Investigation and Improvement of Test Item Delivery Mechanism in Computerized Adaptive Testing Systems

Harutyun, Terteryan

State Engineering University of Armenia
Yerevan, Armenia
e-mail: harroot@gmail.com

ABSTRACT

This paper studies the different mechanisms of test item delivery in adaptive testing systems. Description of drawbacks of available mechanism and suggestion of solution that overcomes the pointed issues are considered. The suggested mechanism is based on two-level caching and asynchronous programming methods and provides evidently valuable performance for the test item delivery process in adaptive testing systems.

Keywords

Computerized adaptive testing, test item delivery, web response time, cache, asynchronous programming, item response theory.

1. INTRODUCTION

Taking into account the growing potential of the Worldwide Web in the last decade, many tests of examinees ability have been modified for delivery on the Web. Typically, a number of items are being downloaded as a scrollable list; the examinee answers the questions, and then returns the completed page through the Web. As long the test item list is as much information will be transferred every time from client to server and vice versa. Web delivery of tests is the next step of evolution of test delivery systems; from the earlier conversion of the tests form paper-and-pencil to delivery by personal computers (PC). However, test PC delivery mainly doesn't affect the test standardization due to PC administration [1], excluding speed tests of course. The things are quite different in case of Web delivery of test items. Each test item in Computerized Adaptive Testing (CAT) systems is being selected based on the examinee's scored answers to all previous items, computations must be implemented after each item response is received to select the next item, and the item bank must be available to deliver that item. It might, therefore, be tempting to deliver an item over the Web, send the answer back to the server for scoring, maximum likelihood estimation and selection of the next item based on item information, and then transmit the selected item to the examinee through the Web. This process would then need to be repeated for every item. Item-by-item delivery of CATs through the web would have its negative impact on the test utility and validity because of the variation of Web response time. Item-by-item delivery of CATs through the Web would likely be a return to this approach of extremely unstandardized test delivery, thereby further compromising the utility and validity of test scores.

2. MATHEMATICAL MODEL OF ADAPTIVE TESTING

2.1 The Concept of Adaptive Testing

The disadvantages of classical test measurement methods are well documented. Numerous tests have been constructed over the years using these models. However, the drawbacks of these models are also well documented [2]. The main issue is that certain properties of the exam and the examinee are defined in terms of one another; one can only be understood relative to the other. For instance, a test item has a certain difficulty. That difficulty is defined by the percentage of correct answers from a group of examinees. However, their ability is the observed performance on a given test. Other properties of the test, such as the reliability of a particular item, are similarly defined. It's obvious that such a cyclic definition of these terms is at least undesirable.
Another problem of these models is one of comparison — attempting to compare ability scores of candidates who took different tests, or comparing item parameters obtained from different groups of examinees. These problems lead to an issue of reliability in the testing methods.
To solve these problems, an alternative measurement method is being used. The most accepted measurement method in computerized adaptive testing systems is called Item Response Theory.

2.2. Item Response Theory

IRT is a set of statistical models that define a test taker's ability using various methods. Statistics based on the items answered, existing knowledge of ability, and performance on each item, are all used to predict the examinee's ability scores.
Each model defines a monotonically increasing function known as the item response function or Item Characteristic Curve (ICC). The various models differ in the form of the function, including the number of parameters relating to the examinee. Provided that a model fits the test data, the aforementioned problems with classical testing procedures are addressed.
The main difference between each model is the number of parameters the ICC takes.

$$P_i(\theta_i) = c_i + \frac{(1 - c_i)}{1 + e^{1.7a_i(b_i - \theta_j)}}$$

Note that Pi(θj) is a simplified version for P(Ui = 1 |θj), where

$$U_i = \begin{cases} 0, i \text{ answered incorrectly} \\ 1, i \text{ answered correctly} \end{cases}$$

The most important parameter is the difficulty parameter of the question, defining the ability required to have a probability of success of 0.5. The higher the value of bi, the

greater the ability required to have a 50% chance of answering the given item correctly. θj is the ability of examinee j, and is measured in logits. The probability that an item, i, is answered correctly by examinee j is. There are other models for probability but tree-parameter model [3] allows for greater accuracy in separating examinees into different ability levels including the item guessing parameter.
Figure 1 shows a graph of the probability function for (ai, bi and ci) = (2, 0, and 0.25). As shown by the lines on the graph, the lower asymptote corresponds to the value of ci, and the point of inflection to bi. Parameter ai is related to the gradient at the point of inflection.

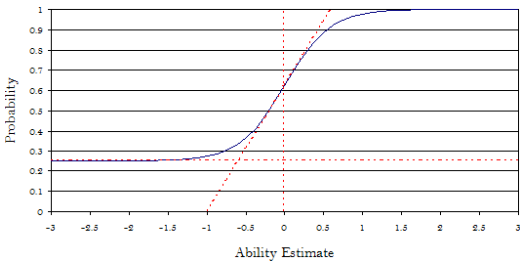

Figure 1. Probability function for (ai, bi, ci) = (2, 0, 0.25)

Probability function for (ai, bi, ci) = (2, 0, 0.25)
The item parameters can be described simply as follows:

- ai the discrimination parameter.
- bi the difficulty parameter.
- ci the guessing parameter.

It should be noted that these are merely descriptive names; in practice, guessing is a factor at the low end of the ability scale, but due to the design of most test questions, this parameter tends to be lower than the expected value from a random guess [5]. A more accurate name would be the pseudo-chance-level parameter. The most effective ranges for the parameters are ai ∈ (0, 2), bi ∈ (-2, 2), and ci ∈ [0, 0.35] [2].
IRT can be used to solve many real-world problems that classical measurement models cannot. For instance, it is difficult to compare scores from different tests using the classical measurement model. With IRT, simple linear equating provides suitably correlated results.

2.3. The CAT Algorithm

The CAT algorithm consists of the following steps:

- All items that are not yet administered and ranked to determine the most suitable item, given the ability estimate.
- This item is administered, and the system waits for a response from the examinee.
- A new ability estimate is calculated based on all the previous responses.
- If the stop criteria are met, the algorithm halts; otherwise, steps 1–3 are repeated.

It is assumed that we already have an item bank with calibrated test items (parameters are defined for each item in test item bank); the algorithm is used to calculate a value of θ that fits the model best. This is the ability estimate.
The Item Information Function (IIF) is used to determine the most suitable item for the current examinee. Intuitively, this function describes the amount of information a test item can provide about an examinee with a given ability. The equation bellow shows how the IIF is being defined:

$$I_i(\theta) = \frac{P_i'(\theta)^2}{P_i(\theta)\,Q_i(\theta)}$$

where Pi(θ) is the response function for item i, Pi'(θ) its first derivative with respect to θ, and Qi(θ) = 1 - Pi(θ). In CAT, the item is selected that has the maximum information in the item pool at θ = θ*, where θ* is the current θ estimate for the examinee [6]. Maximization of information minimizes the estimation error of θ. Suppose O is the set of items that have not been administered, then the most suitable item, a, is given by a = max {Ii | i ∈ O}. This is called Maximum Likelihood Method (MLM).
Suppose A is the set of items that have been administered (so Q ∪ A is the entire item bank). If the cardinality of A equals k (i.e. k items have been administered), then

$$\theta_{j,k+1} = \theta_{j,k} + \frac{\sum_{i \in A} S_i(\theta_{j,k})}{\sum_{i \in A} I_i(\theta_{j,k})}$$

where θj,k is the k-th estimate of ability for an examinee j. and Si is defined as follows:

$$S_i = \frac{(u_i - P_i)P_i'}{P_i(1-P_i)}$$

It should be noted that IRT is the most investigated and suitable measurement method for Computerized Adaptive Testing Systems, but not the only one. There are many other mathematical models based on which CAT System can be implemented (e.g. CAT based on Artificial Intelligence methods- neural networks, decision tree, CAT based on Bayesian Decision Theory, CAT based on Automat Theory, etc.). In spite of their diversity, they all have one similar trait – they all provide an algorithm for the most suitable test item selection. Item selection function is the most sluggish part of the entire Adaptive Testing algorithm. This article studies the test item delivery process which is common for all the CAT systems no matter which measurement method they are using underneath. The reason for demonstrating IRT algorithm in this article is to make the reader familiar with that measurement method to understand the reason why the item selection process is so durable.

3. TEST ITEM DELIVERY

Test item delivery method depends on the platform which is being used in the particular testing system. Accordingly there are two types of test item delivery methods, Web and Desktop. This article examines only the Web delivery, considering the issues regarding test item transfer from server to client and vice versa. It should be noted that Desktop delivery method is free of shortcomings that we talked about. As we have already mentioned, the most important issues with test item Web delivery in adaptive testing systems is **Web Response Time**.

No matter what kind of technology you use for test item delivery and presentation, the issue of test item transfer between the server and client, is still actual.
In case of the Computerized Adaptive Testing (CAT) the duration of test item selection process is also being added to the web response time (Figure 2).

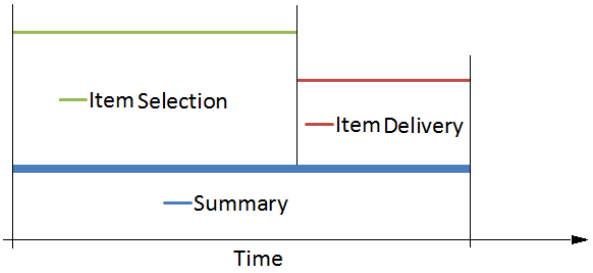

Figure 2. Duration of Test Item Delivery Process

So if we want to reduce the test delivery time we need to consider the duration of item selection procedure as well. In CAT systems test items are being delivered to the client using "item-by-item" principle. Every time the examinee submits his answer for the particular question the system performs the next most suitable item selection procedure and returns the next item to the client.

Issues related to Web response time are not something new in web delivery area; the most recent example could be delivering media content (images, videos and other media) from server to the client. It is particularly conspicuous at low network speed. So, how the modern web browsers have overcome this issue? The answer is simple: Cache. The Web browser engine actually downloads more content than it shows to the client, and while the client watches the previously downloaded content it starts downloading the content that is coming next. This caching mechanism ideally fits the CAT algorithm requirements. As you already know from the sections above; item selection mechanism depends on the examinee answer and his current knowledge level (θ see in section 2.3 The CAT Algorithm). If the examinee's answer is wrong the CAT item selection module will select the next item for lower θ and for higher one, if the answer is correct.

4. TEST ITEM CACHING MECHANISM

Suppose we have already a calibrated test item bank and the system is ready for organizing the actual testing process. Based on the CAT algorithm, one of the test items (Test Item 1) has been selected as a first test item to be shown to the examinee. Without any caching mechanism the following steps are being taken (Figure 3).

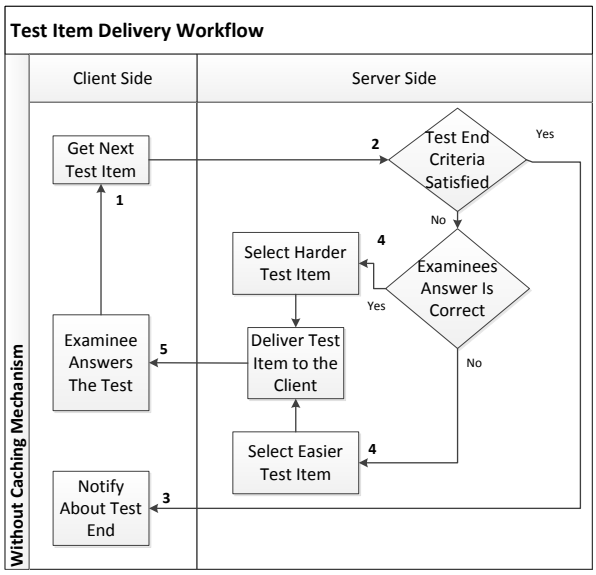

Figure 3. Test Item Deliveries without Caching

1. Submit examinee's answer to the server
2. Validate examinee's answer
3. Apply CAT algorithm to select next test item from the item bank
4. Deliver selected test item to the client.

All the above mentioned steps are durable processes, especially 3-th and 4-th (the duration time is directly proportional to the test item count in the item bank).

As you can see every time when a new item needs to be delivered to the client a new request is being sent to the server to select most suitable test item for the examinee's ability level. So in order to reduce the overall duration of the test item delivery process we need to reduce the duration of the following two processes.

1. Regular test item selection from item bank
2. Selected item delivery to client

Test item caching mechanism that is being suggested in this article is based on two-level caching method cache on server and on client side.

Figures 4.1 and 4.2 demonstrate the same workflow but this time CAT system has a Caching Mechanism applied. Figure 4.1 shows the case when the testing process has just started and the Client and Server Caches are empty.

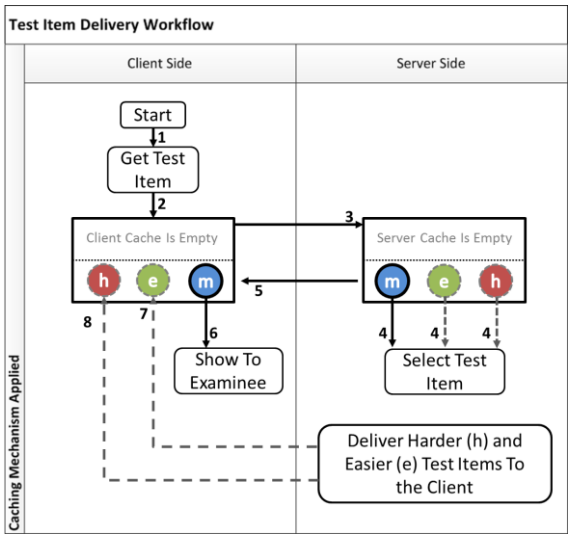

Figure 4.1. Test Item Deliveries When Cache Is Empty

As you can see on the flowchart above, the client side calls the server side method every time when it needs a new test item to show to the examinee. Pay attention to the process order, when the Client Cache is requesting the corresponding test item from the Server Cache (process 3), the server side starts 3 parallel tasks and tries to select medium, harder and easier test items simultaneously using asynchronous programing. Server Cache manager simultaneously imitates two opposite processes; it supposes that the examinee has already given the wrong answer to the current selected item and using CAT algorithm selects an easier test item for him. At the same time, cache manager selects the harder test item from the item bank just like if the examinee would have given the right answer for the current test item. Just after the medium test item has been selected the Server Cache saves it on its side and sends the test item to the client (process 5). Meanwhile the examinee is busy with the medium test item; the system starts downloading two others (harder and easier test items) asynchronously (processes 7 and 8).

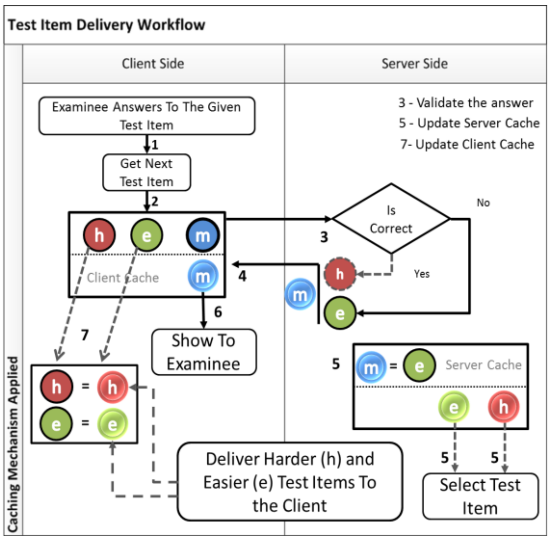

Figure 4.2. Test Item Deliveries with Initialized Cache

The above mentioned process is taking place every time when someone is requesting the current test item from the server. Instead of getting the test item from an item bank directly the Server Cache Manager updates the cache and responds with the current test item. To keep the diagram simple the test ending condition checking has been removed. Pay attention to the process 4 on the diagram above, based on the examinee's answer; the system selects whether harder or easier test items (in the diagram the examinee's answer is not correct, the easier item has been selected). The selected item is being considered as a medium item for the next test item series and sent to the client. Meanwhile the easier and harder test items are being selected for the next item series to update the Server Cache. Just after the selected test item has been shown to the examinee the Client Cache gets updated as well (Process 7).

5. PERFORMANCE IMPROVEMENT RESULTS

This section is describing the experiment results that have been performed during this research. The experiment is showing the performance improvements of test delivery process comparing two CAT systems: one with suggested caching mechanism and another without. The experiment has been performed on the server with the following parameters.

Windows Server 2008 R2 Enterprise	
Processor	Intel(R) Core(TM)2 Quad CPU Q8300 @ 2.50 GHz
Installed memory (RAM)	8 GB
System type	64-bit Operating System
Runtime Environment	
Web Server	IIS Version 7.5
Development Framework	Microsoft .Net Framework 4.5
Web Browser	Google Chrome Version 28.0.1500.71 m
Web Performance Analyzer	Fiddler Web Debugger v2.4.4.5

The experiment takes place with the following steps.

1. Enable logging in the application Item Selection method, this will make it available to observe duration of the item selection process.
2. Publish two different CAT systems under IIS web server.
3. Run the Fiddler Web Debugger.
4. Start Network Capturing for "GetNextTestItem" web method.
5. Repeat the web method call for 100 times and summarize the web response duration.
6. Analyze the collected results with the Fiddler Web Debugger tool.

The experiment results are shown on the following diagrams (Figures 5.1 and 5.2).

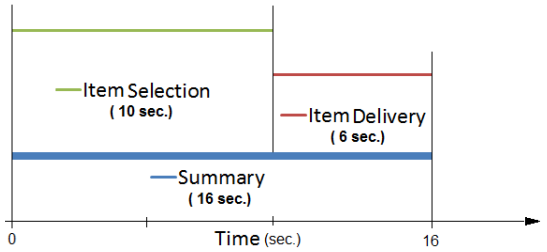

Figure 5.1. Item Delivery Performances without Caching

Figure 5.2. Item Delivery Performances with Caching

6. CONCLUSION

The article described a new mechanism of improving test item delivery performance in Computerized Adaptive Testing Systems. The new suggested mechanism is based on two-level caching mechanism and utilizes the elements of asynchronous programming. The suggested mechanism allows us to avoid the application performance issues during the test item delivery process. As a result of this research a new CAT application has been developed where the test item delivery mechanism is based on the suggested one.

REFERENCES

[1] A. Mead, F. Drasgow, "Equivalence of computerized and paper-and-pencil cognitive ability tests ", *A meta-analysis*, pp. 449-458, 1993.

[2] R. Hambelton, H. Swaminathan, H. Rogers "Fundamentals of Item Response Theory", *Sage Publications*, pp. 45-67, 1991.

[3] A. Birnbaum, "Some Latent Trait Models and Their Use in Inferring an Examinee's Ability", *Addison-Wesley*, pp. 25-56, 1968.

[4] FM. Lord, "A Theory of Test Scores", *Psychometric Monograph Vol. 7*, pp. 13-51, 1952.

[5] FM. Lord, "Estimation of Latent Ability and Item Parameters When There Are Omitted Responses", *Psychometrika*, Vol. 39, 1974.

[6] W. J. Van der Linden, C. A. W. Glas, "Capitalization on item calibration error in adaptive testing", *Applied Measurement in Education*, pp. 35-55, 2000.