**Proceedings 2003 IEEE International Symposium on
Computational Intelligence in Robotics and Automation
July 16-20, 2003, Kobe, Japan**

Progressive Fuzzy Fusion Control of Two Coupled Inverted Penduli

Zhen Song, Pranav Sukthankar, YangQuan Chen and Jason Gu

Abstract— **This paper considers the control of two inverted penduli coupled with a spring using a progressive fuzzy fusion control method. The two penduli have some uncertain and unknown parameters and the coupling spring has an unknown spring constant. We assume that the lengths of the penduli and the spring are known. One of the inverted penduli is excited to generate the force disturbance to the other through the spring link. The aim of this paper is to design a controller based on the idea of fuzzy logic and the classical control methods, such that the second pendulum can reject the disturbance from the first pendulum significantly even without precise mathematical model of the system. The angular position and the angular speed for both inverted penduli are measured to feed into the fuzzy fusion controller via the proposed disturbance decoupling procedure. Based on the hardware-in-the-loop realtime simulation environment provided by MATLAB/Simulink RTW (Real-Time Workshop) and plants in Quanser's Rotary Servo Family, the controller tuning is based on the fusion of the objective and subjective knowledge about this system in a progressive way. Extensive experimental results are reported to demonstrate how to effectively benefit from the fuzzy logic control.**

Keywords— **Fuzzy logic parameter tuning, disturbance decoupling, coupled inverted penduli, fuzzy fusion.**

I. Introduction

This paper considers a system that comprises two rotary inverted penduli coupled with a spring. Figure 1 shows the system configuration. If there are disturbances on pendulum 1, these disturbances will be conducted through the spring to pendulum 2. We want to develop a series of control strategies that can reject the disturbances on the second pendulum. This configuration is very similar to the system considered in [4], [5]. In these two references, the similar system was controlled by robust decentralized control approaches. These solutions are model based strategies, which are not applicable to our problem. For our problem, the basic motivation is to solve the problem in engineering approaches. Since some system parameters, such as the spring constant and the inertia of the motors, are not easy to measure, we assume that we only know the structure of the system model plus some parameters easy to measure, which will be discussed in details later.

This problem is interesting because firstly, we have four sensors that can measure the angle and the angular speed, but there is only one output of the controller system for the disturbance rejection where the "sensor fusion" should be applied to make the most use of the information we have; secondly, from the engineers' prospect, we want the control strategy to be progressive, i.e., its performance can be progressively improved without great modification on the control system structure; thirdly, to achieve the former two requirements, the control system will be relative complex which calls for a systematic approach for control analysis and parameter tuning; fourthly, the spring is a coupling factor among the states of the system. The disturbance rejection is also called "disturbance decoupling" in this paper. This decoupling problem is useful in other applications, such as the autonomous ground vehicle controller by fuzzy logic [7]. It is expected that the fuzzy decoupling strategies in this paper could be useful for other decoupling applications.

Zhen Song, Pranav Sukthankar and YangQuan Chen are with the Center for Self-Organizing and Intelligent Systems (CSOIS), Dept. of Electrical and Computer Engineering, 4160 Old Main Hill, Utah State University, Logan, UT 84322-4160, USA. Jason Gu is with the Dept. of Electrical and Computer Engineering, Dalhousie University, Halifax, Nova Scotia, Canada B3J 2X4. Corresponding author: Dr YangQuan Chen. E-mail: yqchen@ece.usu.edu; Tel. 01-435-7970148; Fax: 01-435-7973054. URL: http://www.csois.usu.edu/people/yqchen

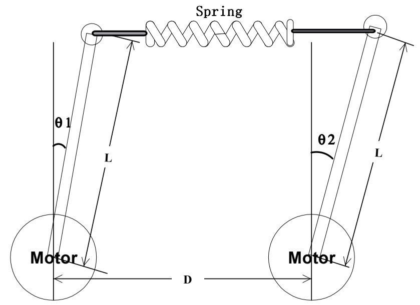

Fig. 1. The configuration of the coupled inverted penduli System.

Section II will state the problem, describe our assumptions, explain our test bed, and analysis the system. Section III will propose a systematical and progressive analysis approach called "iterative binary decomposition." Follow it, we will design the controller step by step. In Section IV, some of our experiment results are presented and in Section V conclusions are drawn on the experiment results.

II. Problem Statement and Analysis

A. Basic Problem

Consider the system shown in Fig. 1. The object is to design FLCs to reject the disturbance on pendulum 2, we have the following assumption, which are likely to be true for engineering applications:

- The features of the disturbances on pendulum1 are unknown.
- We have sensors to measure θ_1, θ_2, $\dot{\theta}_1$ and $\dot{\theta}_2$.
- Among all the parameters of the system, we can only measure L and D. Any other parameters of the system, such as the stiffness of the spring or the inertia of the motor etc., are unknown.

- The designer does not have any experience on the system, i.e., the designer can not write down the control law in natural language with "if-then" statements at the beginning. We expect the designer learn the control law during the tuning procedure.
- To accommodate the designer, the procedure should be progressive, such that the designer does not abandon the current controller framework and start over. Instead, he can just tune the parameters of the exist FLC, and add more FLC if necessary. He should not eliminate any of the existing FLC.
- Fuse the angle and the angular speed information in the framework of fuzzy logic, and, of course, get better performances than without fusion.
- No advanced controller design skills are required. As we stated, the problem is engineering oriented. Any one who knows differential, then fundamental physics and basic fuzzy logic theory should be able to solve this problem.

B. Experiment Platform

Referring to Fig. 1, the system has two rotary inverted penduli, which are coupled by a spring. In this paper, we will refer the pendulum on the right as pendulum 2, and the left one as pendulum 1. Their rotary angles are θ_1 and θ_2, respectively. At the bottom of the penduli, there are gears that connected to the gears of DC motors. The photo of our test bed is shown in Fig. 2. The pendulum 2 is on the right, and the pendulum 1 is on the left.

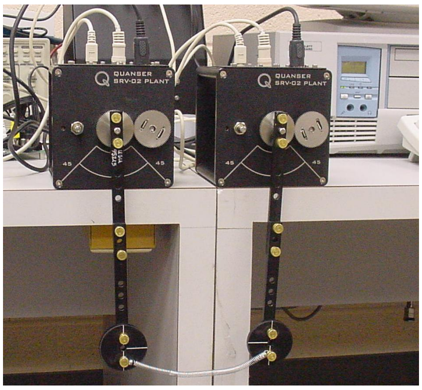

Fig. 2. A picture of our test bed.

The length of the penduli are L, and the distance between the two axes of actuating motors is D. The relaxed length of the spring is assumed to be the same as D.

Both of the two motors are controlled by Matlab and Quanser kits. The model in Matlab Simulink is shown in Fig. 3. The `Signal Generator` in Fig. 3 can make 3 types of signals: sine, square, saw waves, as the reference inputs to the control system of pendulum 1. These signals are periodic. We also use a pseudo random disturbance signal to check the performance when the disturbance is not periodic. When we use this pseudo random signal, we replace the `signal generator` block in Fig. 3 with a C S-Function. The pseudo random input is:

$$\begin{aligned}\sin(\phi + \text{sawsig}(t/3.5/K) * t + \text{sqsig}(t/3/K) * t) + \\ \text{sawsig}(\phi + \text{sqrt}(t/K) * 3.14) * 0.3 + \\ 0.3 * \text{sqsig}(\phi + t/K), \qquad (1)\end{aligned}$$

where $K = 2.2$, $\phi =$ `rand()%10000`, `sqsig()` and `sawsig()` are the functions to generate square and saw wave signals, respectively. As shown in Fig. 4, the frequency of this pseudo random signal is about 1~ 5 Hz. We use this low frequency signal, instead of pure white noise with high frequency components, in order to protect our testing hardware.

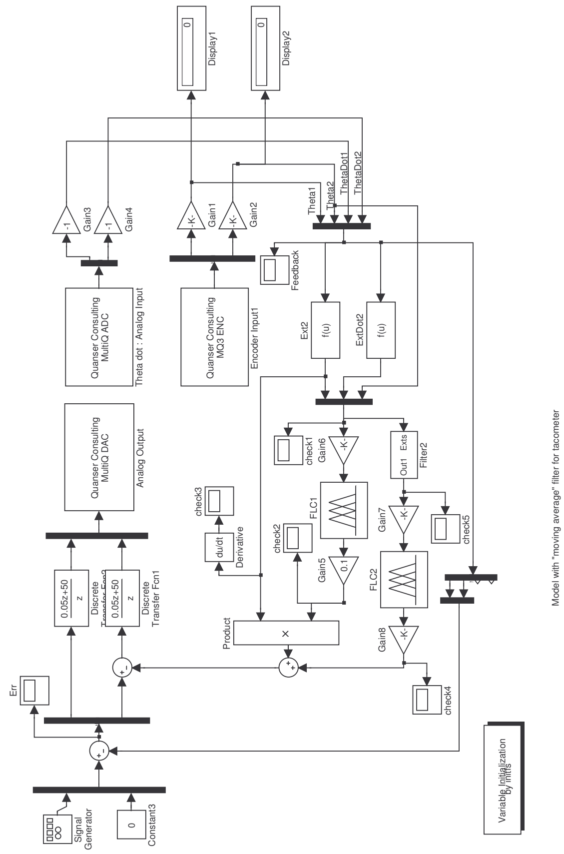

Fig. 3. Coupled inverted penduli system model.

We will refer these inputs as "disturbances generator" in this paper since the actual disturbance from pendulum 1 to pendulum 2 via the spring link is due to the above defined input signal. For each pendulum, we have an incremental optical encoder and one potentiometer connected on the pendulum's joint. The resolution of the encoder is 4096 counts per revolution. The two motor shafts drive two anti-backlash gears to the gears that connect the inverted penduli. If correctly installed, the anti-backlash gears can greatly reduce the side effect of the backlash nonlinearities between gears, and improve the system performance. During the experiment, the four sensors will measure θ_1, θ_2, $\dot{\theta}_1$ and $\dot{\theta}_2$ as the indirect inputs to the controller of pendulum

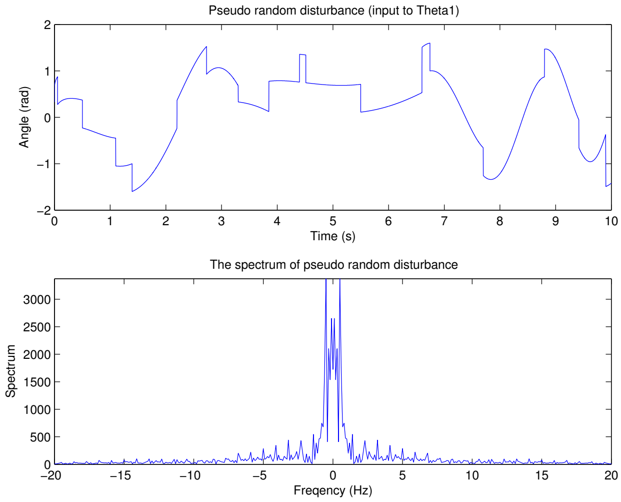

Fig. 4. Pseudo random disturbance.

2. To design a controller for pendulum 2, no knowledge on the disturbance signal of pendulum 1 is assumed to be known.

Before each experiment, we held the penduli at the exact upward positions, and then started to download the executable code from within Matlab RTW. The initial angle of the penduli is 0, and positive angles are considered as in the clockwise direction. When we apply positive signals to DC motors, the motors rotate in the clockwise direction.

C. A Simple System Model

Simply using a ruler, we can measure D and L. With this limited knowledge, we can setup a model as the following. The coordinates of the top of pendulum 1 are assumed to be that

$$P_{m1} = \begin{bmatrix} L\sin(\theta_1) \\ L(1-\cos(\theta_1)) \end{bmatrix}, \qquad (2)$$

and the top of pendulum 2 is that

$$P_{m2} = \begin{bmatrix} D + L\sin(\theta_2) \\ L(1-\cos(\theta_2)) \end{bmatrix}. \qquad (3)$$

If the relaxed length of the spring is D, the extension of the spring is

$$E_{xt} = -D + \big((L\sin(\theta_1) - D - L\sin(\theta_2))^2 + (L\cos(\theta_1) - L\cos(\theta_2))^2\big)^{1/2}, \qquad (4)$$

and the derivative of the extension is

$$\dot{E}_{xt} = \Big(DL\cos(\theta_1)\dot{\theta}_1 - DL\cos(\theta_2)\dot{\theta}_2 + L^2\sin(\theta_1-\theta_2)(\dot{\theta}_1-\dot{\theta}_2)\Big)/(E_{xt}+D). \qquad (5)$$

III. Progressive Fuzzy Fusion Controller Design

A. Iterative Binary Decomposition

The proposed iterative binary decomposition approach is a tool for systematic analysis of the subjective and objective which is explained in Fig. 5. We can decompose the system into a more subjective part and a more objective part at each level, and analyze the system level by level, iteratively. In Fig. 5, the "(obj)" and "(sub)" represent a part which is relatively more objective and the more subjective, respectively. The decoupling strategy of FLC is relatively subjective. However, we can still decompose it into objective and subjective parts. The following spring static property:

$$F = KE_{xt} \qquad (6)$$

is well known and it is objective. Here, the E_{xt} is the extension of the spring in (4). But the coupling dynamics, i.e., the relationship between the angle and the angular speed, is not clearly known as we observed from the experiments. It is considered more subjective.

According to the assumptions in Sec. II,

We could do a comprehensive analysis on the coupling dynamics, then start working on the real system. However, we think a more efficient approach for this practical project is to use the "trial-and-error" method: run the system, record the performance, do simple theoretical analysis, tune the parameters accordingly, and run the system again iteratively. After we accumulated enough knowledge on the coupling dynamics, we can decompose it into subjective and objective parts. By this way, we can "squeeze" more objective from the subjective part. The details will be given in subsection III-D.

In order to be able to improve the performance by more and more precise decomposition, the two parts in each decomposition should have no or less physical coupling. There is, however, no exact way to perform the exact decomposition. The decomposition task should be application-specific.

The advantages of this approach are:

- It is a systematic way for FLC designing and parameter tuning. If the designer can not explain the control law in natural language, this approach can help him to analysis the problem, and proposed the control law. It is different from the adaptive fuzzy logic controller, such as the fuzzy sliding mode controller in [8] which can adjust parameters itself. This approach is analytical method, rather than an architecture for implementation.
- It is a "trial-and-error" approach. So it does not require the designer have sound control background and full knowledge of the system. They can improve the understanding about the system as well as its performance gradually, and stop at a "good enough" stage.
- The designers can improve the performance progressively without significant modification on the system architecture. As far as there is no mistake in the objective parts, they do not need to be modified. The designers can follow the above example, and "squeeze" objective from subjective part in each iteration of this approach. The more objective part in the controller, the better performance we can expect.

- Coupled inverted penduli
  - PI controller(obj)
    - PI tuning(obj)
    - PI and FLC cooperative tuning(sub)
  - Decoupling stragety(sub)
    - Spring static properties(obj)
    - Coupling dynamics(sub)

Fig. 5. Decomposition for subjective and objective

B. Balance The Subjective and Objective by The Concept of Freedom

Our choice on the options is subjective if the consequence is unknown, while the decision on the options with a known consequence is called objective. Subsection III-A gave an example on how to analyze the subjective and the objective part in a practical system. In this subsection, we will focus on the FLC design procedure and discuss how to balance the subjective and the objective by "freedom."

The most important philosophy for the FLC design is the balance between subjective and objective. We need to define a concept called *true freedom*, and maximize it in a certain level of the iterative binary decomposition. In Merrian Webster Dictionary [1], the word "freedom" is defined as *the absence of necessity, coercion, or constraint in choice or action.* This definition does not answer this question: if the constraint of an option is not clearly understood, and an improper choice will lead to punishments, should this option be considered as "freedom?" In this paper, we will still refer it as "freedom," since we do not have any restriction at the time when we make any decision. We will, however, refer it as the "counterfeit freedom," because the limitation of the option will show up after the decision making, and the "over-constraint" option will be punished at last. In a sense, this implicit constraint is even worse than the explicit constraint, because the first one eliminates the freedom of avoiding the punishment. A naive engineer might believe the degree of true freedom is proportional to the number of options. If this idea was tried on a real project, soon we will suffer from the "curse of the degrees of freedom," i.e., we will suffer from the punishments from the counterfeit freedoms. In this article, the counterfeit freedom is defined as an option without a clearly known consequence that might lead to an undesirable result. It is basically a "gamble on bad or good" problem. The true freedom is defined as an option with well known consequence. It is a "avoiding bad" or "choose the better from good" problem.

Note, the true and counterfeit freedoms are always associated with certain level of objective-subjective decomposition. In Fig. 5, if we just consider the first level of decomposition, the PI controller tuning is objective, because there are many references discussed the tuning of PID type controllers [2], [3], [6], and the consequences for parametric options are well known. Thus, the two parameters of each PI controller are considered as the contributions to the true freedoms. On the other hand, if we use FLC to stabilize the penduli, we think the true freedoms reduce while the counterfeit freedoms increase. Although each FLC has more parameters than one PI controller, the additional options did not add more true freedoms. On the contrary, since the relationships among parameters and the stability are not understood as well as the PID controller, the additional options from FLC contribute only to the counterfeit freedoms.

C. PI Controllers

At the beginning, we did not use the spring to connect the 2 inverted penduli, and the gains for the two FLC in the Simulink model (Fig. 3) are 0. Then, we use the PI controllers to stabilize the 2 penduli one by one.

D. Fuzzy Controllers

According to the understanding on the characteristics of the coupling, as mentioned in equations 4 and 5, we propose static and dynamic decoupling approaches to reject the disturbances on the inverted penduli system.

D.1 Static Decoupling

The static decoupling is an approach to decouple the spring by considering the information of extension and angle only. Since we did not put the derivative of the extension into this FLC, i.e., `FLC1` in Fig. 3, the controller did not consider the coupling dynamics. The control law for the static decoupling is straightforward. The block for the dynamic decoupling controller is `FLC2` in the Fig. 3, but the `FLC2` is turned off at the static decoupling stage by setting `Gain8` to 0. Table I summarizes the fuzzy rule base of `FLC1` and `FLC2`, respectively, with the entries format `(Mp, Ad)` explained in the figure caption.

The main idea of the static decoupling is simple. We can use (4) and (6) to estimate the tension force on the spring. Then, we multiply this observed force with the output of the FLC as the compensation to the PI controller. In our experiments, we used `Gain5` and `Gain6` in Fig. 3 to scale the compensation.

When setting up the fuzzy rule base in Table I, we also need to consider the expression of the fuzzy rules in the Matlab fuzzy toolbox. Figure 6 gives an example of how we combined several rows in Table I to one fuzzy rule. Instead of making one fuzzy rule for each entry in Table I, we can use one fuzzy rule to represent many entries, since the static decoupling rules are row dependent and the dynamic decoupling rules are column dependent. Thus, we can use 15 rules in Figs. 6 and 7 to represent the total 75 entries in Table I.

TABLE I

FUZZY RULE BASE FOR TWO INVERTED PENDULI DECOUPLING

θ_2	Ext \ $\dot{\text{Ext}}$	-2	-1	0	1	2
P	-2	(-1,*)	(-1,*)	(-1,*)	(-1,*)	(-1,*)
P	-1	(-1,*)	(-1,*)	(-1,*)	(-1,*)	(-1,*)
P	0	(0,*)	(0,*)	(0,*)	(0,*)	(0,*)
P	1	(2,*)	(2,*)	(2,*)	(2,*)	(2,*)
P	2	(2,*)	(2,*)	(2,*)	(2,*)	(2,*)
N	-2	(0,*)	(0,*)	(0,*)	(0,*)	(0,*)
N	-1	(0,*)	(0,*)	(0,*)	(0,*)	(0,*)
N	0	(0,*)	(0,*)	(0,*)	(0,*)	(0,*)
N	1	(1,*)	(1,*)	(1,*)	(1,*)	(1,*)
N	2	(2,*)	(2,*)	(2,*)	(2,*)	(2,*)
Z	-2	(0,*)	(0,*)	(0,*)	(0,*)	(0,*)
Z	-1	(0,*)	(0,*)	(0,*)	(0,*)	(0,*)
Z	0	(0,2)	(0,1)	(0,0)	(0,-1)	(0,-2)
Z	1	(1,*)	(1,*)	(1,*)	(1,*)	(1,*)
Z	2	(2,*)	(2,*)	(2,*)	(2,*)	(2,*)

The entries in this table follow the format of `(Mp,Ad)` where `Ad` is the output of the `FLC1` block in Fig. 3 , and `Mp` is the output of the `FLC2` block in Fig. 3. The symbol * means we did not use fuzzy rules to force the output to be a specific value.

1. if Ext=-2 and θ_2=P then Mp=-1
2. if Ext=-1 and θ_2=P then Mp=-1
3. if Ext=1 and θ_2=P then Mp=2
4. if Ext=2 and θ_2=P then Mp=2
5. if Ext=1 and $\theta_2 \neq$P then Mp=1
6. if Ext=2 and $\theta_2 \neq$P then Mp=2
7. if Ext=-1 and $\theta_2 \neq$P then Mp=0
8. if Ext=0 and $\theta_2 \neq$P then Mp=0
9. if Ext=-2 and $\theta_2 \neq$P then Mp=0
10. if Ext=0 and θ_2 =P then Mp=0

Fig. 6. Fuzzy rules for static decoupling.

1. if Ext=0 and $\dot{\text{Ext}}$=-2 and θ_2=0 then Ad=2
2. if Ext=0 and $\dot{\text{Ext}}$=-1 and θ_2=0 then Ad=1
3. if Ext=0 and $\dot{\text{Ext}}$=1 and θ_2=0 then Ad=-1
4. if Ext=0 and $\dot{\text{Ext}}$=2 and θ_2=0 then Ad=-2
5. if Ext=0 and $\dot{\text{Ext}}$=0 and θ_2=0 then Ad=0

Fig. 7. Fuzzy rules for dynamic decoupling.

D.2 Dynamic Decoupling

After the static decoupling fuzzy logics were implemented, the system performance is analyzed as plotted in Fig. 8. In order to reduce the magnitude and the ITAE per cycle of θ_2, we basically need to start the compensation before the force is applied to the pendulum 2. This is because the system dynamics of the driving motor and the pendulum will introduce an unavoidable delay. Following this idea, we actually just use `FLC2` in Fig. 3 to set the `Ad` values in 5 entries of Table I. The fuzzy rules in `FLC2` did not control the `Ad` for the rest of the entries, since the experiment shows that we can only consider the system dynamics when θ_2 is near 0. Here, the ITAE is defined as

$$ITAE = \int_0^T t|\theta_2(t)|\mathrm{d}t, \qquad (7)$$

where T is the ending times of a trial. In the experiment, we measured the ITAE of 10s. ITAE, together with the magnitude of θ_2, represent the disturbance rejection performance for pendulum 2.

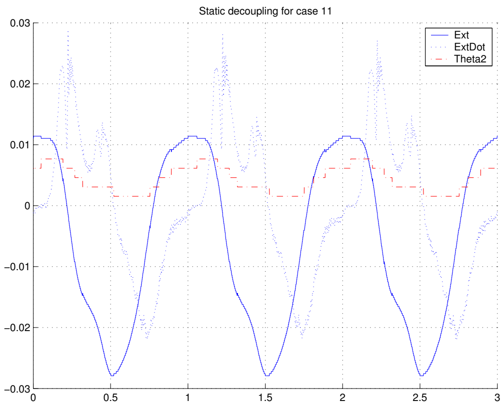

Fig. 8. The extension, its derivative, and θ_2 for the static decoupling.

TABLE II

EXPERIMENT RESULTS WITH PI CONTROLLERS ONLY

Code	Freq. Dist.	Amp. Dist.	Dist. Signal	Amp. θ_2	ITAE per Trial
11	1	5	Sin	0.0215	433.89
12	1	10	Sin	0.0568	934.22
21	2	5	Sin	0.0660	931.71
22	2	10	Sin	0.2684	1976.68
31	3	5	Sin	0.0951	1475.32
41	3	5	Square	0.1396	1448.37
61	1	5	Saw wave	0.0859	1048.96

IV. EXPERIMENT RESULTS

The the results of the static and dynamic decoupling are shown in Figs. 8 and 9, respectively. The above two plots are just figures associated with case 11 in Tables II,III, and IV.

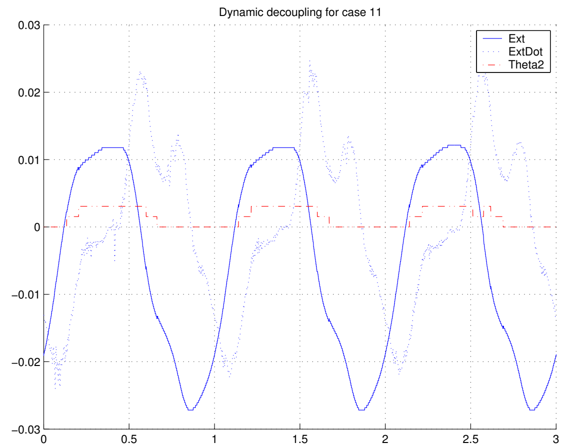

Fig. 9. The extension, its derivative, and θ_2 for the dynamic decoupling.

After tested on all the cases in the above three tables, the results are listed in Tables II, III, IV, V, and VI. In Tables II, III, IV, `Freq` is the frequency in Hz. `Dist` is the disturbance. `Amp` is the amplitude in radian. In these tables, together with Table VI, ITAE is computed based on the 10 sec. testing data. It is easy to see that no matter the disturbance is periodic or not, our static decoupling FLC's can significantly reject the disturbance, and the dynamic decoupling fusion FLC's have even better performance than the static decoupling.

TABLE III
Experiment Results with Static Decoupling

Code	Freq. Dist.	Amp. Dist.	Dist. Signal	Amp. θ_2	ITAE per Trial
11	1	5	Sin	0.0061	134.25
12	1	10	Sin	0.0077	74.60
21	2	5	Sin	0.0077	72.33
22	2	10	Sin	0.0169	117.73
31	3	5	Sin	0.0153	84.50
41	3	5	Square	0.0184	125.96
61	1	5	Saw wave	0.0522	357.79

TABLE IV
Experiment Results with Dynamic Decoupling

Code	Freq. Dist.	Amp. Dist.	Dist. Signal	Amp. θ_2	ITAE per Trial
11	1	5	Sin	0.0031	43.79
12	1	10	Sin	0.0092	51.20
21	2	5	Sin	0.0061	60.41
22	2	10	Sin	0.0138	115.16
31	3	5	Sin	0.0092	61.79
41	3	5	Square	0.0153	116.76
61	1	5	Saw wave	0.0399	272.38

TABLE V
Comparison on the Performances of the PI Only, Static, and Dynamic Decoupling Cases

Item	PI	Static	Dynamic
Amplitude	0.1047	0.0178	0.0138
ITAE	1178.45	138.16	103.07

TABLE VI
Experiment Results with Pseudo Random Disturbance

Controller	Amp. θ_2	ITAE per Trial
PI	0.1611	1220.7416
Static Decoupling	0.0767	609.4851
Dynamic Decoupling	0.0706	575.1869

V. Conclusion

In this project, the progressive fuzzy fusion control of two inverted penduli coupled by a spring was implemented. We successfully used fuzzy logic with fused information from different sensors and different FLCs, and rejected disturbances from one of the penduli. During this experiment, we verified our fuzzy controller tuning strategies, and demonstrated how to apply it to the real system. We hope this example can help engineers to design fuzzy controllers by a systematic and progressive approach with our proposed "iterative binary decomposition" strategies. Although we did not use any mathematic formula to describe this approach, it has sound philosophic background, and could be a very easy and powerful tool for some real projects.

The video clips of this work can be downloaded from `http://scontrol.yeah.net` or `http://mechatronics.ece.usu.edu/ieee_cira03/`.

References

[1] Merrian webster dictionary (online). http://www.m-w.com.
[2] K. J. Astrom and T. Hagglund. *PID Controllers: Theory, Design, and Tuning.* SA - The Instrumentation, Systems, and Automation Society, 2 edition, Jan. 1995.
[3] A. Datta, M. Ho, and S. P. Bhattacharyya. *Structure and Synthesis of PID Controllers.* Springer Verlag, 1 edition, May 2000.
[4] D. T. Gavel and D. D. Siljak. Decentralized adaptive control: Structural conditions for stability. *IEEE Tran. Automatic Control*, 34(4):413–426, Apr. 1989.
[5] Z. Gong, C. Wen, and D. P. Mital. Decentralized robust controller design for a class of interconnected uncertain systems: with unknown bound of uncertainty. *IEEE Tran. Automatic Control*, 41(6):850–854, June 1996.
[6] K. Ogata. *Modern Control Engineering.* Prentice Hall, 4 edition, 2001.
[7] K. R. S. Kodagoda, W. S. Wijessoma, and E. K. Teoh. Fuzzy speed and stering control of an AGV. *IEEE Tran. on control system technology*, 10(1):112–120, Jan. 2002. decoupling steering from speed control.
[8] L. K. Wong, F. H. F. Leung, and P. K. S. Tam. A fuzzy sliding controller for nonlinear systems. *IEEE Tran. on industrial electronics*, 48(1):32–37, Feb. 2001.