

Interactive On-line Exercises of Basic Mathematical Functions

H. Nishizawa

Department of Electrical Engineering, Toyota National College of Technology, Eisei-cho 2-1, Toyota 471-8525, Japan

E-mail: nisizawa@tctcc.cc.toyota-ct.ac.jp

Abstract

In this paper, a system is described for serving on-line exercises of mathematical functions for high-school students. The system basically depends on WWW technology and uses WWW browsers running on students' computers for displaying questions, answers, and explanations. An exercise page has a graph of a mathematical function and requests a student to fill the text field with a mathematical expression appropriate for the graph. When the student clicks on the "evaluate" button, the expression is sent to the system server and compared with the answer expression. If the expression is not equal to the answer, graphs of both expressions, and the comment on how different his expression to the answer are displayed. Comparing the graphs and reading the comment, the student is able to continue his guessing work until he gets the correct expression. The exercises have been developed for students who have difficulty in understanding mathematical functions in the author's algebra course, who seem to lack graphical images of the functions they learned in the classes. They usually try to remember the written rules for converting a mathematical function to its graph and reverting it. Such attempts sometimes succeed in simple cases like linear functions but fails in quadratic functions or more complicated functions because the rules for converting those functions become too complicated just to remember. What they lack is rich experiences of handling real graphs.

Although recent development of electrical worksheets and symbolic computing program like Mathematica made it easy and quick for a student to draw graphs, it is only half the way to the understanding. Because the drawing is automated and includes usually few guessing works, the student does not necessarily have ideas about, for example, the effect of a coefficient value on the shape or the position of a graph. The only rigid knowledge he has is how he operates the program.

The on-line exercises provide with the other half. The first thing a student does in the exercise is a guess which tries to describe the given graph correctly. There is no

instructions of expressing the graph nor hints before the first guess. After the first guess is done, then comes the hint or comment which help him to make the next guess. This makes a kind of experiment on the desktop. A series of the experiments tell him the behavior of graphs in detail and the relation with the expressions. Adding to that, because the evaluation are done symbolically using Mathematica functions, wide variety of expressions are allowed, which is easier to accept for usual students. The exercise provides the students with opportunities to learn from their mistakes and to build their own theory about mathematical functions.

1 Background

The author has been developing a network-based learning environment ¹⁾ for the students in Kosen, College of Technology in Japan. It provides each student with an individual learning environment depending upon his personal data and learning history in the field of algebra ^{2), 3)} and in other fields ^{1), 4)}. As those data are stored in a database, the student can use the environment not only in classes but also in his self-learning after the classes. The on-line exercise is one of the services the learning environment provides. The on-line exercise helps slow learners when they learn the relationship between a graph of a mathematical function and the corresponding mathematical expression which is not easy to understand for them even if it includes only one independent variable and the graph is plotted in two-dimensions. In author's point of view, the difficulty comes from the poor experience in plotting real graphs by himself and in thinking about them. Those students tend to remember the formula or the solution for a certain question blindly and do not think or imagine the meaning of an expression and the effect on the graph. The on-line exercise should give them some practical experiences which lacks them.

2 System Layout of the Learning Environment

The learning environment the on-line exercise based on is composed of a WWW server and WWW clients connected to each other with a computer network (Fig. 1). The WWW server is a Macintosh computer, on which a HTTP server program, WebSTAR, a symbolic computational program, Mathematica, and a database management program, Butler are running. A Plug-in software component to WebSTAR, Tango, and a custom CGI program help the communication between them.

When a student sends data from his WWW browser to the server, the HTTP server program retrieves data from the database through the Plug-in component or calls a Mathematica function through the CGI. The Mathematica function evaluates the input according to the evaluation criteria to be described later, returns the comment to it, and produces a graph file in GIF format. A HTML text for the student's page including a graphic image, his expression, and the comment to it are produced on the fly and returned to the student's WWW browser. Client terminals may be any computers as long as WWW browsers run on them. The network is an Ethernet LAN in the campus and it is

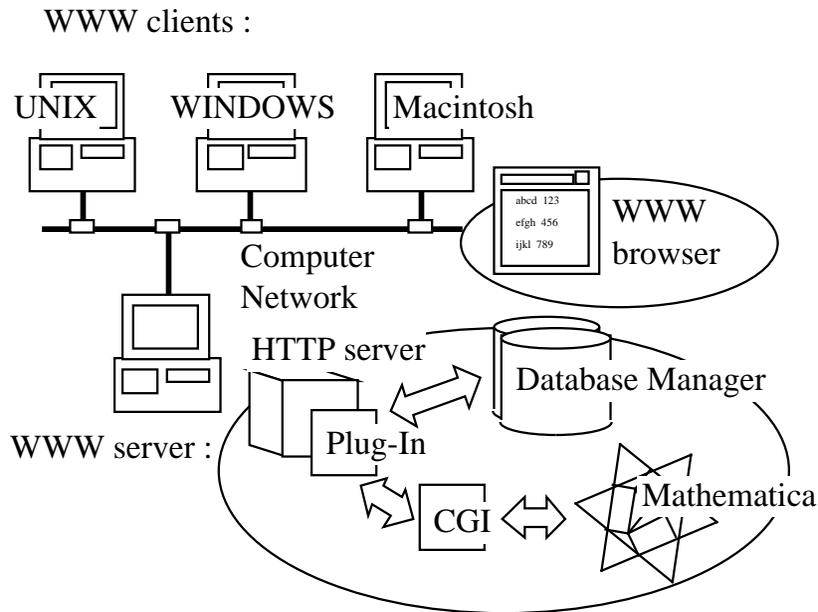


Fig. 1 Layout of the learning environment

also connected to the Internet. As a result, not only the students in the campus but also remote students can make the exercises through the Internet.

3 An Example of the Exercises

Fig. 2 shows an example of a student's page after he typed his first expression in the text field for describing the given function and requested the evaluation. The curved line in black color located left side of the page is the graph of a given function $y = f(x)$. The blue one is added to it by the evaluation process using the student's expression when it doesn't describes the function $f(x)$ correctly. There is only curved line in the graph if his expression describes the function correctly. Right to the graphs lies the student's expression with a comment to it. The comment is quite simple if the expression correctly describe the function. Otherwise, the comment is an explanation of how different the two curves. There appear also an input field for another expression and "evaluate again" button at the bottom of the page.

The student is allowed to request evaluations as many times as he needs after comparing the graphs on the screen and reading the comment. The graphs, expressions, and the comments increase in number until the last expression correctly describe the function. When he has correctly described the function, "next question" button appears on the screen instead of the "Evaluate again!" button for him to go to the next question. It is also possible to skip the current exercise or end the exercise anytime by clicking on the "Skip the question" or "End the exercises" button respectively.

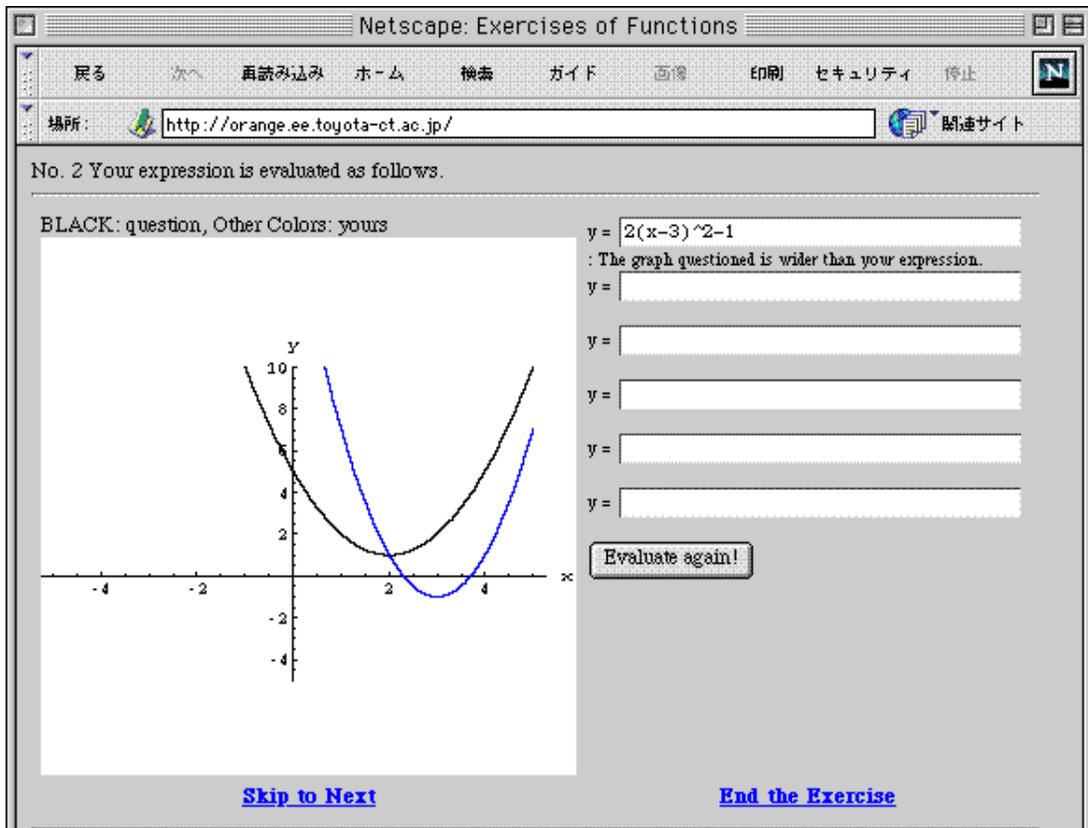


Fig. 2 A screen shot of a student's page after the evaluation

4 Data Processing on the Server

An expression sent from a student is processed on the server as follows. Every time a student requests a new question to the server (Step 1 in Fig. 3), WebSTAR, the HTTP server program sends a query to Butler, the database manager, through its Plug-in software Tango, the behavior of which is customized by its so called query documents (Step 2). The database stores question Ids, question types, answer expressions, and the location of graphic files which include graphs to be questioned. Retrieving those data (Step 3), a HTML text for a question page including an instruction text and a link to the graph file in GIF format is made up (Step 4), and is sent to the student (Step 5). The answer expression and the question type are saved in the variables managed by the Plug-in software to prepare the evaluation process (Step 6).

When the student returns an expression for describing the graph (Step 7), it is sent to Mathematica through the CGI program with the answer expression and the question type (Steps 8 and 9). The custom evaluation function for the question type is coded with Mathematica language. It compares the student's expression and the answer partly using external functions⁵⁾ (Step 10) and returns a comment to it (Step 11 and 12). If it finds the two expressions not equal mathematically, it plots two graphs representing the expressions, saves it into a GIF file (Step 13), and returns the location of the file, too. A HTML text including those data is sent back to the student (Step 14 and 15).

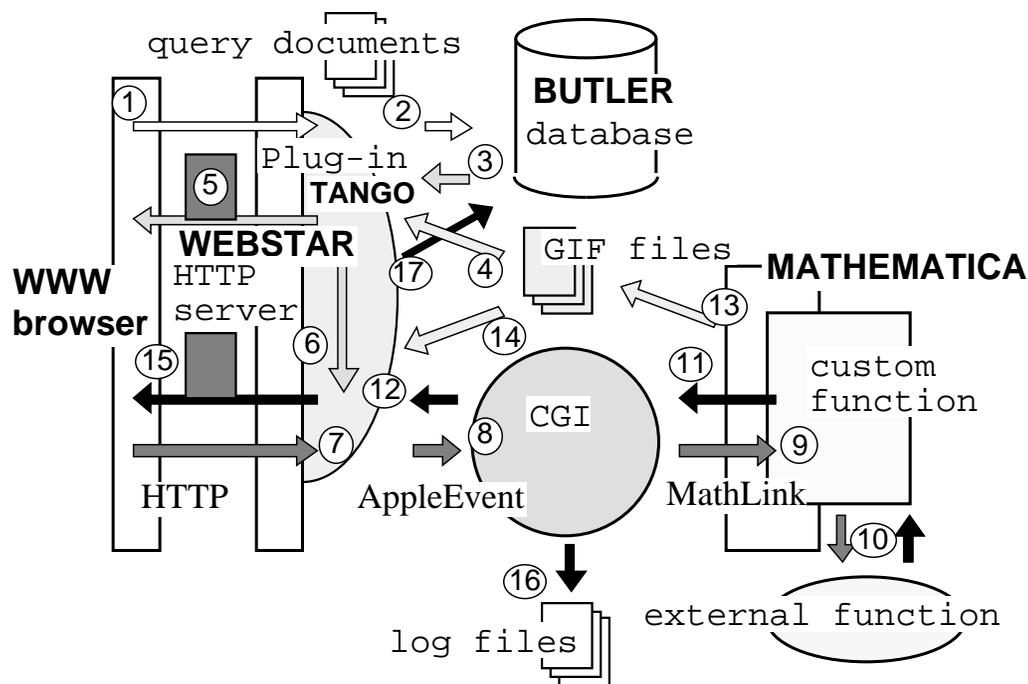


Fig. 3 Data Processing on the Server

During the process, his learning history is added to the log file (Step 16) and his data in the database is updated (Step 17).

5 An Example of the Evaluation

The evaluation of the student's expression uses mathematical knowledge and is done symbolically. Here is an example of the evaluation in an exercise of expressing a quadratic function.

Mathematical knowledge used here is as follows. When two quadratic functions of x , expressed in the form of $y = A(x - x_0)^2 + y_0$, are equal to each other, three parameters A, x_0, y_0 must be the same. The parameter A determines the sharpness and the direction of the graph. The graph makes a convex or concave curve if the parameter is positive or negative respectively. Larger absolute value of it makes the curve sharper. The parameters x_0 and y_0 represent the horizontal and vertical coordinates of the peak or the bottom point of the graph respectively, where x and y are the independent and dependent variables.

The evaluation uses these facts to compare the student's expression with the answer expression (Fig. 4). It firstly confirms the grammatical correctness of the expression. If the expression includes incomplete description like unpaired parentheses, operators with insufficient parameters, it is rejected at this moment and commented as "syntax error". The rejection is necessary because such an expression causes evaluation errors in the following steps. The student easily recognizes and corrects such errors if he is notified.

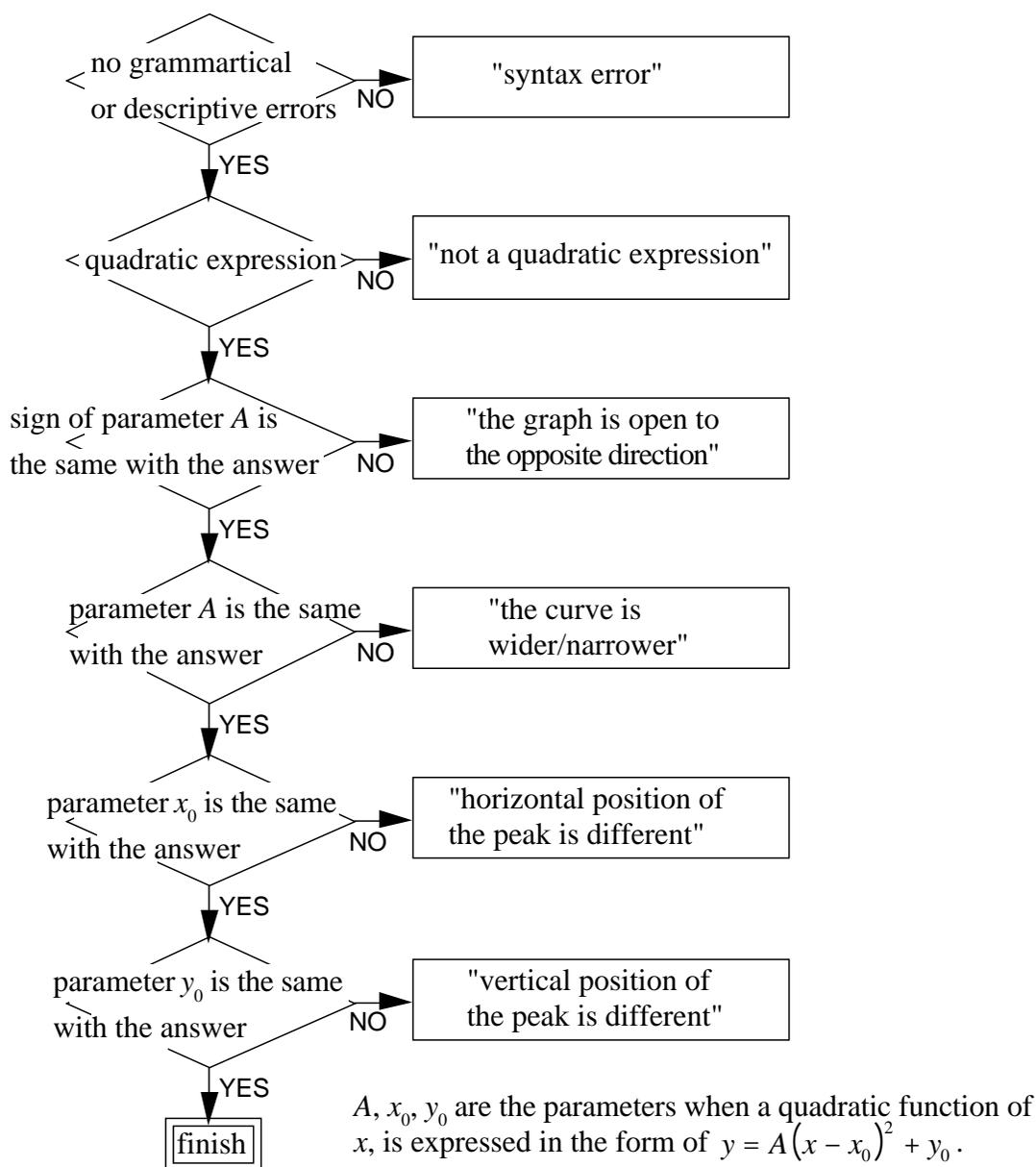


Fig. 4 Evaluation steps of an expression for a quadratic function

Next, a built-in Mathematica function `Exponent[f(x), x]` checks if it is a quadratic function or not. When the result is other than 2, the expression is not a quadratic function. Showing both graphs in the same figure helps the student to recognize the difference between them because a polynomial other than a quadratic expression makes quite a different shape of a graph.

The sign of the parameter A changes the direction which the graph is open to and the absolute value affects the sharpness of the graph. Built-in function `Coefficient[f(x), x, 2]` gives the value of parameter A , the coefficient of the term where the degree powers is two. Horizontal coordinate of the peak x_0 is where the gradient of the function with x is zero, which is given by the Mathematica

function $x_0 = \text{Solve}[D[f(x), x] == 0][[1, 1, 2]]$.

Vertical coordinate y_0 is easily calculated by substituting x_0 for x in $f(x)$.

When the expression $f(x)$ is not equal to the answer, a GIF file including the graphs of student's function $f(x)$ and the answer function $g(x)$ in the x values between x_{\min} and x_{\max} are produced by the function

`Display[file location, Plot[{g(x), f(x)}, {x, xmin, xmax}], "GIF"],`
and the file location is used to embed it in the HTML file for sending to the student. The expression passed through the all steps is evaluated to be equal to the answer.

6 Conclusion

The on-line exercise of mathematical functions for high-school students is described. It is supported by recent WWW technology and requests a student to find the expression correctly describe the graph displayed on his screen. The student's expression is evaluated at the server by a custom Mathematica evaluation function and the result is returned immediately. It is like an experiment on the desktop and provides the student with experiences on handling graphs and thinking of the rule governing the relationship between the mathematical expression and the graph. Because the exercise is personalized by the data stored in the databases, it is utilized not only in the classes of algebra but also in a self-learning of a student in the campus or at home.

References

- 1) H. Nishizawa, T. Saito, S. Pohjolainen : "An Implementation of a Hypermedia Learning Environment for a Small Group" Trans. of Japanese Society for Information and Systems in Education, (to be published)
- 2) H. Nishizawa, K. Nakayama, T. Kanaya : "A remote tutoring system for algebra enhanced by Mathematica", Proc. IMS'97 pp361-368 (1997)
- 3) H. Nishizawa, T. Saito, K. Inuzuka : "A Network-based, Remote Tutoring System for Algebra" in Japanese, Kosen Kyoiku, Vol.21, pp225-232 (1998)
- 4) H. Nishizawa, T. Saito, S. Pohjolainen : "Individual Learning of Technical English Words on a WWW-based CAI System" in Japanese, Kosen Kyoiku, Vol.22 , pp301-307 (1999)
- 5) H. Nishizawa, T. Saito, S. Pohjolainen : "An Analysis of Mathematical Symbolic Expressions by External Mathematica Functions ElemStr" in Japanese, Journal of Toyota National College of Technology, Vol. 31, pp55-59 (1998)