2 later $S(t+\tau_2) = (\alpha(t), g_2-\beta(t), g_2, 1)$.

Rule 3:

IF $u = Z$ AND $\boldsymbol{O}(S(t)) = (\alpha(t), s(t), \beta(t), 1)$ THEN state $= (\alpha(t), s(t), \beta(t), 1)$.

The remaining components of $\boldsymbol{M}$ are

$\Sigma = \sigma(N\cup P\cup Z)$, universe of the input *words*,

$F = [g_1, g_2]$, and

$T = \{dt\}=\{\tau_1, \tau_2\}$.

Start from t, within duration dt, $\boldsymbol{M}$ fires state transition from $S(t)$ to $S(t+dt)$ to the matching degree μ (firing strength). We observe that $\boldsymbol{O}(S(t+dt)) = \boldsymbol{O}\cdot(\alpha(t+dt), s(t+dt), \beta(t+dt), \mu)$, the support of fuzzy set $\boldsymbol{O}(S(t+dt))$ becomes $[k_1(t+dt), k_2(t+dt)]$. Since

$$s(t+dt)-\alpha(t+dt) = k_1(t+dt) \geq \xi_1(t) \geq k_1(t) + le/r \tag{5}$$

and

$$s(t+dt)+\beta(t+dt) = k_2(t+dt) \leq \xi_2(t) \leq g_2, \tag{6}$$

The support of the observed state $[k_1(t+dt), k_2(t+dt)]$ is contained in the system state $[\xi_1(t), \xi_2(t)]$. Therefore, we have the input-state homomorphism of the *1-D FFCS*.

Q.E.D.

Lemma 1 reveals that the generic fuzzy feedback control system is "described" with words. The control u used in transition rules is generalized into $L_2(\Sigma)$ whose members are words. In GFA theory, the additional timing interval component T provides more flexible and accurate descriptions of the dynamics of uncertain FFCSs. The generalization of 1-D FFCSs into multi-dimensional ones is given in Definition 5.

Definition 5. (*Generalized FFCS*) A generalized n-D FFCS has control set a subset of $L_2(\Sigma^{(n)})$ whose members are L-fuzzy [9] sets and L is a complete lattice. For a member j of $L_2(\Sigma^{(n)})$ and $L = [0, 1]$, the membership function is a mapping

$$\mu_j : \Sigma^{(n)} = \prod_{i=1}^{n} \Sigma \to [0,1] \tag{7}$$

□

Note that the automaton used in Lemma 1 is an algebraic representation of fuzzy control system rather than of a controller. However, if a GFA automaton recognizes an uncertain system, the output function of the GFA is just the controller. The design of the feedback controller is an output function η: $U \rightarrow \Sigma$.

Theorem 1. (*Design of FFCS controllers*) Given a GFA $\boldsymbol{M}(\delta, \Sigma, U, S_0, F, T)$ realizing an FFCS, there exists a fuzzy rule base which is an output function of $\boldsymbol{M}$ such that $\boldsymbol{M}$ is attainable and the rule base acts as a controller of the FFCS.

Proof:

We can easily prove the theorem by constructing a fuzzy rule base that outputs control *words* corresponding to state fuzzy sets such that M is attainable. As feedback, the state space U of $\boldsymbol{M}$ is the domain of the output function η and the co-domain is Σ, that is, the rule base is a mapping η: $U \rightarrow \Sigma$. Suppose that the supports of a state fuzzy set $S(t)$ at time t and the final state fuzzy set F are $[k_1(t), k_2(t)]$ and $[g_1, g_2]$ respectively, then we construct the rule base η as follows:

Rule 1:

IF *error*($S(t)$) is *negative*

THEN u is $P=(\alpha_P, m_P, \beta_P, 1)$

Rule 2:

IF *error*($S(t)$) is *zero*

THEN u is $Z=(\alpha_Z, m_Z, \beta_Z, 1)$

Rule 3:

IF *error*($S(t)$) is *positive*

THEN u is $N=(\alpha_N, m_N, \beta_N, 1)$

, where $error(S(t)) = (k_1(t) + k_2(t) - g_1 - g_2)$, and P, Z, N are words in $L_2(\Sigma)$. The membership functions of the fuzzy words *negative*, *zero*, *positive*, P, Z, and N are:

$negative(s) = 1$ for $s < 0$;

$zero(s) = 1$ for $s = 0$;

$positive(s) = 1$ for $s > 0$;

$P(u) = 1$ iff $u = m_P$;

$Z(u) = 1$ iff $u = m_Z$;

$N(u) = 1$ iff $u = m_N$.

According to the rule base, when the support of $\boldsymbol{O}(S(t))$ is $[k_1(t), k_2(t)]$ with $k_1(t) < g_1$ and $k_2(t) < g_2$, such that *error*($S(t)$) is *negative*, by applying positive u with degree $\mu = negative(error(S(t)))$ during time interval $[t, t + dt]$ with $dt = (g_2-k_2(\mathrm{t}))/r$, the support of $\boldsymbol{O}(S(t+dt))$ becomes $[k_1(t)+le/r, k_2(t)]$. When the support of observed state $S(t)$, $\boldsymbol{O}(S(t))$, is $[k_1(t), k_2(t)]$ with $k_1(t) > g_1$ and $k_2(t) > g_2$, such that $error(S(t))$ is *positive*, by applying negative u with degree $\mu = positive(error(S(t)))$ during time interval $[t, t + dt]$ with $dt=(k_1(t)-g_1)/r$ the support of $\boldsymbol{O}(S(t+dt))$ becomes $[k_1(t), g_2]$. According to Lemma 1 and (5)(6), the support of $S(t)$ will fall in F, that is, the rule base of the controller makes the FFCS attains the goal.

Q.E.D.

In multidimensional cases, control base variables u's and state base variables s's are vectors. The description of multidimensional FFCSs and design of respective controllers can also be developed similar to Lemma 1 and Theorem 1. One important property concluded from Theorem 1 is that if a FFCS is properly described with words (with a GFA) under constraints – the maximum state moving rates l and r, and the precision constraint e, a stable fuzzy feedback controller is designed. In other words, if l, r, and e are learned from input-output observations, a corresponding controller with a very simple rule-base can be automatically generated.

4. Examples

In this section two examples show how a control system is "recognized" and described by respective GFAs. In the second example, a non-linear control problem is given and we demonstrate how a controller for this plant is designed according to Lemma 1 and Theorem 1. We do not concentrate on performances of the controller speed and error issues but try to demonstrate how difficult control problems are easily modeled with the GFA.

4.1. Example 1 -- Fuzzy Temporal Knowledge System

We take the first example the same as in articles [7][8]. The example is a complex fuzzy rule: "If the holdup in the buffer drum increases (P_1), and about one minute later the reactor pressure decreases (P_2) and the regenerator temperature increases (P_3), and the regenerator pressure and reactor temperature all decrease about two minutes later than the holdup in the buffer drum increases (P_4 and P_5), then a regenerator slide valve closes about one minute earlier than the holdup in the buffer increases, with confidence 0.95 (P_6)." From the statements with the propositions $P_1, P_2, \ldots, P_6$, we rewrite the rule

IF P_1 AND P_2*dT_1 AND P_3*dT_1 AND P_4*dT_2 AND P_5*dT_2 THEN P_6*dT_3

where "*" is the $\vee\wedge$ composition. Construct GFA $\boldsymbol{M}(\delta, \Sigma, U, S_0, \{f\}, T)$ where U = union of supports of S_0, S_1, S_2, S_3, and f, $\delta(P_4\wedge P_5, S_3, dT_2) = f$, $\delta(P_2\wedge P_3, S_2, dT_1) = S_3$, $\delta(P_1, S_1, dT_1) = S_2$, $\delta(P_6, S_0, dT_3) = S_1$, $L_2(\Sigma) = \{P_1, P_2\wedge P_3, P_4\wedge P_5, P_6\}$, F = support of f and $T = \{dT_1, dT_2, dT_3\}$. In this case, T is the $L_2(t)$ and its members are fuzzy numbers dT_1="about one minute," dT_2 = "about two minutes", and dT_3 = -"about one minute". The observation bound is 0.95. Therefore, GFA $\boldsymbol{M}$ realizes the fuzzy temporal knowledge system. Next step to design a FFCS controller, we are required to design a fuzzy rule base with conclusion parts the propositions in $L_2(\Sigma)$.

4.2. Example 2 -- Controller Design for Non-linear Plant

We use the plant in [6] as the second example. The relationship of plant's output y and input u is represented by the equation $y'' + y' + \ln y = u$. There is a design flow to design a controller for this plant according Lemma 1 and Theorem 1. First, as Fig. 1, we use a simple square wave with amplitudes minimum –1 and maximum +1, and observe the corresponding output slope for every pulse of the square wave. The maximum slope of the output defines r and the minimum one defines l. For stability, the time step size of the controller is chosen smaller than e/r. As Fig. 2, the upper curve is the output by applying the lower curve (square wave). Then, values for $l = 0.5$, and $r = -1.125$ are then respectively measured from the maximum and minimum changing rate of the output curve and the step timing e/r are set $0.008 \approx 0.01/1.125$. In "words," we describe the FFCS as follows: if control is –1, the output will drop down at rate no more than r while if control is +1, the output will turn to increase at rate no lower than l. According to Lemma 1, we then define level 2 fuzzy set $L_2(\Sigma) = \{\sim+1, \sim0, \sim-1\}$ of fuzzy numbers with (α, m, β) parameter triplexes (0.5, +1, 0.5), (0.5, 0, 0.5), and (0.5, -1, 0.5). These are fuzzy words adopted as conclusions in the output function η, a controller's fuzzy rule base. Similar to the ones defined in the proof of Theorem 1, the premise parts' membership functions are defined as

$negative(s) = 1$ iff $error < -0.001$;

$zero(s) = 1$ iff $error = 0$;

$positive(s) = 1$ iff $error > +0.001$;

Consequently, the fuzzy rule base that is very simple is as Fig. 3. The left side is the premise part while the right part is the conclusion part. The implication method adopts minimum operation. The aggregation method adopts maximum operation. Defuzzification uses centroid calculation. Then, we have the whole FFCS as Fig. 4. The reference is a time-varying signal, which is a sum of DC offset 1.6 and AC sine wave. There are two scopes for observing the simulation results. The one "Scope u" is for observing control signal while "Scope for state S(t) and goal" is for the reference signal and output of the FFCS system. The simulation result is given in Fig. 5(a), where one can see that at first the output of the system (lower curve) and the reference (upper curve) are distinguishable while some seconds later, they are almost overlapped with each other, in other words, the control sequence of words attains the goal. From Fig. 5(b), initially the output of the controller is at full level +1 while after that the system output tracks the reference variation; the control output goes with value fall in [-0.8, +0.8].

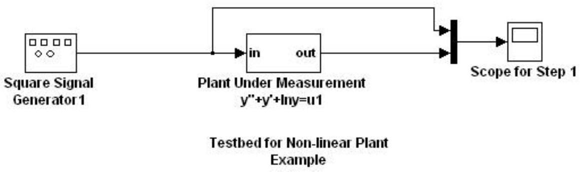

Fig. 1. Observing behavior of the plant.

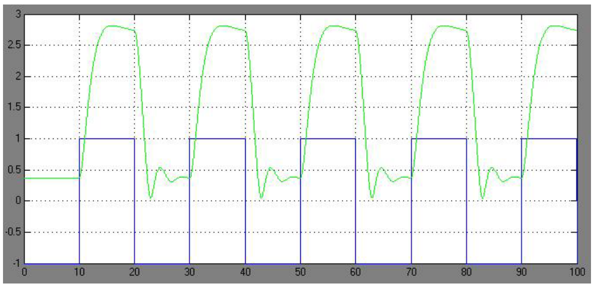

Fig. 2. The square wave (lower) as test pattern and the output responding curve (upper) of the nonlinear plant.

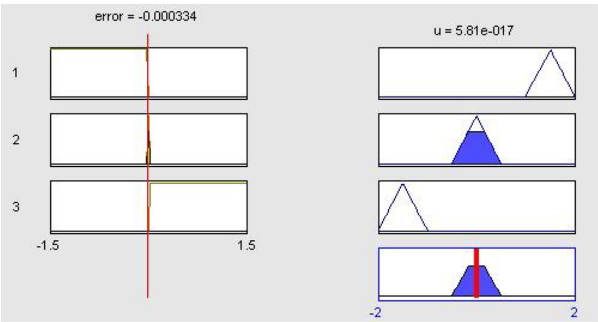

Fig. 3. Snapshot of the fuzzy rule base executing inference for producing control signal.

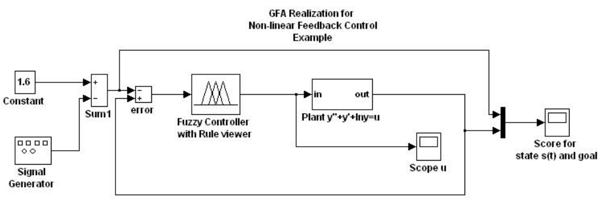

Fig. 4. The whole FFCS system with two reference sources and scopes.

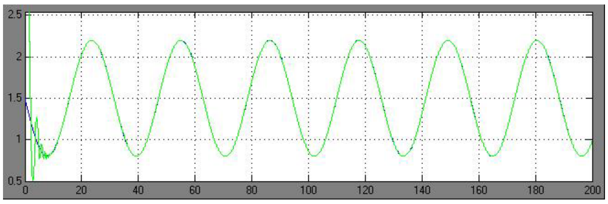

Fig. 5 (a)

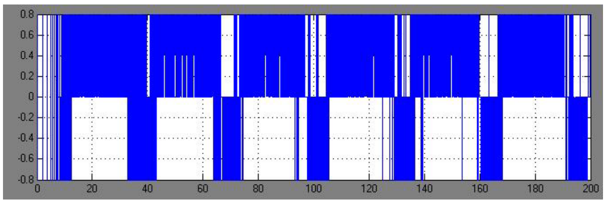

Fig. 5 (b)

Fig. 5. (a) The output signal of the system (initially the lower one) and the reference signal (initially the upper one.) and (b) Output control of the controller.

From the simulations above, if a measurement of a plant/environment is properly described with a GFA automaton, even without much domain knowledge a well performance controller for a nonlinear FFCS is implemented with a very simple fuzzy rule base.

5. Conclusion

In this paper, we develop generalized fuzzy automata (GFA) model for fuzzy feedback control systems (FFCS). The control system is described and controlled with words. The set of sequences of control words is a groupoid over level 2 fuzzy set. The descriptions of the FFCS are the transition rules of the GFA. According to the GFA, we can then define its output function as the feedback controller. Mathematically we prove that GFA realize general fuzzy feedback controls. This novelty provides implementation, no matter in hardware or software, of fuzzy feedback controllers in a systematic, unified, and effective paradigm. The proofs also demonstrate how fuzzy feedback control with words is performed. There exist generic and simple rule base for fuzzy feedback control problems. Examples show that control with words based on GFA theory is feasible. The future works include studies of language properties and more applications of the GFA theory. The applications such as learning algorithms based on GFA for automatic controller generation and generic *word* model for optimum control are to be studied.

Acknowledgment

The author thanks National Science Council in Taiwan R.O.C for supporting this research, which is part of the research project numbered NSC 92-2213-E-168-006-.

References

[1] Sheldon S. L. Chang and L. A. Zadeh, “On fuzzy mapping and control,” *IEEE Tran. Syst., Man, and Cybernet.*, Vol. 2, No. 1, Jan. 1972, pp. 30-34.

[2] Chao-Lieh Chen, “Programmable fuzzy logic device for sequential fuzzy logic synthesis,” *The 10th IEEE International Conference on Fuzzy Systems*, 2001, Vol.1, 2-5 Dec. 2001, pp.107 - 110.

[3] Ehrig H, Kiermeier K.D., Kreowski H.J., Kuehnel W. *Universal Theory of Automata: A Categorical Approach*, B.G. Teubner, Stuttgard.

[4] J. Goguen. "Realization is universal," Mathematical Systems Theory, Vol. 6, 1972, pp. 359-374.

[5] J. Močkoř, “A category of fuzzy automata,” *Int. J. General Syst.*, Vol. 20, pp. 73-82.

[6] Imre J. Rudas and M. Okyay Kaynak, “Entropy-Based Operations on Fuzzy Sets,” *IEEE Trans. Fuzzy Syst*. Vol. 6, No. 1, Feb. 1998, pp. 33-40.

[7] D. Qian, “Representation and use of imprecise temporal knowledge in dynamic systems,” *Fuzzy Sets and Systems*, 50(1992), pp. 59-77.

[8] J. Virant and N. Zimic, “Attention to time in fuzzy Logic,” *Fuzzy Sets and Systems*, 82(1996), pp. 39-49.

[9] J. A. Goguen, “L-fuzzy set,” Journal of Mathematical Analysis and Applications, vol. 18, 1967, pp.145-174.