2016 American Control Conference (ACC)
Boston Marriott Copley Place
July 6-8, 2016. Boston, MA, USA

A Learning Algorithm to Select Consistent Reactions to Human Movements

Carol Young and Fumin Zhang

***Abstract*— A balance between adaptiveness and consistency is desired for a robot to select control laws to generate reactions to human movements. Two existing algorithms, the weighted majority algorithm and the online Winnow algorithm, are biased for either strong adaptiveness or strong consistency. The dual expert algorithm (DEA), proposed in this paper, is able to achieve a tradeoff between consistency and adaptiveness. We give theoretical analysis to rigorously characterize the performance of DEA. Both simulation results and experimental data are demonstrated to confirm that DEA enables a robot to learn the preferred control law to pass a human subject in a hallway setting.**

I. Introduction

As robots move away from the factory floor they encounter environments that are no longer well structured in time or space. This provides motivation for them to be equipped with real time algorithms that tolerate and adapt to noise. These challenges are compounded when humans come in close contact with a robot [1], [2], where robots should behave in predictable ways [3], [4]. Thus any choice of action that a robot makes should occur in real time, while maintaining a balance between adaptiveness and consistency.

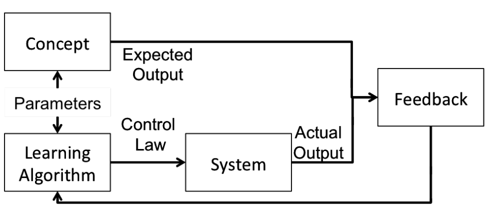

Fig. 1. How Learning Algorithms Work

Typically a learning algorithm interacts with a concept, which maps parameters to an output. An algorithm uses observed parameters to make a selection of a control law. Then when the expected output is known it will be compared to the actual output from the system. If these outputs match then we say that the selection is correct, otherwise an error occurs, and the estimation of concept is updated. The online learning method handles each parameter-output pair and update sequentially and only uses each pair once, hence only requiring limited memory and processing time [5]. This makes online learning ideal for implementation on robotic systems. This paper deals with learning algorithms that choose between two control laws, with the assumption that there are only two possible expected outputs. The output value chosen more often for given parameters is called a preference, while the other output value is called a deviation.

The particular online learning method we are examining is ensemble learning [6], [5]. A notable example of online ensemble learning is the weighted majority algorithm [7]. This algorithm can be used to take the Winnow Algorithm [8] online [9]. It also has inspired popular boosting algorithms such as AdaBoost [10] and its derivative Brown Boost [11].

These online ensemble methods can encounter drift, which is a change in the concept, e.g. mapping from parameters to outputs, over time [12]. Multiple online methods have been implemented to adapt to drift, including the Dynamic Weighted Majority where new experts are added and deleted over time [13]. Other methods for tracking drift include using regret minimization [14], and tracking the average of online parameter-output pairs [15].

These methods can also encounter uncertainty in the concept. Robustness measures how well the selection can tolerate this uncertainty, and has been considered for online learning methods [7], [9]. The novel approach of this paper is to analyze a specialized form of robustness, the consistency, a measure of how easily the selection changes when deviating outputs are observed. Consistency is important for a learning algorithm to meet predictability requirements for easy human robot interaction [3], and prevent chatter in motion.

The goal of this paper is not to create an algorithm that is the most adaptive or the most consistent, but rather an algorithm that can manage the tradeoff between the two. A learning algorithm called the dual expert algorithm (DEA) is developed to select between two possible control laws. We show that the DEA has bounded errors in selection, increasing consistency as the number of data points increases, while retaining its abilitiveness. We compare the DEA with two similar algorithms (weighted majority algorithm (WMA) [7] and the online Winnow algorithm [9]). The DEA and the consistency analysis are novel contributions of this paper.

We justify the conclusion and demonstrate real life applications in HRI by implementing the algorithm onto a moving robot avoiding a known user approaching it in the hallway. Existing methods to solve this problem have the robot stop [16] or treat humans as obstacles in path planing [17], [18], [19]. Our method considers that humans tend to pass on a predetermined side [20], [21] with parameters that are similar in passing robots and humans [22] to allow avoidance in a socially aware manor without large path planing costs.

The layout of this paper is as follows. Section II presents the problem formulation. Section III explains the Dual Expert Algorithm and section IV introduces the Expanded Dual

*The research work is supported by ONR grants N00014-10-10712 (YIP) and N00014-14-1-0635; and NSF grants OCE-1032285, IIS-1319874, and CMMI-1436284, and an ARCS Scholar Award

[2]Carol Young and Fumin Zhang are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332, USA fumin@gatech.edu.

Expert Algorithm. Section V shows the simulation results of both algorithms. Section VI presents the results of the experiment, and Section VII presents the conclusion.

II. BACKGROUND

In the general case the weighted majority algorithm (WMA) [7] and the Winnow algorithm [8], [9] use an arbitrary number of experts to select between two different control laws. Each expert takes in the parameters of the system and uses them to make their selection. The weight of the experts that select each control law are then summed and then the control law with the highest weight is chosen.

In the case without parameters, two experts, one for each control law, are all that are needed. Then in the case of an error, the chosen weight is decreased by a factor of two. The Winnow is very similar to the WMA but a correct response also results in the weight increasing by a factor of two.

Both WMA and Winnow have bounded error [7], [8]. In addition their consistency is determined by how often the algorithm will switch its selection. And their adaptiveness is determined by how quickly the selection changes after the preference changes. WMA is able to adapt easily to drift, however it is not consistent, since one deviation is enough to cause it to switch its selection. The additional doubling of weights allows Winnow to be consistent but at the cost of its adaptiveness, since now a switch in selection can only be caused when there are an equal number of deviations as there are of the preferred output.

The problem that we wish to solve is to create an algorithm that is more consistent than the WMA, and more adaptive than the Winnow. It has bounded error like both the WMA and Winnow while maintaining a balance between consistency and adaptiveness.

III. DUAL EXPERT ALGORITHM

The Dual Expert Algorithm (DEA) is shown in Algorithm 1. The key difference from WMA and Winnow is in line 12. When a correct selection is made and the weight is over 0.25 the weight is increased by taking its square root, which keeps the value $W_a \leq 1$. This bound allows for the DEA to maintain a balance between consistency and adaptiveness.

Algorithm 1 Dual Expert Algorithm

1: Set $W_1 = W_2 = 0.5$
2: Choose selection a from argmax(W_1, W_2)
3: **if** $W_1 = W_2$ **then**
4:     Choose a randomly from $\{1, 2\}$ with equal probability
5: **end if**
6: **if** Error **then**
7:     $W_a = \frac{W_a}{2}$
8: **else if** Correct **then**
9:     **if** $W_a \leq 0.25$ **then**
10:         $W_a = 2W_a$
11:     **else**
12:         $W_a = \sqrt{W_a}$
13:     **end if**
14: **end if**

A. Performance Analysis

Suppose the algorithm has learned the preference of C. Let δ be the number of iterations when C disobeys its preferred option and makes a "deviation". Let the total number of iterations be N, we require that $\delta < \frac{N}{2} - 1$.

A deviation will trigger the algorithm to make a "wrong" selection based on its past knowledge. Then the following claim can be made about the selection error.

Proposition 3.1: The maximum number of selection errors that Algorithm 1 generates is $1 + 2\delta$.

Proof: Let W_p be the weight of the preferred option of C, and let W_d be the weight of the deviation option. When $\frac{W_\text{p}}{W_\text{d}} > 1$ the preferred selection is chosen. When $\frac{W_\text{p}}{W_\text{d}} < 1$ the deviation is chosen. And when $\frac{W_\text{p}}{W_\text{d}} = 1$, which is the starting condition, a random choice is made.

When $\frac{W_\text{p}}{W_\text{d}} \geq 1$ a deviation can cause an error, leading to a total of δ possible errors. Each deviation can decrease the ratio $\frac{W_\text{p}}{W_\text{d}}$ by at most a factor of 2. Since the ratio starts at 1 the minimum that the ratio can ever be is $2^{-\delta}$. Thus to return this ratio back to 1 it must be increased by a factor of 2^{δ}. This requires there are δ times that the preferred selection is chosen by C while $\frac{W_\text{p}}{W_\text{d}} < 1$. Hence the algorithm can make at most δ additional selection errors when $\frac{W_\text{p}}{W_\text{d}} < 1$.

Therefore after considering all deviations, $\frac{W_\text{p}}{W_\text{d}} \geq 1$. And since the deviation can be chosen when $\frac{W_\text{p}}{W_\text{d}} = 1$, but this error will then raise $\frac{W_\text{p}}{W_\text{d}} > 1$, there can be at most 1 additional error. Thus the total number of errors is less than $\delta + \delta + 1$. Making the error bound of algorithm 1 as $E = 1 + 2\delta$. ■

The error bound $1 + 2\delta$ is on the same order of magnitude as the Weighted Majority Algorithm which has an error bound of $E = 2.4 + 2.4\delta$ [7].

B. Consistency Analysis

We show that the selection of the DEA is consistent to deviations made by C. This means that the number of switches between selections of the algorithm are kept small, by ignoring deviations. We say the DEA becomes more consistent as the number of iterations increases the ability of the algorithm to ignore deviations increases.

For comparison, consider the weighted majority algorithm, the two weights will always be within a factor of two from each other. This is because the weights start even, and the only modification to weights that can be done is decreasing the current maximum by a factor of two.

Thus if C makes a deviation, then the weighted majority algorithm will change the weights to be equal because the larger weight will be reduced by half. The maximum number of switchings an algorithm can make is 2δ where δ is the number of deviations C makes. No deviation is ignored.

Next we will argue that using the dual expert algorithm, the algorithm will make less switchings in its selection because it can ignore deviations made by C. Let us define $R = \max\{\frac{W_1}{W_2}, \frac{W_2}{W_1}\}$. First we argue that in a number of situations, R will be increased to a value that is greater than 2.

Claim 3.2: Starting from $W_1 = W_2 = 0.5$, if the DEA makes one error followed by one correct selection, then $R > 2$.

Proof: The weight that is updated is always the maximum weight. This means that when there is an error the maximum weight will decrease by a factor of two, in this case making $\min(W_1,W_2) = .25$. Then the $\max(W_1,W_2) = .5$.

If the next selection is correct then according to algorithm 1 $\max(W_1,W_2) = \sqrt{(.5)} > .5$, which leads to $R = \sqrt{(.5)}/.25 > 2$. ■

Claim 3.3: Starting from $\max\{W_1,W_2\} < 0.25$ and $R \leq 2$, if DEA makes two consecutive correct selections, then $R > 2$.

Proof: R is the maximum of two positive reciprocal values, thus $R \geq 1$. A single correct selection when $\max\{W_1,W_2\} < 0.25$ increases the correct weight, and thus R by a factor of 2. Thus $R \geq 2$, and $\max\{W_1,W_2\} < 0.5$. An additional correct selection will then increase the $\max\{W_1,W_2\}$, thus increasing R so that $R > 2$. ■

Claim 3.4: Starting from $\max\{W_1,W_2\} < 0.5$ and $R \leq 2$, if DEA makes one error followed by one correct selection, then $R > 2$.

Proof: A single error would decrease the max weight by a factor of 2 so that $\min\{W_1,W_2\} < 0.25$ and $1 < R \leq 2$.

If $\max\{W_1,W_2\} \leq 0.25$ then a correct selection would increase the max weight by a factor of 2 so that $2 < R \leq 4$.

And if $\max\{W_1,W_2\} > 0.25$ then a single correct selection would increase the max weight such that $\max\{W_1,W_2\} > 0.5$. Then $R = \max\{W_1,W_2\}/\min\{W_1,W_2\} > 0.5/.25 = 2$. ■

Claim 3.5: By making consecutive correct selections, the value of R can be increased until $R = \frac{1}{\min\{W_1,W_2\}}$.

Proof: Each correct selection only affects the maximum weight. And while $\max\{W_1,W_2\} \leq .25$, $\max\{W_1,W_2\}$ increases by a factor of 2. Once $\max\{W_1,W_2\} > .25$, $\max\{W_1,W_2\}$ is square rooted to form the new maximum. Thus $\max\{W_1,W_2\}$ converges to 1 as the number of consecutive correct selections increase. Since $R = \max\{W_1,W_2\}/\min\{W_1,W_2\}$, and the minimum weight is unchanged, R converges to $\frac{1}{\min\{W_1,W_2\}}$. ■

The number of deviations the dual expert algorithm can ignore depends on the actual ratio between the weights.

Claim 3.6: If $R > 2^K$ where $K > 1$, then $(K-1)$ consecutive deviations can be ignored.

Proof: Each deviation reduces the ratio R by a factor of 2. Because the larger weight is decreased by a factor of 2. Since $R > 2^K$, the larger weight can be decreased by 2 for a total of $K-1$ times, and still be the larger weight, thus allowing the algorithm to ignore $K-1$ deviations. ■

Therefore, if C makes deviations, the DEA will be more robust than the WMA. Since the weighted majority algorithm does not ignore deviations and the DEA does.

C. Adaptiveness Analysis

If we consider drift to be the change in preferred output of the concept C from p to $\tilde{p}$. Then the adaptiveness of the DEA can be shown to be directly linked to the amount of deviations, that have been encountered before drift occurred.

Claim 3.7: R is bounded by the number of consecutive deviations.

Proof: Claim 3.5 showed that the maximum of R is $R = \frac{1}{\min\{W_1,W_2\}}$. And only an error can decrease weights in DEA and Proposition 3.1 showed that the error is bounded by the number of deviations. Thus $\min\{W_1,W_2\}$ is bounded by the number of consecutive deviations.

Therefore $R = \frac{1}{\min\{W_1,W_2\}}$ is bounded by the number of consecutive deviations. ■

Claim 3.8: For any given $2^{K-1} \leq R < 2^K$ after output $\tilde{p}$ is observed for K consecutive iterations, DEA will change from its selection p to the alternate selection $\tilde{p}$.

Proof: Each time the output is $\tilde{p}$ and DEA selects p there is an error. This error decreases the maximum weight, and thus R is reduced by a factor of 2.

Then after $K-1$ consecutive outputs of $\tilde{p}$, $1 \leq R < 2$.

If $1 \leq R < 2$ then an additional output of $\tilde{p}$ will decrease the weight of p to be less than the weight of $\tilde{p}$. This will change the selection from p to $\tilde{p}$.

Thus a maximum of K consecutive C outputs of $\tilde{p}$ are needed to change the selection of the DEA. ■

These two claims imply that the number of errors needed to change selections is limited by the number of deviations leading up to the change. This can be compared to Winnow algorithm where the 1 in $R = \frac{1}{\min\{W_1,W_2\}}$ is replaced by $\max\{W_1,W_2\}$, which can be very large. This allows Winnow to be consistent but not adaptive.

IV. Expanded Dual Expert Algorithm

The DEA uses two weights to select potential outcomes. However an algorithm can encounter situations where the preference depends on a parameter. If the parameter is smaller than a threshold then output 1 is preferred otherwise output 2 is preferred. An example of this type of parameter would be the relative position that a human starts in a hallway. Starting along the left wall would imply that the human prefers passing on the left, and starting along the right wall would imply the opposite. Somewhere between the walls this preference changes. We propose the expanded dual expert algorithm (EDEA) to learn these preferences using weights that are function of the parameter.

The parameter is discretized into M values. We call each values of the parameter a section and each section is indexed by n. Let us assume that there are two edges $n = 1$ and $n = M$, and that output 1 is the preference of edge $n = 1$, while output 2 is the preference at edge $n = M$. And the two weights are now represented as two M dimensional vectors W_1^i and W_2^i indexed by $i = 1,2,...,M$.

The EDEA algorithm in Algorithm 2 updates its weights based on whether a correct or an incorrect selection is made. Line 8 shows that when an incorrect selection is made the weights between n and the edge that prefers the observed output are decreased. And when a correct selection is made the weights between n and the edge that prefers the observed output are increased, as shown in line 10.

Algorithm 2 Expanded Dual Expert Algorithm

Set $W_1^i = W_2^i = 0.5 \; \forall i = 1:M$
The parameter is in section n
Choose selection a from $\text{argmax}(W_1^n, W_2^n)$
if $W_1^n = W_2^n$ **then**
  Choose selection randomly
end if
if Error **then**
  $W_a^i = W_a^i/2, \; \forall i = \begin{cases} n:M & \text{if } a = 1 \\ 1:n & \text{if } a = 2 \end{cases}$
else if Correct **then**
  $W_a^i = \begin{cases} 2W_a^i & \text{if } W_a^i < 0.25 \\ \sqrt{W_a^i} & \text{if } W_a^i \geq 0.25 \end{cases} \quad \forall i = \begin{cases} 1:n & \text{if } a = 1 \\ n:M & \text{if } a = 2 \end{cases}$
end if

A. Performance Analysis

In order to discuss the performance of the algorithm we will make the following additional assumptions. We assume that there is a section index m so if $n \leq m$, then the preferred output is 1, and if $n > m$, then the preferred output is 2. Since preference is dependent on sections and "deviations" are defined as when the output varies from the concept preference, we must allow deviations to be dependent on sections. Define δ_n as the number of deviations in section n. Thus the total number of deviations denoted as Δ is $\Delta = \sum_{n=1}^{M} \delta_n$. Under this assumption, the error bound of the EDEA is given below:

Proposition 4.1: The upper bound on the total error for the EDEA algorithm is $E = (M+1)\Delta + M$.

Proof: There are two potential sources of errors in the EDEA algorithm. Consider section n. An error that occurs when a deviation occurs will decrease the ratio of $\frac{W_p^n}{W_d^n}$ by a factor of 2 in the section where the error took place. Each error that occurs without a deviation will increase the ratio of $\frac{W_p^n}{W_d^n}$ by a factor of 2 in the section where the error took place. Each error that occurs can affect the ratio in a maximum of M sections and a minimum of 1 section.

The maximum number of errors that can be caused by deviations is the total number of deviations Δ. Since each error due to a deviation can effect a maximum of M sections, the ratio of $\frac{W_p^i}{W_d^i}$ for $i = 1 \ldots M$ would decrease by a factor of 2^Δ. In order to increase this ratio to be greater than 1, the ratio in each section must have an increase $\Delta + 1$ times.

In the worst case, each of these increases would be caused by an error made without a deviation. And in the worst case, the ratio in each section is brought to be greater than 1 individually. Since there are M sections this means that there are at most $M\Delta + M$ errors that occur without a deviation. Therefore the bound on the error is $E = (M+1)\Delta + M$. ■

This error bound can be tightened to $E = (\max\{m, M-m\} + 1)\Delta + M$ if the section m where the preference changes is known. Meaning one deviation can only decrease the weight of the preferred output in $\max\{m, M-m\}$ sections.

B. Consistency Analysis

In EDEA, because weights within sections are updated in the same way as weights are updated in DEA, the ratio of weights in a section determines how many deviations in that section can be ignored. The consistency also has a dependence related to parameter. Suppose that the same selection is preferred in sections w and s.

Proposition 4.2: If $|m-w| > |m-s|$ then $\frac{W_p^w}{W_d^w} \geq \frac{W_p^s}{W_d^s}$.

Proof: From EDEA, if W_p^s is increased, then W_p^w is also increased. And if W_d^s is decreased then W_d^w is also decreased. But an increase of W_p^w does not guarantee an increase of W_p^s and a decrease of W_d^w does not guarantee a decrease of W_d^s. Thus W_p^w/W_d^w increases if W_p^s/W_d^s increases, but is not guaranteed to decrease if W_p^s/W_d^s decreases.

Since $W_p/W_d = 1$ in all sections before the first iteration. It is true that $\frac{W_p^w}{W_d^w} \geq \frac{W_p^s}{W_d^s}$. ■

C. Adaptiveness Analysis

Because the increase of weights within each section is bounded similar to the bound in DEA, there will be the same bound on the number of errors needed to change selections. Proposition 4.2 showed that if $|m-w| > |m-s|$ then $\frac{W_p^w}{W_d^w} \geq \frac{W_p^s}{W_d^s}$. This also implies that adaptiveness has a dependence related to the measured variable, and sections near m will be more adaptive than sections near $n = 1$ or $n = M$.

V. Simulation

A. Dual Expert Algorithm

1) Setup: The learning of a concept C by the DEA, WMA, and Winnow algorithms was simulated. C was created to chose one of two outputs in each iteration of a trial. This output was randomly generated using 5%, 10%, and 30% probability of choosing the deviation. After 150 iterations, drift was added by changing which output was the deviation while keeping the probability of making a deviation constant. This was repeated for 50 tests and the results were averaged.

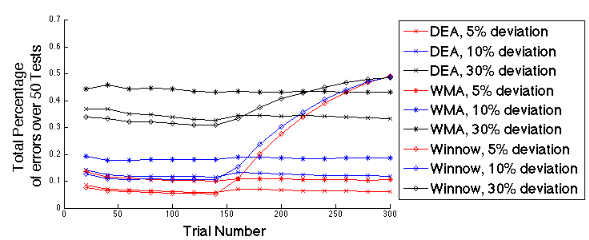

Fig. 2. Total percentage of errors

2) Results: Figure 2 shows that before drift DEA's error rate is in the same range as Winnow. After drift DEA's error rate remains constant like WMA's error rate, while Winnow's increases. Additionally figure 3 shows DEA's switch rate is close to Winnow, other than a spike at the time of drift, and decreases towards 0, before and after drift, as the number of trials increases. Thus DEA balances the consistency of Winnow and the adaptiveness of WMA.

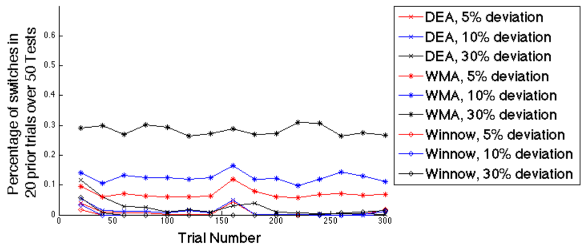

Fig. 3. Percentage of switches for each group of 20 trials

B. Expanded Dual Expert Algorithm

1) Setup: The learning of a concept C by 5 different algorithm was simulated. At the start of the trial the section m, where the preference changes, was set to 7. Each iteration could take place in one of 10 sections, randomly selected with equal probability. And the output of C was randomly chosen using the preference of the selected section and a constant probability of deviation. After half of the iterations (150) drift was simulated by changing m to 3. This was repeated for 50 tests and the results were averaged.

The algorithms used to learn concept C are the EDEA, the WMA, the Winnow, the DEA applied to each section individually, the DEA applied with no regards to sections, and the DEA modified to use more than two experts (multi expert). Both the Winnow and WMA are created according to their traditional multi-expert use [8], [9]. And the multi expert algorithm uses experts in the same way as the WMA and Winnow, however the update of weights is limited by the weight of the selection. The experts used in all three algorithms predicted 1 while $n < i$ and 2 otherwise. Since i is an integer between 1 and M +1, 11 control laws were created. One of which would exactly match the concept preference.

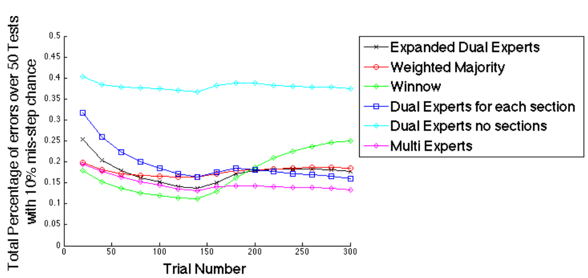

Fig. 4. Total percentage of incorrect selections.

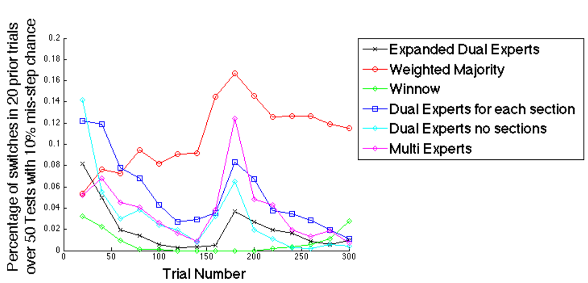

Fig. 5. Percentage of switches for each group of 20 trials.

2) Results: Figure 4 shows the average total rate of errors. Before drift occurs EDEA can be seen to be decreasing quickly, approaching the lowest error rate algorithm, Winnow, along with the multi expert. After drift occurs Winnow's error rate dramatically increases. EDEA's error rate moderately increases, into the range or WMA's error rate. And the multi expert algorithm remains relatively constant.

Figure 5 shows that the average rate of switches. EDEA and Winnow both have a small rate of switches that decrease towards 0 as time increases before drift. After the drift, the rate of EDEA switches spikes but then resumes descending towards 0. Winnow's switch rate also spikes, but is delayed in comparison to EDEA. In addition the multi expert case spikes at the same time as the EDEA, but to a switching rate approximately four times as large.

The continued good performance both before and after drift in the EDEA, and the limited spike in switching at the time of drift supports the fact that it manages the balance between consistency and adaptiveness well.

VI. Experiment

To test the applicability of the EDEA in an embodied robot we implemented it onto a Turtlebot passing a human in a hallway. The concept it was trying to learn was the internal beliefs of the human that determined the wall they approached when avoiding the robot. The parameter was the distance the human was from the right wall. The two selections that the Turtlebot could make was if the human would pass by deflecting to the left or right. It then acted by implementing the control law that moved it towards the opposite wall.

The Turtlebot was chosen for this experiment since it satisfied several important conditions. Its footprint is similar to that of an averaged human. Hence it is able to pass in a typical hallway setting. It can move at a speed that is comparable to the averaged casual walking speed of a human. It is equipped with a kinect and necessary software that can identify an approaching object and its movements to the left or to the right. And it can use obstacle avoidance behaviors that can be modified to adjust the avoidance direction.

A. Setup

The experiments were performed using a single Turtlebot that started centered in the hallway. The Turtlebot continued down the center of the hallway until it detected a human approaching. Then it selected a control law to avoid the human. While it was moving towards the wall it continued to watch the human to see if its selection was correct. It then used this information to update algorithm 2.

Some limitations were found during the implementation of the program. Most notably the fact that the noise from the depth based human detection made it difficult to identify human error quickly from a simple jerky motion. It was however usually able to identify errors when the human continued moving towards their intended wall.

Each test was set up with 10 iterations. The human initial location was arbitrarily chosen. We placed a marker on the floor so that the human could determine if they started in one of the 6 lanes with a left preference or one of the 4 lanes with a right preference. The tests were repeated for 4 times.

B. Results

The following table compiles the averaged result over all four tests along with average results from simulation data. It includes the rate of errors, and switches. The lower portion of the table compares the average of the four tests with simulation results for 10% and 30% error rates, that were 10 trials long and averaged over 50 tests.

Test	Error Rate	Switch Rate
Experiment average	35.0%	12.5%
Simulation 10% error rate	27.2%	10.7%
Simulation 30% error rate	42.2%	12.1%

The average of the tests' error was within the error bound of the 30% simulation error. While the average number of switching was only slightly higher than the 30 % error bound. This shows even with the difficulties inherent in perception the EDEA could still have a bounded error and switch rate.

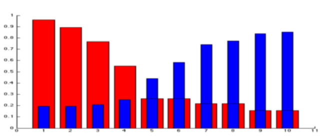

Fig. 6. Four run average of weights for selection of human turn direction. X-axis is hallway position. Red bars are the weight for the human passing via the left wall and blue bars are the weight for passing via the right wall.

Figure 6 demonstrates the increased difference in weights near the walls which is key for EDEA's consistency, and the decreased difference in weights near the switch point which is key for the EDEA's adaptiveness.

Overall the robot did learn the human's preference which was consistent throughout the tests. The number of errors and switches is consistent with simulation. And the weight ratio displays the same spatial dependence as selected. This means that EDEA transfers well to physical implementation.

VII. Conclusions

This paper has shown that a balance between adaptiveness and consistency can and should exist in robotic systems. And we show that the weight adjustment of the DEA and EDEA allow those algorithms to have that balance.

Increasing the applicability of these algorithms could be accomplished by expanding them to consider additional parameters. As seen in the EDEA expansion, increasing the number of variables could also help with decreasing system noise. These algorithms can also be more tailored to robotic systems by increasing the number of output selections. More closely resembling a behavior based robotic system.

VIII. Acknowledgments

The authors would like to thank Aida Khosroshahi of San Jose State University and Chris Paucar of Georgia Institute of Technology for their work on the Turtlebot test-bed.

References

[1] H. Huttenrauch, A. Green, M. Norman, L. Oestreicher, and K. S. Eklundh, "Involving users in the design of a mobile office robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 34, no. 2, pp. 113–124, 2004.

[2] B. Mutlu and J. Forlizzi, "Robots in organizations: the role of workflow, social, and environmental factors in human-robot interaction," in *3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2008, pp. 287–294.

[3] C. Breazeal, "Social interactions in hri: the robot view," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 34, no. 2, pp. 181–186, 2004.

[4] K. Dautenhahn, S. Woods, C. Kaouri, M. L. Walters, K. L. Koay, and I. Werry, "What is a robot companion-friend, assistant or butler?" in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005, pp. 1192–1197.

[5] L. Kuncheva, "Classifier ensembles for changing environments," in *Multiple Classifier Systems*, ser. Lecture Notes in Computer Science, F. Roli, J. Kittler, and T. Windeatt, Eds. Springer Berlin Heidelberg, 2004, vol. 3077, pp. 1–15.

[6] R. Polikar, "Ensemble learning," in *Ensemble Machine Learning*, C. Zhang and Y. Ma, Eds. Springer US, 2012, pp. 1–34.

[7] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," in *Foundations of Computer Science, 1989., 30th Annual Symposium on*. IEEE, 1989, pp. 256–261.

[8] N. Littlestone, "Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm," *Machine learning*, vol. 2, no. 4, pp. 285–318, 1988.

[9] A. Blum, "Empirical support for winnow and weighted-majority algorithms: Results on a calendar scheduling domain," *Machine Learning*, vol. 26, no. 1, pp. 5–23, 1997.

[10] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[11] Y. Freund, "An adaptive version of the boost by majority algorithm," *Machine learning*, vol. 43, no. 3, pp. 293–318, 2001.

[12] A. Tsymbal, "The problem of concept drift: definitions and related work," *Computer Science Department, Trinity College Dublin*, vol. 106, 2004.

[13] J. Z. Kolter and M. A. Maloof, "Dynamic weighted majority: An ensemble method for drifting concepts," *The Journal of Machine Learning Research*, vol. 8, pp. 2755–2790, 2007.

[14] K. Crammer, Y. Mansour, E. Even-Dar, and J. W. Vaughan, "Regret minimization with concept drift." in *COLT*, 2010, pp. 168–180.

[15] G. J. Ross, N. M. Adams, D. K. Tasoulis, and D. J. Hand, "Exponentially weighted moving average charts for detecting concept drift," *Pattern Recognition Letters*, vol. 33, no. 2, pp. 191–198, 2012.

[16] K. Severinson-Eklundh, A. Green, and H. Hüttenrauch, "Social and collaborative aspects of interaction with a service robot," *Robotics and Autonomous Systems*, vol. 42, no. 3, pp. 223–234, 2003.

[17] S.-Y. Chung and H.-P. Huang, "A mobile robot that understands pedestrian spatial behaviors," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2010, pp. 5861–5866.

[18] ——, "Predictive navigation by understanding human motion patterns," *International Journal of Advanced Robotic Systems*, vol. 8, no. 1, pp. 52–64, 2011.

[19] E. A. Sisbot, L. F. Marin-Urias, R. Alami, and T. Simeon, "A human aware mobile robot motion planner," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 874–883, 2007.

[20] E. Pacchierotti, H. I. Christensen, and P. Jensfelt, "Design of an office-guide robot for social interaction studies," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 4965–4970.

[21] ——, "Evaluation of passing distance for social robots," in *The 15th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2006, pp. 315–320.

[22] M. Lauckner, F. Kobiela, and D. Manzey, "hey robot, please step back!-exploration of a spatial threshold of comfort for human-mechanoid spatial interaction in a hallway scenario," in *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2014, pp. 780–787.