ATE (Automated Testing Equipment) Drift Prediction

Yating Chang, Wei-Hsiang Chu, Cheng-Hao Tsai
University of California, Davis
cyating@ucdavis.edu, ruchu@ucdavis.edu, chhtsai@ucdavis.edu

Abstract

The performance of Automated Test Equipment (ATE) can be affected by the test drift under the long-term operation since the environmental conditions such as temperature, humidity, and the equipment aging. It will also affect the chip yield classification's accuracy. We use the Double Q-Learning algorithm to deal with the challenging problem of optimal timing for recalibration. This algorithm can determine when to recalibrate. Moreover, we also design the simulation environment to simulate drift accumulation and contamination events.

In the experiment, we compare the classification result between the strategy based on the learned Q value and the fixed policy (baseline to always choose to test as next action instead of conducting any recalibration). The result shows that the learned strategy can avoid recalibration too early or too late. Moreover, the classification precision and stability following the learned strategy are also better than the baseline. This experiment shows that RL can automatically learn the appropriate control strategy under uncertainty and long-term testing scenario. It's helpful to improve testing performance, reduce the mistake classification rate, and minimize the overall cost.

1 Introduction

In the modern semiconductor manufacturing procedure, the ATE is very important. Since it can evaluate if the chip satisfies the production specification. However, the ATE can also be affected by the environmental issues such as temperature, humidity, and aging as the equipment operates overtime. It will cause the output drift, increase chip testing error, and decrease yield. So, recalibrating ATE during the appropriate timing is a challenging control problem.

Generally, the recalibration is usually implemented by regular period or based on human experience. However, these strategies are difficult to deal with the changing equipment status. It will cause the error testing since recalibrating too early or too late. In our project, we use the Double Q-Learning to automatically learn optimal timing for recalibration according to drift value. In this way, it can improve the chip classification accuracy and ATE testing performance.

We design the simulation environment in this project. We write a model for temperature, humidity, aging, and sudden contamination events as random variables that can affect the drift value. Moreover, we also design a reasonable reward logic to guide the RL agent to learn the strategy that can balance the test accuracy and the cost of recalibration. By comparing with the baseline strategy, the experiment result shows that the learned decision-making strategy has improved significantly in many aspects. It also shows the strength of RL in the semiconductor testing automation area. Furthermore, the environment parameters such as temperature, humidity, equipment aging are configurable. Currently, the environment parameters default to a common setting. But it can easily change to be more suitable for real-world ATE setting if needed.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022).

2 Background

The Q-Learning is a kind of value-based learning method in RL. It basically estimates the long-term reward for each action by building the Q value table for (state, action) pairs. The traditional Q-Learning causes overestimated bias since the max operator. The Double Q-Learning uses two independent Q table (Q1 and Q2) to deal with this issue. The Q1 and Q2 are responsible for action selection and value estimation. In this way, it can reduce the estimation bias and improve the stability of the learned policy.

During chip manufacturing, ATE is responsible for measuring if the chip meets the design specification. The measurement drift would be caused by fluctuating temperature, fluctuating humidity, and equipment aging. If we don't recalibrate it on time, the good chips will be incorrectly rejected and the bad chips will pass undetected. Generally, the ATE usually be recalibrated based on a fixed period or threshold, but these rules can't dynamically adapt environment changing. For example, the drift will accumulate very fast in the high temperature and high humidity environment. In this situation, the fixed strategy will no longer be usable. If we can make ATE to use RL method to automatically learn the optimal recalibration timing according to the practical situation, it will improve the testing performance. So, our project mainly applies Double Q-Learning on ATE control strategy. By simulating the environment to model the drift accumulation and contamination events, the RL model can learn the efficient recalibration strategy in the long-term operation. It will further improve the chips classification accuracy and reduce the error rate.

3 Methodology

3.1 ATE Environment Modeling

In this project, we design the specific environment model to simulate the practical ATE drift during the long-term operation. In the environment, it randomly initializes environmental parameters including temperature (20-30°C), relative humidity (30–60% RH), and equipment age (default to 5 years) and using this condition to estimate the equipment's current drift trend. Moreover, the drift increases proportionally to these three parameters, and we also add the Gaussian noise to simulate the sensor uncertainty. Furthermore, we also add the randomly contamination events with a very low probability to simulate unexpected real-world situation. Since it causes the testing inaccurate. If the drift exceeds 100 μV, it will be considered as failed and finish the simulation round. In this environment, we can select to continue testing or recalibrate immediately. If we select to continue testing, the drift will increase overtime. If we select the recalibrate immediately, it will reset all environmental parameters and it will generate the positive or negative reward according to recalibration timing (too early, on time, or too late). It has a small negative reward to reflect the associate throughput loss since recalibration need to take some time and temporary stop chip testing. In this way, it can gradually learn the optimal timing to control the drift.

3.2 Chip Generating Model

In our project, we design a chip generator to simulate practical chip sample testing. There are three testing parameters for each chip to represent their quality : output voltage, leakage current, and logic delay. Each chip is generated within these three parameters in separate periods, and they can simulate the different batches of chip's yield rate by adjusting their means and standard deviations. We set that if the chip's output voltage between 2.8-3.3 V, leakage current is less than 20 μA, delay is less than 5 ns, it will be considered as a good chip (label = 1). Otherwise, it will be considered as a bad chip (label = 0). By using this chip generator, we can generate the large batch of chip dataset with different quality. And we can use these chips on training stage and testing evaluation. Besides, the measured values of these chips are affected by ATE drift, so as the drift increases, each chip is more likely to get measured values that are more different from what they really are, which is the main problem causing the classification accuracy drops. Each chip is generated with randomly parameters. This makes sure the RL model can learn to make decisions under more realistic conditions.

3.3 Double Q-Learning Training

In this project, we use the Double Q-Learning as our RL algorithm to deal with the overestimation bias issue that traditional Q-Learning has during the maximum selection process. This method basically use two Q-value tables (Q1 and Q2). We randomly select one of the Q-table at each step for update. And we select the action based on the sum of Q-value from Q1 and Q2. In this way, it can ensure stable estimation and reduce the bias from overestimation. We initialize a new ATE environment for each training episode and select the action based on the current state, then update the Q table according to practical reward. We implement 100000 episode in total for training. We use the $\epsilon - greedy$ policy ($\epsilon = 0.3$) to encourage explore in the early phase,but we use $\epsilon = 0$ to conduct exploitation at the final episode. This action disables random action noise, and can also help us clearly observe the learned policy. We can see how the RL model behaves based on the Q-value estimation. The RL model will gradually learn the strategy that can balance the cost of recalibration and testing accuracy by repeatedly interacting with environment. At the end, we plot the first and last episode of drift-changing and recalibration timing in order to compare if the recalibration timing is learned and improved.

3.4 Model Testing

After training, we use three strategies to compare during the testing phase:

- Using the learned RL policy
- Using the fixed baseline policy (never recalibrates - baseline 1)
- Threshold-based Strategy (baseline 2)

During the testing process, we select the samples from chip dataset and send it into ATE simulation environment. Then, we simulate the whole testing process under the decision-making of the learning policy. And we add the extra disturbance according to the current drift level for the chip parameter to simulate practical testing bias. After testing, we evaluate the prediction result by checking if we successfully classify the quality of the chip. Lastly, we use the confusion matrices and classification reports such as precision, recall, and F1-score as the performance metric to compare their testing accuracy between the RL policy and baseline. According to this situation, we can verify if the RL policy can correctly control recalibration timing, reduce error rate, and improve the whole testing system's stability.

In order to compare with RL policy, we also design a threshold-based baseline policy to simulate the fixed threshold recalibration that exists in the industry today. This policy didn't need the learning process. It basically use the simple rule to make the decision. For example, when the ATE drift over the threshold such as 50 μV, it will implement recalibration. Otherwise, it will continuously conduct testing. In this strategy, we assume the user already know an appropriate drift boundary that can be used as a critical point of drift inaccuracy. During the chip testing process for each time, the system will monitor the current drift and decide whether to recalibrate according to the threshold. This method is simple and easily be used in the practical manufacturing procedure. However, it's lake of adaptive ability to deal with environmental changing and contamination events.

4 Results

In order to verify if Double Q-Learning recalibration strategy can improve the ATE testing accuracy and stability, we design two experiments : the first one is using the learned RL model, and the other is the fixed baseline strategy (never recalibrates). Both policies are tested by the same chip dataset. There are three parameters for each chip : output voltage, leakage current, and logic delay. And we also label them as good chip (label = 0) or bad chip (label = 1) in advance to compare the classification result.

During the training process, we also record the drift changing and recalibration timing for episode 0 and episode 99999 and then plot them. The green doted line in Figure 1 and Figure 2 means recalibrate too early. The blue line means recalibration on time between 50-100 μV. The red line means recalibration too late (drift over 100 μV). You can see that during the training early stage in Figure 1, the RL model is still driven by random exploration due to $\epsilon - greedy$ policy. It will cause the agent to choose to recalibrate frequently and most of the recalibration occur before the drift reaches the threshold. These recalibration are considered as too early, they are shown in Figure 1 by using green dotted line. There's almost no any drift over 100 μV so there's almost no any red line shown in Figure 1. In Figure 2, you can see that the model has successfully learned how to recalibrate when the drift accumulation reaches 50 μV (blue dotted line). It shows that the policy becomes more stable and successfully avoids the the cost that caused by too early recalibration or too late recalibration.

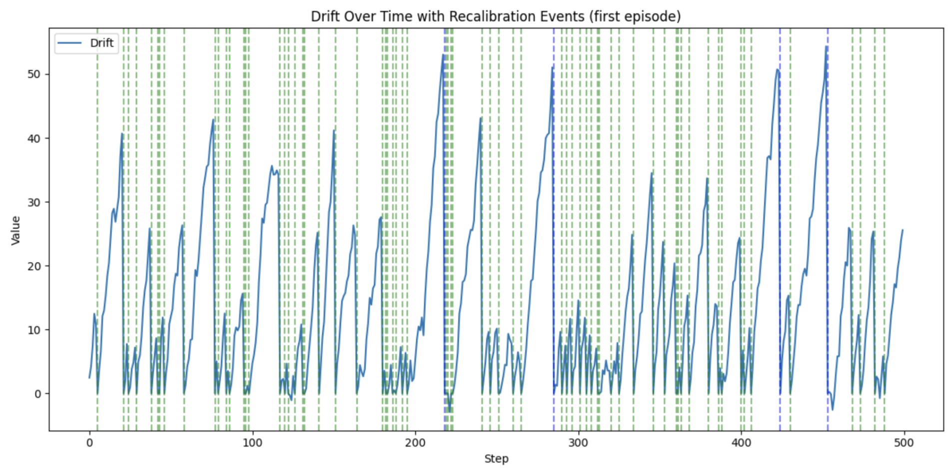

Figure 1: Episode 0 - strategy not convergence yet, recalibration frequency are too high and unstable

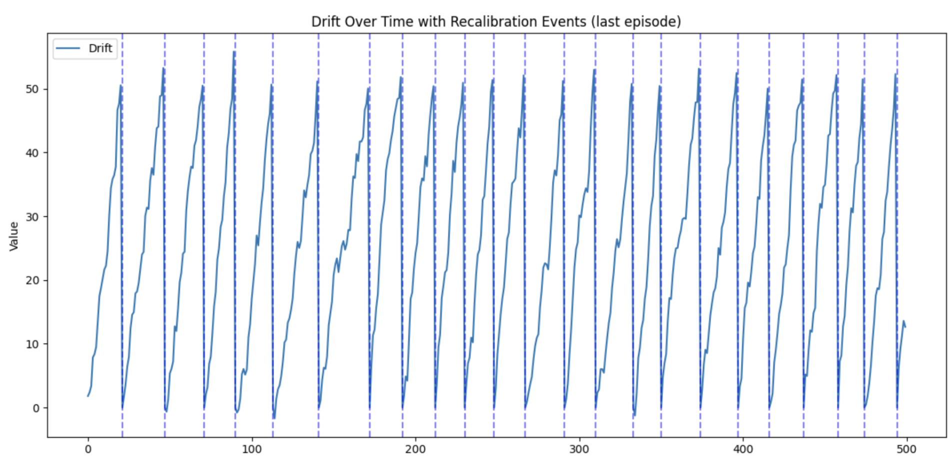

Figure 2: Episode 99999 - strategy have convergence, recalibration stable and located in the reasonable range

In order to compare the chip classification accuracy for different strategies, we simulate 10000 chips testing under two different policies : the first one is using the learned RL strategy, and the other one is using a fixed baseline strategy with no recalibration. And we use the confusion matrix and classification reports as evaluation metric. The model achieves 93% of the overall classification accuracy rate when we use the RL strategy. The recall rate of 88% for bad chips and the accuracy rate of good chips is 94% , and the overall F1-score is 0.93. For the fixed baseline strategy, the uncontrolled drift causes the incorrect classification rate to increase. So, the recall rate for bad chip is 68% , the accuracy for good chips reduces to 69%, and the overall classification rate is only 69%. Apparently, using the learned policy to recalibrate on time is much better than having uncontrolled drift.

In addition, we also design an industry-inspired threshold-based baseline for comparison. This policy will conduct recalibration when the drift over 50 μV to simulate the threshold-triggered recalibration logic in practical production. By using this method, it didn't need to learning process, it just set the properly drift threshold that has been known in advance. According to the testing result, the accuracy of threshold-based policy achieve 92%. The recall rate for the bad chip is 88%, the accuracy for the good chip is 93%, and the overall F1-score is 0.92. You can see that the RL policy is better than threshold-based baseline from the below table 1. Moreover, the RL policy also have better generalization and environmental adaptive. Furthermore, the RL policy can dynamically adjustment according to the drift changing rather than rely on fixed threshold. So, the RL policy can more flexible to deal with several condition including contamination events and different environment condition.

Table 1: Compare with chips' classification performance under different strategy

strategy	**Accuracy**	**Bad Recall**	**Good Recall**	**F1-Score**
RL strategy	93%	88%	94%	0.93
Threshold (baseline 2)	92%	88%	93%	0.92
Never Recalibrate (baseline 1)	69%	68%	69%	0.71

5 Discussion & Conclusion

By building simulation ATE drift RL framework, our project successfully verified that Double Q-Learning can be applied effectively to the learning recalibration timing. From the training process, you can see that RL model tends to randomly exploration since its strategy is unstable in the early stage. However, the RL model can precisely make the decision for the recalibration timing when the drift falls between 50-100 μV after training.

During the testing, we compare the classification accuracy and stability in three different kinds of policy including the learned RL policy, a baseline policy that never recalibrates, and a threshold-based policy that triggers recalibration when the drift exceeds 50 μV. The result shows that the accuracy in RL policy is 93%. It's better that the 69% accuracy in baseline policy and 92% accuracy in threshold-based policy. In the recall rate for the bad chips, the RL policy and threshold-based policy are 88%. But, the RL policy is the best in the good chips' classification, and its F1-score is also a little bit higher than threshold-based policy. Even though the result in threshold-based policy is very close to the RL policy, it rely on fixed drift threshold setting in advanced is lake of flexibility and generalization. However, the RL policy didn't need to set the threshold in person. It can automatically adjustment according to different environment and contamination events. So, it has the higher adaptive ability and generalization

6 Team members and division of work

- Yating Chang: Project idea, Generate ATE data, Initialize environment setting
- Wei-Hsiang Chu:Implement RL, Testing, Initialize environment setting
- Cheng-Hao Tsai: Implement RL, Integrate process, and Results