ISBN: 978-628-95207-8-1. ISSN: 2414-6390. Digital Object Identifier: 10.18687/LACCEI2024.1.1.1430

Intelligent system generator of numerical derivation exercises with gamification

Hugo Pablo Leyva, M. SC.[1], Rafaela Blanca Silva-López, Ph.D. [2]

[1]Universidad Autónoma Metropolitana Lerma, México, r.silva@correo.ler.uam.mx,
[2]Universidad Autónoma Metropolitana Azcapotzalco, México, hpl@azc.uam.mx

Abstract* – An action that is carried out in each school period of a class, such as numerical methods, consists of the generation of exercises. Exercises are required to be used as examples in class, assignments, or test items. It takes a considerable amount of exercises to foresee that students have the solutions in advance. The exercises will have their respective solution, which must be found in a period such as the one you have to solve an exam. This paper presents an exercise generator for the topic of numerical derivation. The developed software facilitates the generation of several exercises with their answer, such that students can solve in the interval provided to take an exam. A technique in which a course can be taught is gamification, which tries to establish a gaming environment in areas as varied as: education, business, among others. The exercises that are generated are of the topic of numerical derivation such that they have an interpretation that is given in a gamified context. The algorithms for the development of the software, its implementation and the results obtained for the generation of exercises used in the topic of numerical derivation for the course of Numerical Methods in Engineering of the UAM Azcapotzalco are shown.

Keywords-- e-learning, gamification, problem-based learning, numerical methods, numerical derivation, intelligent algorithms.

I. INTRODUCTION

In the engineering careers at the UAM (Universidad Autónoma Metropolitana) Unidad Azcapotzalco, as in other Higher Education Institutions (IES), there are subjects that are important, such as Numerical Methods in Engineering. An action that is heavy and demanding in this subject is the generation of problems that are required in class, to leave as homework and for exams. You have to have different problems for each group that you have, to prevent students from sharing the solutions with each other. At the same time, you have to solve them and have the answers to qualify them. This is a task that takes up considerable time.

Problem-based learning can be used in engineering and science classes such as Numerical Methods with good results. This course requires a high amount of problems. An option to make a difficult course such as numerical methods more entertaining is gamification [1]. By giving a playful context to the activities of the course, it allows you to catch the student's interest [2].

E-learning is an option to conduct the teaching and learning process that is widely used during the times of the 2020 contingency. It allows you to serve a large number of students using an LMS (Learning Management System). These softwares have features to handle a battery of problems, so that different versions of an exam can be used without being repeated. To have such a battery requires a vast number of problems.

At UAM Azcapotzalco, one of the topics of the Numerical Methods in Engineering course is numerical derivation. The analytic solution is impossible to find, as it applies to the case of knowing only one table of a function, where the derivative definition cannot be used [3]. The numerical solution is simpler, due to the nature of the calculations involved, although rather imprecise. This is why the software described in this introduction is of interest.

To have enough problems, you can search for them from different sources such as: websites, books or articles, however, it is difficult to have enough problems to use them in each school period, especially over the years. Another possibility is that if they become aware of the source of the problems, students will be able to access them from one year to the next, and the value as assessment tools will be lost. For this reason, intelligent algorithms were created and programs were developed for the generation of problems in such a way that: there is a considerable number of problems (thousands) to prevent them from being shared by the students, there is the solution for the proper feedback, they are generated in a reasonable time, they can be used for exercises, tasks or exams, as well as to supply the battery of problems that an LMS requires to apply the exams without repeating the reagents.

To comply with all these constraints, a series of intelligent algorithms were programmed to meet all the needs raised.

- Generate problems of the topic of numerical derivation of the subject of Numerical Methods in Engineering, along with its answer.
- Problems are interpreted in a gamified context.
- The number of issues is configurable.
- The parameters of the solution (maximum number of iterations, significant figures found, etc.) are configurable.
- Problems that are solved in a few iterations.
- Additional filters are applied to generate less demanding problems for students.
- A file is generated, if desired, that solves a battery of problems that is properly formatted for use with an LMS.

Several authors have touched on the concept of gamification, they consider that it is a technique that is used to achieve the motivation of people in activities of daily life [4], it is also about using play in unrelated contexts to influence the way people act [5].

Under these assumptions, gamification [6] encourages student learning, as well as growing interest and motivation to cover a particular topic [2].

Gamification in a course such as Numerical Methods can motivate students, since usually the teachers who give these courses use only mathematical problems as examples, which unless the student likes mathematics, does not motivate them to solve equations when the utility of the solution or its interpretation is not seen. Another option would be to use problems from their specialty, but since this course is taught at the beginning of the degree, the student has not seen enough of their career to understand the meaning of the solution. Currently, students are very immersed in a gaming culture, so if the problems are given an interpretation in a game context, it can be more motivating to solve the problems, because an interpretation can be given to the solution, in addition to being fun.

In [7] Exercises developed by expert teachers are presented, which are solved with numerical software to develop the teaching material. Problems are solved with Maple or Matlab programs. The theme is the elaboration of the material and its use for teaching.

In [8] exercises were designed using the Mathematica program with learning objects for the subject of Numerical Methods.

In [9] GeoGebra is used, this is a website where you have resources for learning math and science. The resources obtained are used in a numerical methods course used with the Moodle LMS.

In [10] exercises are prepared with the Excel spreadsheet to be used with the flipped classroom approach. The authors are interested in the implementation of their course rather than in the elaboration of exercises.

In [11] They prepare exercises based on textbook problems that employ the order-oriented adaptive learning approach to presenting material from a numerical methods course.

In [12] They use the e-status web tool with which they generate numerical problems. The problems they generate are related to the subject of statistics, rather than to that of numerical methods.

In [13] created a problem generator for Introduction to Chemical Engineering courses. The problems considered include random values, fixed values, and those based on real-life cases.

In [14] A repository of already created exercises is used. The exercises are from the discipline of chemical engineering, they are a collection of 10 numerical problems. The problems are typical of chemical engineers, where the solution is numerical. The solution is explained with several programs.

In [15] OptiA is used to solve mathematical programming problems, which are optimization problems. It is not intended to be a problem generator.

In [16] A software that allows the generation of mathematics exercises is presented. The exercises are solved with analytical, rather than numerical, methods.

Runge-Kutta methods are used to solve ordinary differential equations in such a way that they approximate the Taylor series development without using higher-order derivatives. Heavy algebra is required to obtain the coefficients of different explicit hybrid methods.

In [17], a generator of the coefficients necessary for these explicit hybrid methods is discussed. It does not mention how differential equation problems are generated to test these methods.

A similar case to the previous one is the solution of partial differential equations [18]. The authors solve problems using various software.

We did not find any article that shows intelligent algorithms, nor does it develop programs that produce problems that are solved numerically with their answers complying with the specifications and restrictions mentioned. The programs developed in this paper satisfactorily address these shortcomings.

II. METHODOLOGY

Algorithms were created that comply with the constraints set for the generation of numerical derivation problems. Considering the didactic technique of gamification in the teaching of numerical methods. A gamified use case is proposed in which numerical derivation is employed.

Case: To produce the potion.

As the potion has turned out so well in its functioning. The archaeologist's nephew has proposed to his uncle to enter the weekend dating business.

Modern researchers at the UAM, Dr. Silvana López and Chemical Engineer Pedro Lugo, have determined that in order to design a reactor to prepare the recipe for the potion, which will help them enter the weekend dating business, the derivatives of the concentration with respect to the age of the prospect are required. Calculates the requested derivatives.

See the attachment at this level. Use finite difference formulas. How many significant figures can the answers have at most? Of the calculated derivatives, sort them based on precision. (Consider order of derivation and order of differences) Which are more accurate?

If you were to go into business with them selling this potion, what should be the most appropriate methodology to estimate the derivatives and thus make sure that the potion was manufactured properly?

The functions from which the tables are generated have one of 2 forms:

y=a2*f(x)+a1*x^n+a0 (1)
y=a4*x^4+a3*x^3+a2*x^2+a1*x+a0 (2)

where:

f(x):

exp(Ax) (3)
exp(x) (4)
exp(-x) (5)
exp(x^2) (6)

exp(-x^2) (7)
ln(x) (8)
log(x) (9
sin(x) (10)
cos(x) (11)
sinh(x) (12)
cosh(x) (13)
tanh(x) (14)
sqrt(x) (15)
curt(x) (16)
1/x (17)
x^A (18)

Sqrt is the square root function.
Curt is the cubic root function.
The functions are chosen randomly.
a4, a3, a2, a1, a0, A: Randomly generated coefficients.
n: Exponent from 1 to 4, randomly generated.

With the function obtained, a table will be generated that will be provided to the student to find the derivatives that are requested.

Derivatives are of the order 1 to 4 randomly generated.

The generated tables are of the form:

TABLE I Format of the generated tables

X	Y
X_1	Y_1
X_2	Y_2
.	.
.	.
.	.
X_M	Y_M

where:

Xi,Yi: Table points generated with the above function. The index ranges from 1 to m.

m: Number of points in the table, randomly generated.

The points where you are asked to find the derivatives are the first, the middle, and the last in the table. This is so that the formulas for forward, backward, and central numerical derivation are applied.

The coefficients, and the exponent, are generated in such a way that the equations have values, when evaluated at a randomly determined interval, neither too large that they cause problems to handle, nor too small that they are considered zero. Repeated values are also avoided. Some functions such as ln(x) that have problems with negative values and zero are also considered to be avoided. Finally, with the equation obtained, equally spaced tables are generated.

The range in which random coefficients are generated varies in the range of -30 to 30. The exponents vary randomly from 1 to 4. The range of the table is randomly generated from 0 to 30.

The algorithm for generating the problems is shown in Figure 1.

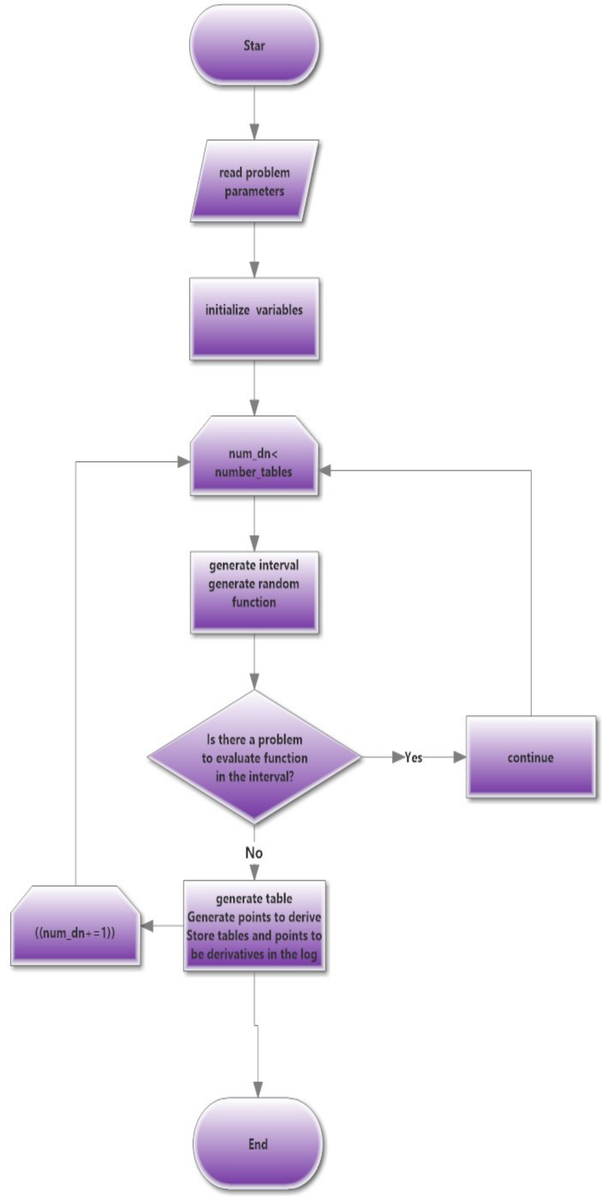

Figure 1. Algorithm for Generating Numerical Derivation Problems

In this flowchart, the parameters to be read at the beginning are:

number_tables: The number of problems to generate.
num_points: The number of points that the table can have.
num_digit: The number of digits with which the calculations will be performed.
limit_val: The range of values for tabulating.
name_dir: The name of the directory where the problems will be generated.
The variables to be initialized are:
num_dn: A counter that counts how many problems have been generated.

When the random interval is generated where the points of the table will be obtained, the limit is obtained with a maximum random value with respect to the limit_val parameter.

The random function is chosen randomly based on a value that varies from 1 to 18, which corresponds to the types of functions mentioned in previous paragraphs.

Then, as many random coefficients are generated as the randomly chosen function requires.

With the interval and the function already defined, it is validated if there are problematic points to evaluate the function as described in previous paragraphs.

If so, both the interval and the function are discarded to generate new intervals and functions.

This continues until there is a satisfactory interval and function.

Once there is a satisfactory interval and function, the table is generated.

The num_points parameter is used to randomly determine how many points the table will have.

The num_digit parameter is used to generate the points of the table to the desired number of digits.

The order of the derivative is randomly generated from 1 to 4, for the first point, the middle point, and the end of the table.

The above is stored in files in the name_dir directory.

The num_dn counter is incremented and it is validated if the number of problems to generate num_tables has already been reached.

The flowchart describes the process of generating random math problems.

The problems are stored in files in a directory.

The parameters of the problem generation process can be customized.

The technologies used to develop the program are:

- Rocky Linux for the server operating system.
- Shell scripts in Korn Shell for equation generation.
- FORTRAN language for problem solving.
- XML for the question batteries for the LMS.

The LMS used was Sakai [19], [20], [19], [21], [22], [23].

III. Results

The software allows you to generate a huge number of problems in excess of 1000. The time needed is around 45 minutes per 1000 problems. This results in achieving in a short time what was previously done in several days with a few problems.

A test was done with 5000 problems. Figure 2 shows the % of each function that was generated for the numeric derivation problems.

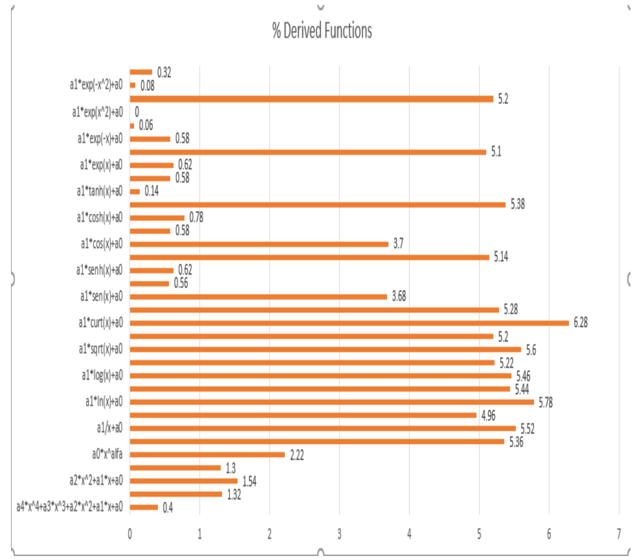

Figure 2. Percentage generated from each function of the numeric derivation problems.

From these results, it can be observed that the time required to generate many problems is acceptable, compared to the time it would take to manually generate a few problems. The 5,000 problems were generated in approximately 7 and a half hours.

The reason for the wide disparity in the types of functions obtained is that several functions have problems when evaluated in the generated interval. This is because the function cannot be evaluated in negative values, for example, or it falls into some of the cases mentioned above, where there would be tables with very small values practically zero, very large values that exceed what can be evaluated on a calculator, or tables with practically the same value for the entire table.

According to the results obtained, the function that does not degenerate in any of the mentioned cases is the cubic root.

Unfortunately, at the time, a survey was not applied to the students to find out their opinion on the problems that were applied to them during the course, such as homework, class exercises or exams, obtained with this software.

An example of the problems generated is

Case: To produce the potion.

As the potion has turned out so well in its functioning. The archaeologist's nephew has proposed to his uncle to enter the weekend dating business.

Modern researchers at the UAM, Dr. Silvana López and Chemical Engineer Pedro Lugo, have determined that in order to design a reactor to prepare the recipe for the potion, which will help them enter the weekend dating business, the derivatives of the concentration with respect to the age of the prospect are required. Calculates the requested derivatives.

TABLE II Concentration vs Age

X	Y
12.7967	0.338738
13.9541	0.296024
15.1115	0.261492
16.2689	0.233114

17.4262	0.209463
18.5836	0.189511
19.7410	0.172500
20.8984	0.157860
22.0557	0.145155
23.2131	0.134047
24.3705	0.124269
25.5279	0.115610
26.6852	0.107899
27.8426	0.100999
29.0003	0.094794
30.1577	0.089192
31.3151	0.084114
32.4725	0.079493
33.6299	0.075276
34.7873	0.071413
35.9447	0.067866
37.1021	0.064599
38.2595	0.061583
39.4169	0.058791
40.5743	0.056202
41.7317	0.053794
42.8891	0.051551
44.0465	0.049458
45.2039	0.047501
46.3613	0.045668

Find the following derivatives

y'(20.8984)
y''(27.8426)
y'''(46.3613)

Use finite difference formulas. How many significant figures can the answers have at most? Of the calculated derivatives, sort them based on precision. (Consider order of derivation and order of differences) Which are more accurate?

If you were to go into business with them selling this potion, what should be the most appropriate methodology to estimate the derivatives and thus make sure that the potion was manufactured properly?

IV. Conclusions

With the creation of this software it is simpler, more suitable and faster, to have more than enough problems for exercises, class work and exams for groups of numerical methods in engineering. All this with the elimination of the inconvenience of producing the problems manually.

Additionally, it is also possible to achieve batteries of problems, to feed the exam reagents in an LMS, in such a way that the same exam is not applied to more than one student.

With problem solving, feedback can be given to students' doubts.

It completely eliminates students sharing solutions to problems from one group to another.

You also avoid a potential copyright issue by using third-party material issues.

Of the problems generated, most correspond to the function a1*curt(x)+a0

For future work, the following are contemplated:

- Change the code from Korn shell, to pass it to C.
- Pass the FORTRAN code, to pass it to C.
- Increase the type of derived equations.
- Fine-tune the runtime.
- Employ with a database to handle the problems generated.
- Add the functionality to locate stored problems.
- Add with export to formats such as PDF and spreadsheet.
- Generate problems applied to real topics of numerical derivation.
- Generate problems such that the solution can be given a physical interpretation, in such a way that it can be used in other gamification contexts.
- Generate problems from other subjects such as probability and statistics, optimization, etc.

Acknowledgment

We are grateful for the facilities provided to the Universidad Autónoma Metropolitana Unidad Azcapotzalco, for the realization and implementation of this software.

References

[1] I. Designers, H. Tinmaz, and I. Technologies, "The Perception on Fundamentals of Online Courses: A Case on Prospective Instructional Designers," *Eur. J. Contemp. Educ.*, vol. 15, no. 1, pp. 163–172, 2016, doi: 10.13187/ejced.2016.15.163.

[2] J. Simões, R. D. Redondo, and A. F. Vilas, "A social gamification framework for a K-6 learning platform," *Comput. Human Behav.*, vol. 29, no. 2, pp. 345–353, 2013, doi: 10.1016/j.chb.2012.06.007.

[3] R. L. Burden, D. J. Faires, and A. M. Burden, *Análisis Numerico*, 10a ed. México, D.F.: Cengage Learning Editores, S.A. de C.V, 2017.

[4] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, "From game design elements to gamefulness: Defining 'gamification,'" *Proc. 15th Int. Acad. MindTrek Conf. Envisioning Futur. Media Environ. MindTrek 2011*, no. March 2014, pp. 9–15, 2011, doi: 10.1145/2181037.2181040.

[5] J. H. Aristizábal, H. Colorado, and H. Gutiérrez, "El

juego como una estrategia didáctica para desarrollar el pensamiento numérico en las cuatro operaciones básicas," *Sophia*, vol. 12, no. 1, pp. 117–125, 2016, doi: www.scielo.org.co/pdf/sph/v12n1/v12n1a08.pdf.
[6] F. F.-H. Nah, V. R. Telaprolu, S. Rallapalli, and P. R. Venkata, "Gamification of Education Using Computer Games," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8018, no. PART 3, pp. 99–107, 2013, doi: 10.1007/978-3-642-39226-9.
[7] A. D. Handayani, T. Herman, and S. Fatimah, "Developing Teaching Material Software Assisted for Numerical Methods," *J. Phys. Conf. Ser.*, vol. 895, no. 1, 2017, doi: 10.1088/1742-6596/895/1/012019.
[8] M. Caligaris, G. Rodríguez, A. Favieri, and L. Laugero, "Developing Mathematical Abilities Using Learning Objects When Learning Numerical Methods," *ICERI2017 Proc.*, vol. 1, no. November, pp. 6018–6026, 2017, doi: 10.21125/iceri.2017.1569.
[9] A. Becerra-Romero, M. Díaz-Rodríguez, and O. A. González-Estrada, "Development of a virtual learning environment for the subject numerical methods under Moodle," *J. Phys. Conf. Ser.*, vol. 1161, no. 1, 2019, doi: 10.1088/1742-6596/1161/1/012010.
[10] M. Pramita, R. A. Sukmawati, and D. P. Sari, "The Implementation of Flipped Classroom Assisted by Learning Management System for Numerical Method Courses," vol. 274, pp. 158–162, 2018, doi: 10.2991/iccite-18.2018.36.
[11] R. M. Clark and A. Kaw, "Adaptive learning in a numerical methods course for engineers: Evaluation in blended and flipped classrooms," *Comput. Appl. Eng. Educ.*, vol. 28, no. 1, pp. 62–79, 2020, doi: 10.1002/cae.22175.
[12] G. Alastrué and J. Antonio, "Problemas numéricos con corrección automática: qué se puede y qué se debe hacer," in *V Jornadas para la Enseñanza y Aprendizaje de la Estadística y la Investigación Operativa*, 2014, pp. 35–52.
[13] J. C. Domínguez *et al.*, "Teaching chemical engineering using Jupyter notebook: Problem generators and lecturing tools," *Educ. Chem. Eng.*, vol. 37, no. June, pp. 1–10, 2021, doi: 10.1016/j.ece.2021.06.004.
[14] M. B. Cutlip *et al.*, "A collection of 10 numerical problems in chemical engineering solved by various mathematical software packages," *Comput. Appl. Eng. Educ.*, vol. 6, pp. 169–180, 1998, doi: 10.1002/(SICI)1099-0542(1998)6:3<169::AID-CAE6>3.0.CO;2-B.
[15] N. Solution, U. Parametrization, and O. Software, "Numerical Solution of Dynamic Optimization Problems Using Parametrization and OptiA Software," vol. 121, no. 1996, pp. 111–121.
[16] VaxaSoftware, "Generador de ejercicios de Matemáticas." 2020.
[17] G. Avdelas, A. Konguetsof, and T. E. Simos, "A generator of hybrid explicit methods for the numerical solution of the Schrödinger equation and related problems," *Comput. Phys. Commun.*, vol. 136, no. 1–2, pp. 14–28, 2001, doi: 10.1016/S0010-4655(00)00249-6.
[18] J. F. Hake, "The numerical solution of PDE problems with standard software packages," *J. Comput. Appl. Math.*, vol. 20, no. C, pp. 257–265, 1987, doi: 10.1016/0377-0427(87)90142-7.
[19] A. Berg and M. Korcuska, *Sakai Courseware Management The Offcial Guide*, First. Birmingham, UK: © 2009 Packt Publishing, 209AD.
[20] M. Ignjatovic and S. Jovanovic, "Implementing Sakai Open Academy Environment Pros and Cons.," *Int. J. Emerg. Technol. …*, no. April, pp. 64–68, 2013.
[21] D. Martínez Roldán, R. Mengod López, E., and D. Merino Echeverría, *SAKAI Administración, configuración y desarrollo de aplicaciones*, Primera ed. México, DF: Alfaomega Grupo Editor, 2011.
[22] "Sakai History | Sakai," 2014. [Online]. Available: https://sakaiproject.org/sakai-history. [Accessed: 11-Aug-2015].
[23] L. M. R.- Moreno, "Análisis Comparativo entre las Plataforma de más Frecuente Implantación en los Sistemas Virtuales de Formación frente a un Modelo : Proyecto Sakai (Comparative Analysis between the most Common Platform in the Virtual Training Systems against a Model : Sa."