Analysis of Different Mediation Equations and Tightness of Control to Finely Regulate the Exchange of Control Between Expert and Novice Controllers in a Fuzzy Mediation Environment

Giovanni Vincenti
Research & Development
Gruppo Vincenti, S.r.l.
Rome, Italy
giovanni.vincenti@gmail.com

Goran Trajkovski
Dept. of Information Technology
South University
Savannah, GA
gtrajkovski@southuniversity.edu

Abstract

Fuzzy mediation is an innovative approach to the creation of a framework geared towards supervised and collaborative learning in systems with two controllers, one being an expert controller and the second being a novice one. The nature of fuzzy sets allows for the comparison of inputs to reach a consensus on the overall difference between the controls. In previous works we have highlighted the importance of this concept and the aptitude and ease with which this framework adapts itself to agents based on fuzzy learning mechanisms to navigate through a preset course. In this paper we explore the relationship that different mediation equations have when related to different learning environments. We explore five different functions in relation to the tasks being accomplished by an artificial agent navigating through a preset course. We also look in detail at the reaction times between algorithms when different tightness of control is used within the fuzzy mediation core engine. This paper highlights the appropriateness of each of the five mediation equations in relation to the setting in which the fuzzy mediation framework is deployed. Moreover, we guide potential users through the analysis of tightness of control also as it is appropriate to the same environments of deployment of this framework.

Introduction

Information fusion is a domain that deals with the integration of input from multiple sources in order to create a single output (Kokar, Tomasik, & Weyman 2004). This task is particularly difficult in the event of conflicting information coming from different sources. The mediation that needs to take place is the heart of any information fusion mechanism. One of the applications that may come to mind when talking about information fusion systems is the domain of shared control. As technology advances, more vehicles rely heavily on computing for their control (OMalley 2005; Norman 2005; Wallace 2000). This form of control that does not allow a human to interact directly with the machine any more leaves room for the application of shared control concepts to a domain that, although cites shared control, is really based on supervised control with little to no shared interaction (Norman 2005). Fuzzy mediation is an innovative concept that allows two agents to interact in real-time while controlling an object. The majority of information fusion algorithms involve a static method of evaluation set at the time of system design (Kokar, Tomasik, & Weyman 2004). The concept introduced in this article is based on a dynamic concept of mediation, assuming that the two controllers can interact and learn, thus the need for a reward system shifting control to the learner, as its knowledge allows for actions that resemble the trainer's inputs.

Fuzzy Mediation

As it would be impossible for a vehicle to be controlled simultaneously by two different controllers, we looked at concepts of information fusion for a solution. The fuzzy controller performs three distinct operations. The first is the analysis of the inputs to determine the closeness of control; it then performs a revision of the weight of control between the expert and the novice controller; finally it computes the value of the single output (Vincenti 2007; Vincenti & Trajkovski 2007b).

Analysis of the inputs

Cantorian set theory leaves little room for gray areas; a room's temperature can be classified as hot, medium or cold. This system does not take into consideration the possibility of the same room being perceived as comfortable or chilly by two different persons. Fuzzy sets provide a solution that takes into consideration values that fall within multiple sets (Zadeh 1965). These kinds of sets can be utilized within the field of information fusion applied to a situation of multiple controllers trying to interact with a vehicle by a means of comparison.

The analysis of the inputs coming from the two controllers aims at understanding the distance between the values. The input coming from the expert user is mapped to the center of the range [-10, 10], as shown in Figure 1.

The range of [-10, 10] represents the highest possible deviation between the input of the expert and the one of the novice. After the input of the expert becomes the center of this domain, we calculate the difference between the value of the original input of the expert and the one of the novice. This value is then also mapped to the domain shown above. When the distance between the inputs is mapped, it will fall

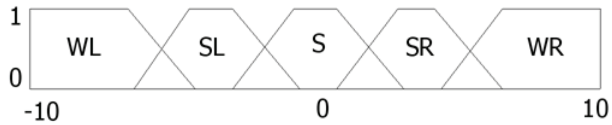

Figure 1: Fuzzy sets used for classifying the deviation between inputs

within one or two sets that span over the range, as shown in Figure 1. The five sets we deal with are: WL (Wide deviation to the left), SL (Slight deviation to the left), S (Similar), SR (Slight deviation to the right), and WR (Wide deviation to the right). In fuzzy set theory, the value can belong to a set with a certain degree of belonging. Such degree of belonging is calculated by a membership function. In our case, we use a simple membership function, also shown in Figure 1.
The application of a linguistic modifier to the deviation between the controller inputs also keeps in consideration the degree of belonging of the difference to the different sets. It is important to note that a value may fall completely (degree of belonging = 1) within one set, or the value may belong mostly (.80) to the set of deviations deemed as Similar (S), and partly (.20) to the set of Slight deviations to the left (SL).

Revision of the weight of control

Fuzzy mediation sees the fusion of the inputs of the expert and the novice as a balance between the two. The more the novice performs similarly to the expert, the more control will shift in favor of the second controller. Likewise, the more the control of the novice differs from the one of the expert, the more control will shift back towards the expert.

Given this preamble, this second part of the fuzzy mediation algorithm analyzes the linguistic modifiers applied to the deviation of the inputs during the first phase. Control is mapped to the range of [-1, 1], where a control weight of -1 identifies a control fully in the hands of the expert, a value of 1 instead refers to control managed by the novice. Figure 2 shows the visualization of this concept. As we apply the concept of fuzzy sets to this section of the algorithm, a value in the range (-1, 1) identifies a control that is mixed in a certain proportion between the expert and the novice. The arrow in Figure 2 shows a possible weight of the mediation of control between the trainer and the trainee. At the beginning of the simulation the weight has a value of -1. When the classification of the distance between inputs is analyzed, there are several actions that can be taken. If the inputs are classified as similar, control is given more to the novice. If the deviation between inputs is slight, then control stays unvaried. If instead the deviation is wide, more control is given to the novice. The shifting of the weight from one controller to the next occurs in a linear fashion, with increments or decrements of 0.2 points on the range [-1, 1] presented earlier. In the case of a distance between inputs that belongs to two sets, then we will multiply the degree of belonging to each of the sets to the action associated with that particular set. Using the example given earlier, a value that belongs to the set S with membership 0.8 will receive an increase in control of 0.2 (the standard increment) multiplied by the membership value, which means an increase of the weight of control of 0.16. The same value also belongs to the set SL with membership 0.2. The action associated with a deviation that is classified as slight is a movement of 0 of the balance between controllers, so the action for this set is calculated by multiplying 0.2, the membership value, to 0. The addition of these values, 0.16 and 0, shows the overall shift in control, which is of 0.16 in favor of the trainee.

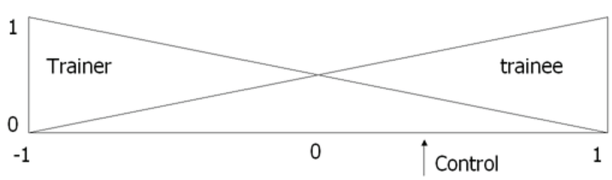

Figure 2: Fuzzy sets that regulate the balance of control

Calculation of a single output

After the weight of control is updated, we need to calculate a value that will serve as a single input stemming from the original inputs of the two controllers. For this computation we need to refer to the original values. Equation 1 regulates the third section of this algorithm.

$$MO = \mu_T \cdot E_I + \mu_t \cdot N_I \tag{1}$$

where MO symbolizes the mediated output, μ_T refers to the membership value of the weight of control to the Trainer set (Expert) and μ_t refers to the membership value of the weight of control to the trainee set (Novice). The inputs of the two controllers are represented by E_I for the Expert's input and N_I for the Novice's.
In the case of a driving simulator, we may have an expert applying a turn of 15 degrees to the right and a novice applying a turn of 25 degrees. If the weight of control has a value of -1, the mediated output will have a value of 15 degrees to the right. Likewise, if the weight of control has a weight of 1, the mediated output will be of 25 degrees to the right. If the weight is anywhere in between, for example and, the mediated output will be of a 20 degree turn to the right.

Demonstration of Concept

The first simulation that we performed focuses on two linear paths that intersect at some point, without any deviation on the side of the novice to try and resemble the actions performed by the expert (Figure 3) (Vincenti & Trajkovski 2007a). In this figure, the expert is represented by the line with the smallest slope, the novice by the one with the largest, and the mediated output is shown by the third line. As we can see, the lines that represent the expert and the novice intersect at one point only, and they do not converge past the intersection. The output that is produced by the fuzzy mediation engine overlaps the path taken by the expert, since the path traced by the novice is quite distant.

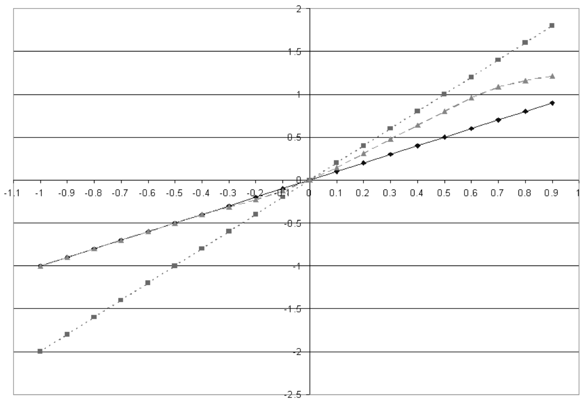

Figure 3: Simulation with two linear functions

As the controllers perform actions that are closer, the mediated output starts shifting towards the novice's line. After the intersection, the output of the fuzzy controller shows that control is shared somewhat equally between the expert and the novice for a few steps, but then it leans towards the expert's output once again. This is due to the fact that the novice's controls deviate widely from the ones of the expert, thus taking control away from the first and giving it more to the second. The output created by the fuzzy mediator does not adhere much to the line of control carried out by the novice because the time during which the output of the novice and the ones of the expert were somewhat similar is rather short. In cases where the lines would not intersect, the control will never move away from the expert's line.

In the second simulation we have taken into consideration a situation where the expert is controlling the vehicle in a linear fashion, and the novice performs a motion trying to align to the direction of the expert in a logarithmic manner (Figure 4). Also in this case, when the simulation starts, the output of the fuzzy controller is overlapping the control of the expert. As the novice's output gets closer to the expert's, the output of the fuzzy mediation shifts. This time,

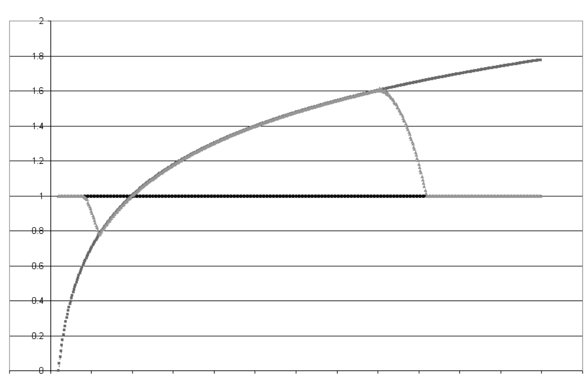

Figure 4: Simulation with an expert linear function and a novice logarithmic one

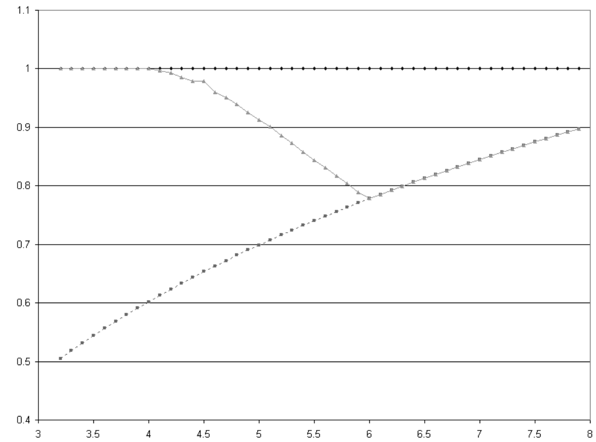

Figure 5: Detail of the area in Figure 4 where control shifts to the novice

the two values are close enough for quite a few cycles of comparison within the fuzzy mediation system, where each time the two values are considered similar, giving more control to the novice. Figure 5 shows a closer look at the area of the plot where the output of the fuzzy mediation engine starts moving towards the novice's controls. We should note that the difference between values, at that point, is only of 0.4 units. At this point, control is in the hands of the novice completely. After the lines intersect and then diverge, control will be retained by the novice, at least until the fuzzy mediator perceives a difference between the two inputs that is significant enough to start shifting control back in favor of the expert. This happens when the two values diverge by more than 0.6 in this simulation. The area where control starts shifting back to the expert is highlighted in Figure 6.

In Figure 7 we can see yet another simulation. In this case, the expert and the novice perform actions that are at times

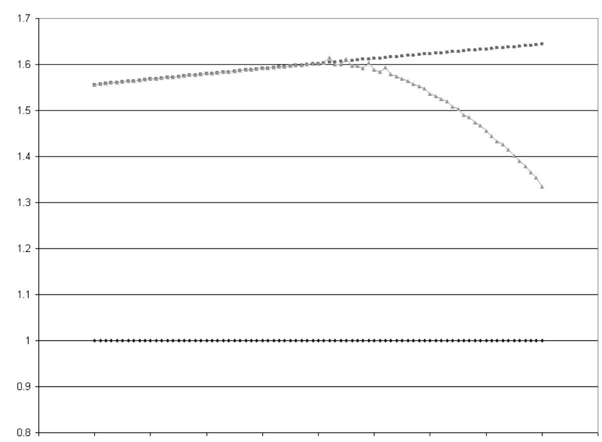

Figure 6: Detail of the area in Figure 4 where control shifts to the expert

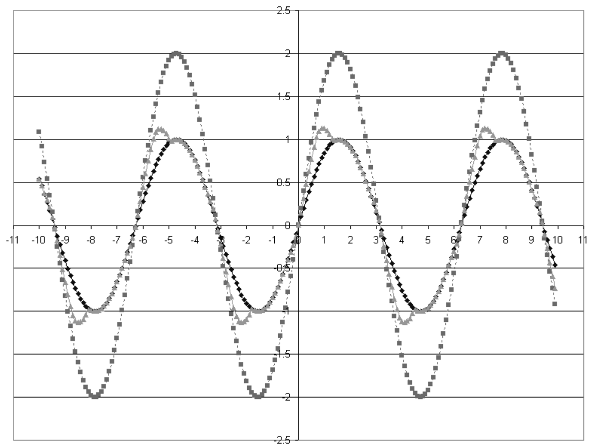

Figure 7: Simulation with two sine waves

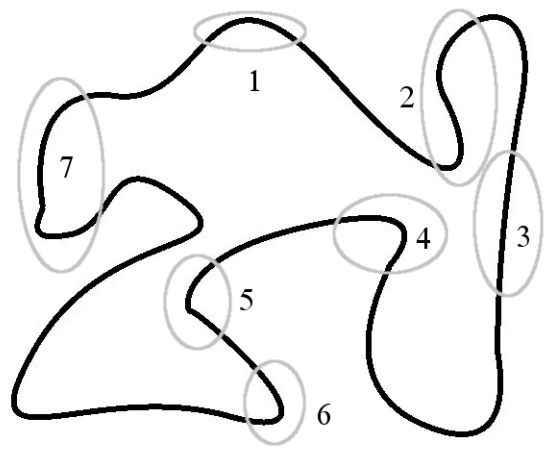

Figure 9: Path of the simulation

similar, and then diverge to become similar again later in the simulation. In this particular one, the actions are periodic, so that we can see that the fuzzy mediation engine performs consistently. The expert's control is represented by wave closer to the X-axis. The control is initially given to the expert and is slowly shifting towards the novice. As the novice's performance deviates from the expert's, the expert's input starts weighing more and more in the overall output of the fuzzy controller. Figure 8 shows the particular where control shifts back towards the expert.

Description of the Environment

The environment used for these experiments is a simple agent that follows a line. Figure 9 shows the pattern that was used. The pattern contains only the white background and the black line. The areas that have been highlighted show the sections of most interest. Section 1 was created to see the behavior of the agent in a mild turn to the right; section 2 instead simulates a sharp turn to the left followed by a moderate turn to the right. Section 3 mimics a straight path. Section 4 reveals a tight turn to the left, just like section 5. Section 6 is another tight turn to the right after a straightway and finally section 7 shows quick changes in direction.

In these experiments we use different levels of control. When we want to create an environment when the two inputs need to be closer in order to shift the weight from the expert to the novice we simply need to set the boundaries of the sets shown in Figure 1 to tighter limits. The three levels of control we use are the following: Tight control, Moderate control and Loose control. Table 1 shows the values associated with each level of control. Each set carries four values, (OL, IL, IR, OR), where OL refers to the outer left boundary, IL to the inner left, IR to the inner right and OR to the outer right. The simulated agent is composed of a central unit that contains sensors. The sensors check the terrain in front of the agent for color. The sensors can either pick up white, which is the background, or black, which is the line. The sensors are arranged on a probe that scans the range $[-45, 45]$ in front of the agent at 5-degree intervals. Figure 10 shows an image of the agent that is following a line. The light gray is the body of the agent while the dark gray spots in front of it represent the range of action of the sensors.

The sensors communicate to the agent the color of the terrain at each angle. Then the agent will group together the angles that recorded a reading of a line and calculate the av-

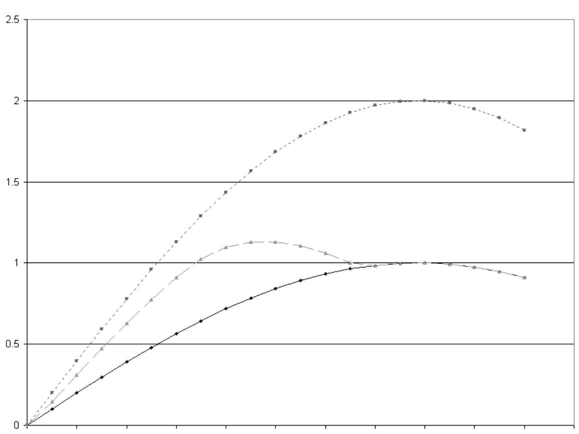

Figure 8: Detail of one of the areas in Figure 7 where control shifts to the expert

Table 1: Boundaries of fuzzy sets used to classify the deviation between controllers

	Tight	*Moderate*	*Loose*
WL	($-\infty$, $-\infty$, -5, -3)	($-\infty$, $-\infty$, -6, -4)	($-\infty$, $-\infty$, -8, -6)
SL	(-5, -3, -2, -1)	(-6, -4, -2, -1)	(-8, -6, -4, -2)
S	(-2, -1, 1, 2)	(-2, -1, 1, 2)	(-4, -2, 2, 4)
SR	(1, 2, 3, 5)	(1, 2, 4, 6)	(2, 4, 6, 8)
WR	(3, 5, $+\infty$, $+\infty$)	(4, 6, $+\infty$, $+\infty$)	(6, 8, $+\infty$, $+\infty$)

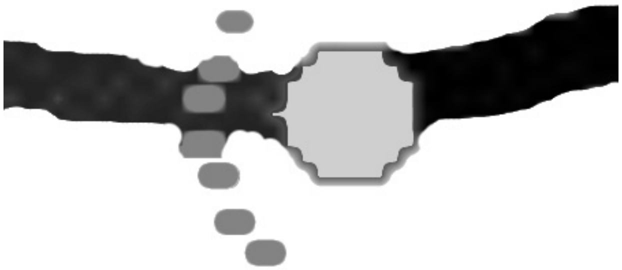

Figure 10: Diagram of the simulated agent with its sensors

erage. Such average will be analyzed by the agent, which will select the new heading. An agent can perceive changes in direction up to $\pm$ 45 degrees.

In order to simulate the behavior of agents we assigned them a preset behavior that allows them to navigate successfully through the pattern selected. In order to simulate an expert agent and a novice one, we chose equations that are slightly different. The typical expert is represented by a simple linear function. When the agent receives the reading from the sensors, the difference in heading is applied directly to the heading, so if the sensors read that, in order to follow the line the agent needs to apply a 15-degree turn to the right, the agent will perform a 15-degree turn to the right.

The simulation that represents the case of a novice is powered by an agent that relies on the cube of the difference normalized to the range $[-90, 90]$. The two equations that are used to drive the agents are reported in Figure 18. The dotted line represents the behavior of the expert and the solid one the behavior of the novice.

This interpretation of the agents shows an expert that acts as expected, with a linear response to the situation. The novice instead reacts more slowly only to overcompensate as the deviation required in order to remain on track increases. We also carry out other simulations where the difference between the inputs of the expert and the one of the novice are very different.

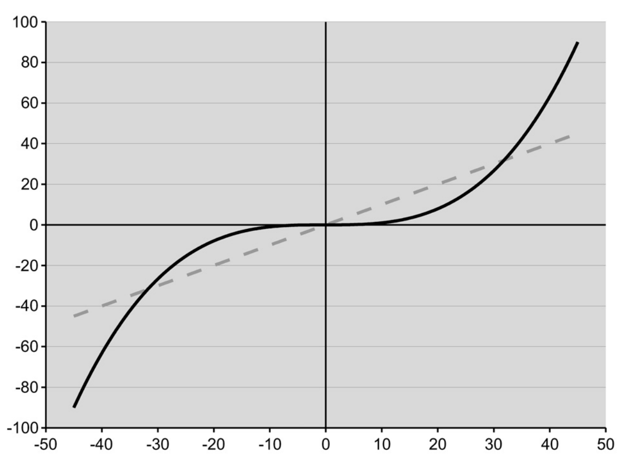

Figure 11: Control functions, expert (dotted) and novice (solid) controllers

Mediation Engines

An important component of this approach is the engine that drives the regulation of the mediation between inputs (Vincenti & Trajkovski 2006). The scenarios that have been analyzed so far used a linear function. This component of fuzzy mediation can be replaced with any other equation. The main objective for this experiment is to analyze the behavior of the fuzzy mediation engine as we use different mediation equations. The equations tested are the following:

$$Y = X \tag{2}$$

$$Y = log(X) \tag{3}$$

$$Y = exp(X) \tag{4}$$

$$Y = X^3 \tag{5}$$

$$Y = \sqrt[3]{X} \tag{6}$$

It is important to note that the output of equations 3 and 4 was normalized to the range of [-1, 1]. These equations will determine the dynamics as control shifts in favor or either agent.

For this experiment we used the simple two-controller line follower. The two agents were represented by an expert agent powered by the equation $Y = X$ and a novice controller powered by $Y = X^3$, both shown in Figure 11.

Experiments

This experiment also explored the dynamics of the agent with two controllers with different increments within the process of mediation. Going back to the description of the algorithm, we explained how control shifts towards the trainer or towards the trainee with a certain increment over the domain [-1, 1], with -1 signifying total control to the trainer and 1 allowing the trainee to control the object. For other experiments we used the increment of 0.2, which is a choice that allows a dynamic shift of control that is not either too rapid or too slow.

Before we get any further into the description of the experiments, it is important to note that each run was conducted over a single lap around the course. While conducting preliminary experiments, we noticed that the performance of an agent for each lap is quite similar through a 10-lap test. The values shown in Table 2 show the average mediation weight for different tightness of control after each lap.

We analyzed the average and the standard deviation of the values reported in Table 2, shown in Table 3. We can see that the standard deviation for each lap is small enough for which a lap performed under a tight comparison method cannot be misinterpreted as a lap that was mediated through a moderate controller. When we analyze the performance of a moderate tightness of control versus a loose one, we can see that the difference is large enough for which confusion is very unlikely.

The equations were chosen with specific purposes in mind. We wanted to analyze the behavior of each one so that we

Table 2: Agent performance over 10 laps

Lap	*Tight*	*Moderate*	*Loose*
1	0.591630435	0.621014947	0.82562658
2	0.587499453	0.625134209	0.807701818
3	0.599046749	0.629752809	0.802773797
4	0.592637091	0.625473723	0.810562727
5	0.600289971	0.621890988	0.809399855
6	0.594071299	0.623954958	0.830268743
7	0.5858	0.625674231	0.812529946
8	0.58594359	0.630120029	0.825644846
9	0.603001442	0.618650759	0.814850542
10	0.590514801	0.6208102	0.817715729

Table 3: Average and Standard Deviation values for the base experiment

	Tight	*Moderate*	*Loose*
Average	0.593043483	0.624247685	0.815707458
St. Dev.	0.006063892	0.003776346	0.008954575

could match a certain purpose for the mediation to each engine. Some equations leave the control to the trainer for a longer time, while others favor the trainee. Possible applications of this algorithm may require certain settings over others, and this work outlines some suggestions for the initial setup of the environment. Tables 4 through 8 show the average mediation weights for the simulations just described.

Discussion

We can see that Equation 2 (Table 4) offers a good balance among all the ones tested. The average mediator weight for any of the simulations performed, compared to the same setup of other equations always shows to be in the middle of the range. Equation 3 shows a control (Table 5) that favors the trainer when performing the simulations. Equation 4 instead offers the highest average mediation weight, showing (Table 6) that the trainee was the controller that was most in control of the agent. Equation 5 also shows a preference for the trainee (Table 7), but not as much as Equation 4. Finally, Equation 6 seems to leave more control to the trainer (Table 8), making this the simulation that leaves the least leeway to the trainee.

The second aspect that is relevant to this work is the measure by which control was shifted towards one or the other controller. As stated previously, the default value is an increment of 0.2. In these simulations we tested the increments

Table 4: Performance using the mediation engine based on Equation 2

	0.05	*0.1*	*0.2*	*0.5*
Tight	0.598188	0.591931	0.593775	0.609668
Moderate	0.654717	0.630193	0.622115	0.647160
Loose	0.837451	0.807864	0.823890	0.810044

Table 5: Performance using the mediation engine based on Equation 3

	0.05	*0.1*	*0.2*	*0.5*
Tight	0.278387	0.246983	0.279714	0.326584
Moderate	0.319679	0.329286	0.310036	0.383242
Loose	0.698477	0.645600	0.597444	0.624078

Table 6: Performance using the mediation engine based on Equation 4

	0.05	*0.1*	*0.2*	*0.5*
Tight	0.729840	0.737177	0.727779	0.759397
Moderate	0.780996	0.774061	0.769686	0.770878
Loose	0.889485	0.896667	0.891935	0.883299

of 0.05, 0.1, 0.2 and 0.5.

As we review the results of the experiments, we can see that in the majority of the cases the different increment in shift of control does not play a significant difference. Although values fluctuate up and down by a few points, generally the tightness of control does not play a major factor in preferring the trainer versus the trainee.

The first point that is interesting is that the simulation with the increment set to 0.05, a loose control and a mediation engine driven by Equation 3 did not lead to a successful completion of a lap. The agent was not able to perform a full rotation. Although this mediation engine is considered to prefer the trainer to the trainee, an increment of 0.05 seems to be too slow to let the trainer gain back control in situations where the performance of the trainee is not adequate.

The most notable effect of the different increment in the shift of the mediation can be seen in the simulations driven by Equation 6, the one that retains control closest to the trainer. As the increments decrease, the trainer retains control even more through the simulation.

When we first explored possible equations that would power the mediation engine, we assumed that we were to base our decision of which would relinquish more or less control dynamically by analyzing the behavior of the behavior along the range . The data shows that, given the simulation setup

Table 7: Performance using the mediation engine based on Equation 5

	0.05	*0.1*	*0.2*	*0.5*
Tight	0.701060	0.698216	0.703044	0.690018
Moderate	0.773597	0.749747	0.730973	0.733443
Loose	0.902676	0.909070	0.888819	0.863218

Table 8: Performance using the mediation engine based on Equation 6

	0.05	*0.1*	*0.2*	*0.5*
Tight	-0.72860	-0.62684	-0.57903	-0.55150
Moderate	-0.70347	-0.60455	-0.56145	-0.52779
Loose	-0.30275	-0.22137	-0.20789	-0.18675

described earlier, the most important interval to keep in mind is actually the behavior within the range , as that seems to be the most dynamic portion of control, and at the same time the range where we can find the mediation the majority of the time.

Conclusions

Overall, we feel that we can choose which area needs to be most dynamic as far as the mediation goes by selecting the appropriate mediation engine. The findings of these experiments suggest that Equation 2 offers a balanced mediation between the two controllers.

Equations 4 and 5 are the ones that leave the most room to novice controllers, thus stimulating the exploratory nature of certain learning algorithms that construct rules as the agent explores and creates associations that show good behaviors and bad behaviors.

Equations 3 and 6 are preferable in situations where the learning mechanism of the novice agent thrives on examples that thrive on examples that show it what to do, instead of also showing what not to do. These equations also seem fitted for environments where there is a need for an agent to learn rules that are very similar, if not identical, to the expert's knowledge.

Moreover, a smaller value regulating the shift of control between the trainer and the trainee seems more appropriate for precision type agents. We do recommend though that the smallest increment adopted for situations similar to the one described by these experiments be at most reduced to 0.1, as a smaller value may lead to problems in the performance of the task.

References

Kokar, M. M.; Tomasik, J. A.; and Weyman, J. 2004. Formalizing classes of information fusion systems. *Information Fusion* 5(3):189–202.

Norman, D. 2005. *There's an Automobile in HCI's Future.* Interactions.

OMalley, P. G. 2005. *Troubleshooting Machine Control: Contractors Talk About Making the Most of 3D GPS.* Gradient and Excavator Contractor.

Vincenti, G., and Trajkovski, G. 2006. Fuzzy mediation for online learning in autonomous agents. In *2006 Fall AAAI Symposium*.

Vincenti, G., and Trajkovski, G. 2007a. Fusion applied to learning environments. In *11th International Conference on Cognitive and Neural Systems*.

Vincenti, G., and Trajkovski, G. 2007b. Fuzzy mediation as a dynamic extension to information fusion. In *Fusion2007*.

Vincenti, G. 2007. *Fuzzy Mediation as a Dynamic Method of Information Fusion and Shared Control.* Ph.D. Dissertation, Towson University.

Wallace, J. 2000. *Unlike Airbus, Boeing lets aviator override fly-by-wire technology.* Seattle Post-Intelligencer.

Zadeh, L. 1965. Fuzzy sets. *Information and Control* 8:338–353.