Mathematics Exercise Generator:

the language of parameterized exercises

Paula Oliveira
University of Aveiro, Portugal

Paula Carvalho
University of Aveiro, Portugal

ABSTRACT

The process of teaching and learning is changing from a traditional model in which teachers were the source of information for a model in which teachers appear as advisors who carefully observe students, assist in the selection of information by identifying their learning needs and support students in their autonomous study.

This chapter describes an approach used in curricular units of first year in Science and Engineer degrees, which results from a connection of three projects born in University of Aveiro: MEGUA, SIACUA and PmatE, and the interconnections of their informatics platforms. Although any scientific area besides mathematics can use this tool, the authors focus in a case study using an example on a specific topic of Calculus courses for first year students on Engineering: Sequences and Series of Functions. The methodology described allows teachers to achieve further goals on learning strategies and students to have enough material to practice.

Keywords: Computer-based Assessment, Question Generation, Automated Feedback, Siacua, Pmate, Mathematics, Teaching, Learning

INTRODUCTION

In recent years, the process of teaching and learning has undergone great transformations in its essence. It has been a gradual change from a traditional model in which teachers were the source of information and knowledge to a model in which teachers appear as advisors, or tutors, who carefully observe students, assist in the selection of information by identifying their learning needs and support students in their autonomous study. Every academic year instructors face the problem of producing several materials for their students. In particular, they seek and produce new problems and new exercises, although what they really do are slight variations of previous exercises as changing the values in an equation, or a parameter in a function. This is a tedious and time-consuming process, and this time is precious for other professional requests as new methodologies to foster learning. It is taken for granted that students want on-line exercises and an immediate feedback on their performance. Furthermore, the paradigm

studying mathematics = solving hundreds of exercises

is a problem teachers have to deal with frequently, although they know that there are students who need to practice a lot until they understand the concepts underlying the exercises, whereas some others only need to solve a few exercises to catch all those concepts. Creating new exercises and finding the correspondent solutions is a slow-paced process. To overcome this feature, it appeared in University of Aveiro in the early 90's, a computer platform, named *PmatE*, that developed the concept of question generator model, QGM (Oliveira, Carvalho, & Vieira, 2004). Since then it has been created a large set of QGMs producing a huge number of exercises on Calculus of one, or several variables, that are being used for self-assessment and evaluation in Calculus courses, enrolling hundreds of students each academic year.

Although random generation data begins to be a tool used in several areas, automated exercise generation and grading is a much more immature problem. Automatic exercise grading is a problem that has been mostly solved, but automated problem generation is a much more open problem (DuFrene, 2016).

Regarding Mathematics, an exercise generator more than simply producing many exercises, is a methodology that works for a general class of proof problems, which involves establishing the validity of a given set of algebraic identities. An exercise generator might be considered as a class of exercises, on one curricular topic, with the same didactic purpose and similar complexity. Each of those classes is characterized by a set of variables assuming random values in their domains, usually subject to some constraints. Whenever the random variables are instantiated, with random values taken from the range defined by the exercise´s author, these values spread all over the scope of the generated exercise. In this way, the whole class of exercises is authored at once by authoring one highly annotated randomized exercise.

This work presents a design of an automated generation of template-based questions, ensuring that most students get different questions of similar complexity, thus benefiting the (self-) assessment process and allowing each student to learn at his own pace. The authors describe an approach used in curricular units of first year in Science and Engineer degrees, which results from a connection of three projects born in University of Aveiro: *MEGUA*, *SIACUA* and the already cited *PmatE*, and the interconnections of their informatics platforms. To be more precise, the main goal of MEGUA project (Mathematics Exercise Generator, University of Aveiro) (Cruz, Oliveira, & Seabra, 2013) is to create and share parameterized content among several authors. On the other hand, the SIACUA project (Siacua: Interactive Computer Learning System, University of Aveiro, 2017) aims to create computer systems with interaction and feedback to support the autonomous study of students and, in addition with other resources, makes use of a large amount of content created under the MEGUA project. Finally, the *PmatE* project is an evaluation platform, that uses contents created both in the MEGUA platform and in PmatE itself (Projecto Matemática Ensino da Universidade de Aveiro – PmatE/UA, 2019).

The chapter is organized as follows: after a brief background, the formal description of a Mathematics Exercise Generator is introduced, followed by an example withdrawn from a Calculus course for Science and Engineering degrees. Then there are some instantiations of this example and the discussion of some results regarding the curricular unit where it was used.

Mathematics Exercise Generators (MEGs) are language independent: MEGUA uses *Python* and *LaTeX* to produce these exercises and *PmatE* uses ModelMaker, an application developed in *ASP.NET* with *SQL* databases and *Javascript* frameworks (Camejo, Silva, Descalço, & P., 2016).
The motivation behind the creation of parametric exercises is based on a new learning approach - active learning. The decrease of contact hours in classes forces students to work on their own, and the need of getting new exercises with detailed solutions, even for motivated students, as well as other resources to foster learning, is a demand for teachers and instructors. However, either generating fresh problems, that can be used for studying or for exams every academic year, is a tedious task for the instructors, particularly when they have to create several exercises involving the same concept and the same difficulty level. In spite of creating a MEG is also time-consuming, the benefits of generating some thousands of similar exercises at once is a long-term benefit, thus freeing up time to invest on implementing new learning scenarios or paths.

Although any scientific area besides mathematics can use this tool, the authors focus in a concrete example on a specific topic of Calculus courses for first year Engineering students: Sequences and Series of Functions.

The methodology described allows teachers to achieve further goals on learning strategies, either creating more material for students or investigating about other teaching strategies that could improve learning, and students to have enough material to practice.

BACKGROUND

In the literature one can find several definitions of parameterized exercises. In (Rioja, Gutierrez-Santos, Pardo, & Delgado-Kloos, 2003) *parametric exercises are simple extensions of conventional exercises that allow multiple replications with different data. The variation captured by the parameters not only refers to specific data inside the exercise, but also to the format in which content is presented to the student.*

In (Goguadze & Tsigler, 2017) it is defined as *exercises in which the task and solution space are randomized.* Moreover, (Goguadze & Tsigler, 2017) states *When starting such an exercise, each time the variables to be randomized are instantiated with random values taken from the range defined by the author and these values are propagated throughout the whole solution space of the exercise. This way, the whole class of exercises is authored at once by authoring one highly annotated randomized exercise.*

In (Oliveira, Carvalho, & Vieira, 2004), a parameterized exercise is a question generator on a specific curricular subject, satisfying a classification on scientific and didactic goals as well as a measure of difficulty (on a scale from 1 to 5). The questions are randomly generated by parameterized expressions, where the parameters assume values on a set named parameters' domain. In this sense, this definition does not differ from the ones referred above.

A parameterized exercise might be considered as a class of exercises, on a curricular topic, with the same didactic purpose and similar complexity. That class is characterized by a set of variables assuming random values in their domains, usually subject to some constraints.

The learning objects MEG herein described can be used in multiple contexts:

- **Online**
  - in autonomous study
  - in self-assessment
  - in computer-aided (summative) assessment
- **Paper format**
  - In written exams
  - In course textbooks or worksheets to support students work

The number of MEGs already existing embraces a wide range of topics on mathematics, thus requiring a classification. In SIACUA (see Figure 1) and in PmatE (see Figure 2) each course is organised as a tree: the course is the trunk and the branches are the topics and/or concepts which are part of the course syllabus. The leaves are the exercises on the specific topic/concept.

Tools like SIACUA and PmatE platforms can be used to support various types of learning strategies such as Flipped Learning. The instructional material produced by parametrized exercises, allows students with different profiles and backgrounds to learn at their own pace, and where and whenever it suits them (Descalço L. C., 2018).

In (Descalço L. C., 2015) is presented an experience in Multivariate Calculus, supported by SIACUA. Also, an example of the use of exercises generators is the case of National Science Competitions boosted by PmatE and described in (Descalço & Oliveira, Science Competitions: do they foster Learning?, 2018).

Figure 1. The concepts tree for SIACUA

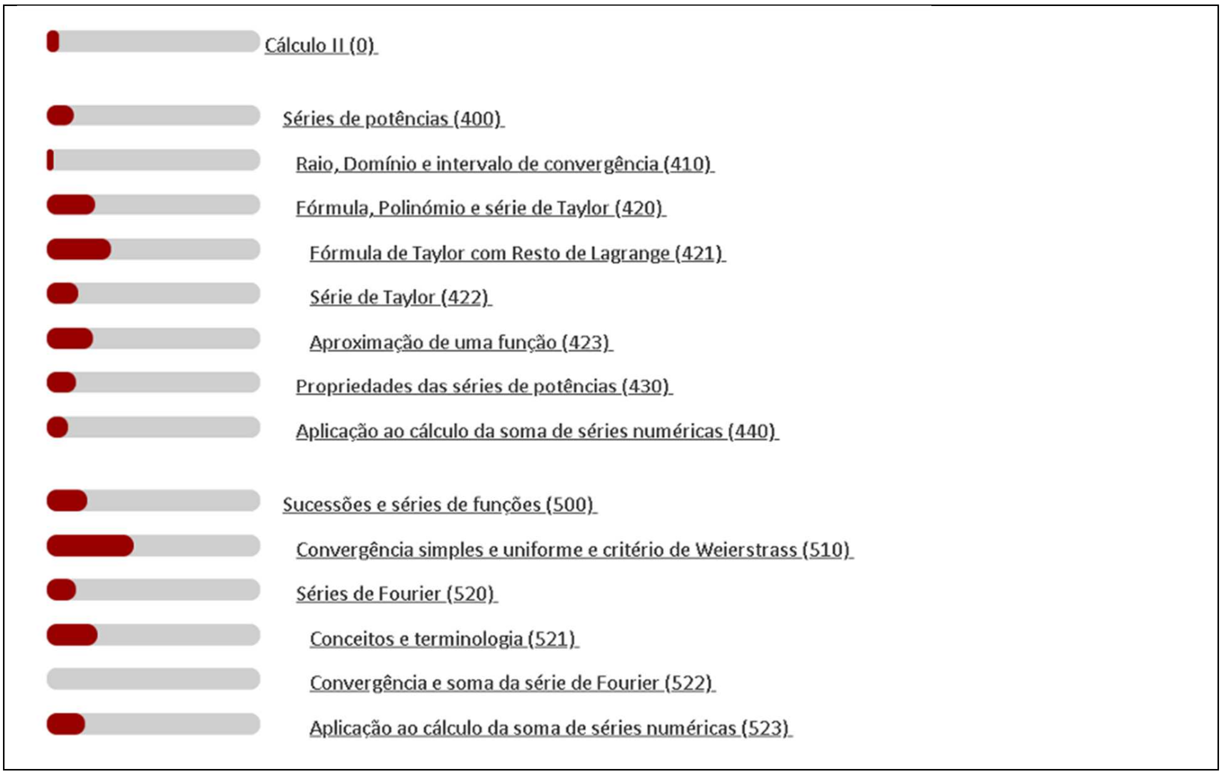


Figure 2. Concepts tree in PmatE

MATHEMATICS EXERCISE GENERATOR (MEG)

The exercises generator herein described is in the context of mathematics, nevertheless, it can be used to parametrize questions in any scientific (or even nonscientific) area, and the process of parameterizing can be simplified without the mathematical formalism characteristic of this area.

In general, a common mathematics exercise has a statement followed by a question, for example:

> *Let f be a real function defined on* $\mathbb{R}$ *by the expression f(x)=cos(2x).*
> *Find the zeros of the function f in the interval* $[-5\,\pi, 5\,\pi]$.

An exercise generator intends to randomize, within some limits, all the variable fields that appear on it. The example above could be modelled as follows:

> *Let f be a real function defined on* `domain` *by the expression f(x)=* `expression.`
> *Find the zeros of the function f in the interval* [`initial value`, `final value`].

The expressions `domain`, `expression`, `initial value` and `final value`, are randomly chosen from sets given by the MEG's author in the context of the exercise.
The construction of such generators is a demanding work that involves scientific and didactic abilities; as referred in (Sadigh, Seshia, & Gupta, 2012), *it is unrealistic and also undesirable to completely remove the instructor from the problem generation process, since this is a creative process that requires the instructor's input to emphasize the right concepts*.

In the following paragraphs it will be described the structure of a Mathematics Exercise Generator. Each MEG has its own purpose and formulation. In general, a MEG is a 4-tuple (`IT;CI;MCQ;DA`) or (`IT,CI,Q,DA`) where `IT` is an introductory text, `CI` is the complementary information, these two fields combined form the text question, `MCQ` is the multiple choice field and `Q` is the text of an essay (or open answer) question and `DA` is the detailed answer to the proposed essay question or multiple choice question. Each of these 4 fields has its specific domain: the set of parameters where they vary. For instance, the domain for introductory text is

$$D_{IT}=\{IT_i : i=1,\ldots,k_{IT}\}$$

The complementary information field may consist on sentences, expressions using variables (such as the expression defining a function), images, graphics or just numerical parameters.
The text of the exercise will be the pair

$$(IT_i;CI_j) \in D_{IT}\times D_{CI}$$

If there are no constraints, the number of concrete pairs $(IT_i;CI_j)$ is the product of the cardinality of D_{IT} by the cardinality of D_{CI}. However, quite often these combinations are subject to some constraints, which are imposed on the choice of the indices i and j (cf. with the example in the next section).

A MEG's randomness is improved if the questions on the same concept are differently formulated, thus avoiding memorizing answers or procedures. This purpose is accomplished introducing in the domain of `Q`, D_Q, different formulations for the same question or different questions on the same subject.
In the case of a Multiple Choice Question, the correct item is chosen from a set of true statements and the wrong items, the distractors, are collected from the most common errors observed by experimented teachers as the misapplication of a theorem or formula, the correct identification of a concept or mistakes in calculation. Analyzing students' responses, teachers can identify the reason of failure and provide an appropriate feedback. An element of `MCQ` is a n-tuple with $n \geq 4$ of the form

$$(\mathtt{correct}, \mathtt{wrong}_1, \mathtt{wrong}_2, \ldots, \mathtt{wrong}_{n-1})$$

and the domain for this MCQ variable is a pair of sets one the domain for the correct item and the other the domain for the wrong items: $D_{correct} \times D_{wrong}$.

Creating a multiple-choice question is a thorough task in this context. Each choice has associated a specific goal depending on the knowledge one wants to observe. The construction of the choice items follows the misapplication of a theorem or formula, the correct identification of a concept, or the most common errors observed by experimented teachers. Therefore, when a student chooses a wrong sentence it is possible, in general, to identify the reason of failure and give an appropriate feedback. In this context there are new variables, `correct choice` and `wrong choice`. Sometimes there are different ways of formulating a correct answer, enriching the understanding of a concept. Often instead of the necessary minimum of three wrong choices there are more that are also randomly chosen to appear in each concrete question. Each one of those variables has its specific domain, as in the other fields of the generator.
The most delicate issue of a MEG is the detailed answer. The author has to be very careful when the intention is giving an answer of the exercise, including the theoretical results used to solve it. In the technical point of view, the DA is conditioned by the 3-tuple $(IT_i; CI_j; Q_k)$ or $(IT_i; CI_j; MCQ_k)$ and in the Multiple Choice case it is often a non-trivial combinatorial problem, because the detailed answer must be consistent with the correct statement chosen for the correct item.
The generator just described has been used in Mathematics, but also in Natural Sciences as Biology and Geology, Physics, Chemistry, Portuguese Language and English Language, with the necessary adjustments within PmatE's scope.

AN EXAMPLE OF A MEG

This section illustrates MEG objects with an example withdrawn from a Calculus course for Science and Engineering degrees.
Among the subjects which Calculus students find particularly difficult is *uniform and pointwise convergence of sequences and series of functions*. Although the topic isn't thoroughly studied, it is very important for applications in *Power and Fourier Series*, that is why it is part of the course syllabus. The example here presented was used in the spring semester of 2018/19 in online assessment and in a written exam and the results of its use will be discussed in this chapter.

The introductory text *IT* is randomly generated from the set with two elements, D_{IT}:

$$D_{IT} = \left\{ \text{The sequence of functions } \left(f_n(x)\right);\ \text{The series of functions } \sum_{n=1}^{+\infty} f_n(x) \right\}$$

and the complementary information is

$$\text{where } f_n(x) = \text{function}_i \text{ for } x \in \text{domain}_j$$

Here function_i, is randomly chosen from the set of 13 elements listed on Table 1.

Table 1. Domain of the variable `function`

i	Function	i	Function	i	Function
0	$\frac{a_1\, n\, x}{a_2 n\, x + b_1}$	1	$\frac{a_1\, n\, x}{a_2 n\, x + b_1\, n}$	2	$\left(a_1 x - \frac{b_1}{n}\right)^2$
3	$\frac{a_1\, x^n}{b_1 n + a_1 x^n}$	4	$\exp\left(-\frac{a_2 x^2}{n}\right)$	5	$\frac{trig(x)}{b_1 x}$
6	$\frac{b_1}{a_2 n\, x + a_1}$	7	$\frac{a_2 x}{b_1\, n\, x + a_1}$	8	$\frac{b_1 x}{a_2 + a_1 n x^2}$

9	$\frac{a_2 trig(b_1 x)}{a_1 n^2}$	10	$\begin{cases} \frac{a_1}{n^2} \ if \ \lvert x\rvert \le n \\ \frac{b_1}{x^2} \ if \ \lvert x\rvert > n \end{cases}$	11	$\frac{a_2}{a_1 x^2 + b_1 n^2}$
12	$\frac{trig(x)}{a_2 n^2}$				

This list of 13 functions is complemented with the numerical parameters $a_1, a_2, b_1 \in \{1,2,\dots,8\}$ and in cases 5, 9 and 12, `trig` stands either for sine or cosine function.

Moreover, each `function`$_i$ is defined on a specific set, its (mathematical) domain which can be one of the 4 elements in the following set:

$$D_{domain} = \{\mathbb{R};\ \mathbb{R}_0^+;\ \ ]0,1[\,;\ [0,1]\}.$$

The formulation `(IT`$_i$`;CI`$_j$`)` $\in D_{IT} \times D_{CI}$ is now complete:

> `IT`$_k$ where $f_n(x)$ =`function`$_i$ for $x \in$ `domain`$_j$.

For example, if $k = 2$, $i = 9$, $j = 1$, `trig`$=\ cos$, $a_1 = 2$, $a_2 = 1$ and $b_1 = 6$, the first part of the generated exercise is:

> Consider the series of functions $\sum_{n=1}^{+\infty} f_n(x)$ where $f_n(x) = \frac{\cos(6x)}{2n^2}$ for $x \in \mathbb{R}$.

In the case of an open answer question, the field `Q` consists on the following questions:

> `Q`$_0$: Show that it converges uniformly on D=`domain`$_j$.
>
> `Q`$_1$: Find its pointwise limit.
>
> `Q`$_2$: Show that the sequence of functions converges to `limitfunction` on D=`domain`$_j$.
>
> `Q`$_3$: Show that it converges pointwise on D=`domain`$_j$. Justify that the convergence is not uniform.
>
> `Q`$_4$: Justify that the sum of the series is a continuous function on D=`domain`$_j$.

In Q2, the `limitfunction` is a variable depending on `function`$_i$.

Some constraints are imposed due to the profile of the students. For instance, since determining the sum of a series of functions is not easy, when the choice is `IT`$_1$, the options `Q`$_1$ and `Q`$_2$ are omitted.

Regarding the Multiple Choice Question format the options for the correct item are the same as for the open answer format, with the necessary adjustments in the sentences. The wrong items include sentences in the negative form of the correct one, wrong domains of convergence or wrong expressions for the `limitfunction`, the use of the words *uniform convergence* when the correct form is *pointwise convergence* (or the opposite) and are illustrated in several examples in this chapter.

An instantiation of this MEG in the Multiple Choice format is

Consider the series of functions $\sum_{n=1}^{+\infty} f_n(x)$ where $f_n(x) = \frac{\cos(6x)}{2n^2}$ for $x \in \mathbb{R}$.

correct: The series is *uniformly convergent on* $\mathbb{R}$.

wrong1: Weierstrass's criterion can't be applied.

wrong2: The series *is uniformly convergent* if and only if $|x| \leq 1$.

wrong3: The series *converges pointwise but not uniformly* on $\mathbb{R}$.

The main goal of this question is to find out whether students know that the Weierstrass criterion is applicable and what is the conclusion that results from it.
To clarify, the Weierstrass criterion (also known as Weierstrass M-test) can be stated as follows.

Let the functions (f_n) *defined on* $D \subseteq \mathbb{R}$ *satisfy the inequality* $|f_n(x)| \leq a_n$, *for all* $x \in D$ *and for each* $n \in \mathbb{N}$, *where* $\sum_{n=1}^{+\infty} a_n$ *is a convergent series of non-negative real terms. Then the function series* $\sum_{n=1}^{+\infty} f_n(x)$ *converges uniformly in* D.

In this exercise the Weierstrass criterion allows the conclusion about uniform convergence, however, there are uniformly convergent series for which it is not applicable, as for instance the series $\sum_{n=1}^{+\infty} \frac{\phi(x-n)}{n}$ where $\phi(x) = max\,\{0, min\{2x, 2-2x\}\}$ (see (Gantumur, 2015)).

The distractor in wrong2 is the domain of convergence. The interval was used by two main reasons: (1) the fact that $|\cos(6x)| \leq 1$, not $|x| \leq 1$; (2) it is common in some exercises that do not involve *sine* or *cossine* functions, to use the interval $[-1,1]$ to easily prove the uniform convergence. wrong3 tests the correct classification of the convergence: pointwise or uniform.

The detailed answer has to be consistent with the parameters instantiated. For this particular exercise, the answer is as follows:

As forall $x \in \mathbb{R}$, $|\cos(6x)| \leq 1$, we may apply Weierstrass's criterion, by considering the sequence $a_n = \frac{1}{2n^2}$, and

$$\left|\frac{\cos(6x)}{2n^2}\right| \leq \frac{1}{2n^2}.$$

The series $\sum_{n=1}^{+\infty} \frac{1}{2n^2}$ converges (as studied in numerical series chapter) so, by Weierstrass's criterion, we may conclude that the series

$$\sum_{n=1}^{+\infty} \frac{\cos(6x)}{2n^2}$$

converges uniformly on $\mathbb{R}$.

USING A PARAMETERIZED EXERCISE IN A CALCULUS COURSE

In the academic year 2018/19, in a Calculus course for Engineering degrees, MEGs were used in different contexts:

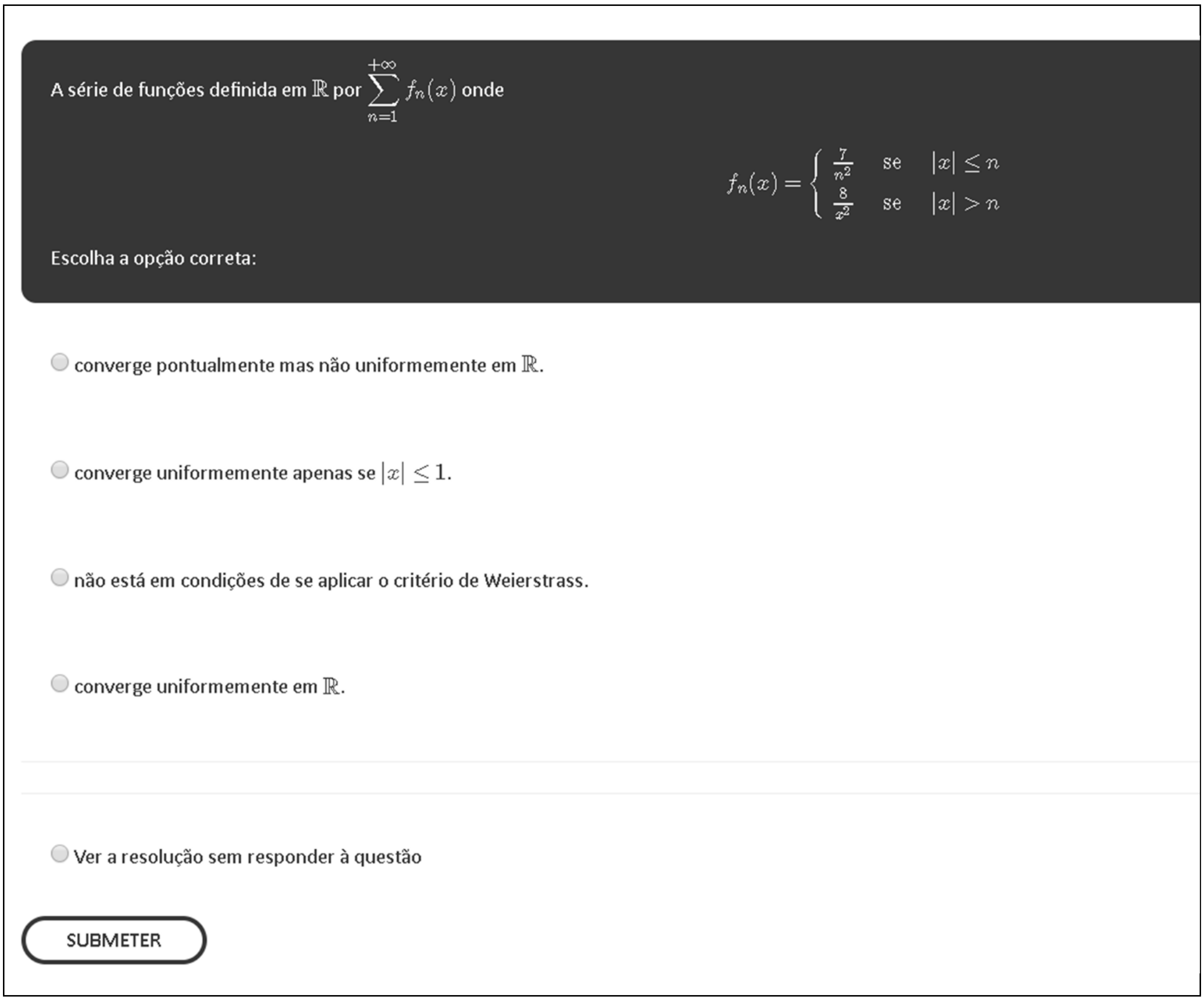
- for self-assessment (in SIACUA platform),
- for evaluation (online in PmatE platform and in written tests or exams) and
- to study (as examples in textbook supporting the curricular unit).

The first chapter on this course is *Function Sequences and Function Series*, with a special emphasis in *Pointwise Convergence and Uniform Convergence*, which is a difficult topic for first year students.

The Mathematics Exercise Generator described in the previous section was conceived on this topic and was used in the cases listed above, as illustrated on the following figures. All the examples are in Portuguese as the course is lectured in this language. The translation of these exercises is included in Appendix 2.

In Figure 3 is an example used in SIACUA platform where the topic *Sequences and series of functions – Pointwise and uniform convergence and Weierstrass's criterion* has the concept id. 510 (see Figure 1).

Figure 3. An instantiation of the exercise in SIACUA platform



In SIACUA platform the students can see the detailed answer to the exercise or choose, from the four choices available, the correct one. In case of failure, the detailed answer is displayed, otherwise the system returns to the main page of the course and the knowledge on the specific subject is updated. (Siacua: Interactive Computer Learning System, University of Aveiro, 2017).

Figure 4. The detailed answer to the exercise in Figure 3

Repare-se que, se $|x| \leq n$, $|f_n(x)| \leq \frac{7}{n^2}$ e se $|x| > n$, tem-se $|f_n(x)| \leq \frac{8}{n^2}$.

Tomando o máximo entre $\frac{7}{n^2}$ e $\frac{8}{n^2}$, podemos afirmar que

$$|f_n(x)| \leq \frac{8}{n^2}$$

Como a série $\sum_{n=1}^{+\infty} \frac{8}{n^2}$ é convergente, podemos afirmar, pelo critério de Weierstrass, que a série $\sum_{n=1}^{+\infty} f_n(x)$ converge uniformemente.

In the online test on PmatE, in the beginning of the semester, this exercise was one of its five questions, each of them an instantiation of one particular MEG, thus ensuring that each student had a different test. In Figure 5 is an example of a question instantiated from the MEG described.

Figure 5. Another instantiation of the exercise in PmatE's online test. In this one the function randomly chosen was function3 from Table 1

MT1_Calculo2 | 33:43 | questões: 5

5

A sucessão de funções definida em $\mathbb{R}_0^+$ por

$$f_n(x) = \frac{4\,x^n}{5\,n + 4\,x^n}$$

Escolha a opção correta:

converge uniformemente para a função

$$f(x) = \begin{cases} 0 & \text{se} \quad x \in [0,1] \\ 1 & \text{se} \quad x > 1 \end{cases}.$$

tem como limite pontual a função

$$f(x) = \begin{cases} 0 & \text{se} \quad x \in [0,1] \\ 1 & \text{se} \quad x > 1 \end{cases}$$

tem limite

$$\lim_{n \to +\infty} f_n(x) = 1, \ \forall x \geq 0.$$

tem limite

$$\lim_{n \to +\infty} f_n(x) = 0, \ \forall x \geq 0.$$

The exercise was also used in the textbook supporting the curricular unit in 2018/19, as illustrated in Figure 6.

Figure 6. An exercise in the textbook supporting the curricular unit

Exercício 1.5.13 Considere a série de funções $\sum_{n=1}^{+\infty} \frac{nx}{3nx+1}$ definida em $\mathbb{R}_0^+$. Justifique que esta série não converge uniformemente.

Another formulation of this MEG (cf. Figure 7) was used in the first test of the semester, in order to support the results of this study.

Figure 7. The exercise included in the first written test of Calculus, in the academic year 2018/2019

1. Considere a série de funções $\sum_{n=1}^{+\infty} \frac{3}{5\,n^2 + 8\,x^2}$.

   (a) Determine o domínio de convergência da série e justifique que nesse domínio a série é uniformemente convergente.

   (b) Justifique que a funçao S, soma da série, é contínua no domínio de convergência.

These four examples suggest the multiplier effect of constructing parameterized exercises, in spite of the difficulty in conceiving them.

RESULTS

Calculus is a second semester curricular unit for first year Engineering degrees. The main evaluation method consists in two written exams, the first one during the semester and the second after classes are over. However, students have the option to complement their assessment taking online tests in PmatE platform along the semester.

The first written exam took place in April 2019 and included a question (Question 1) on Sequences and Series of Functions (Figure 7) and the first online test on PmatE platform (in the end of February 2019) also included a question on this subject, as illustrated in Figure 5.

From the students who took the written exam and the first online test it was chosen a sample of 56 that accessed SIACUA bar on the specific concept – *Pointwise and uniform convergence. Weierstrass's criterion* (concept id. 510). All the grades obtained in the written exam and in Question 1 as well as the marks in the online test are gathered in Table 3 in Appendix 1. This table also includes a column with the number of accesses to SIACUA bar concept id. 510.

Table 2 establishes the relation between the grades in Question 1 (Q1), from 0 to 25 points, grouped in three classes, and the number of accesses to SIACUA bar concept id. 510 (A).

Table 2. Relation between the marks obtained in Q1 and the number of accesses to SIACUA, in percentage

	Q1<5	**5<=Q1<15**	**15<=Q1<=25**
A=1	12,5%	1,8%	5,4%
2<=A<5	5,4%	7,1%	10,7%
A>=5	14,3%	12,5%	30,4%

The classes used above translate the authors' opinion on the use of SIACUA. Students accessing only once, 19,6% of the total, are just curious, and didn't use the platform for studying. The second class, 2<=A<5, might suggest different readings, for example:

- students understood the concept after this number of attempts (which could be the case of the 10,7% of the students who had a good mark in Question 1)
- or found it too difficult and gave up studying the subject (which could be the case of the 5,4% of the students who had a bad grade in Question 1).

A number of accesses greater or equal to 5 suggests the use of the platform to study the subject, consisting of 57,1% of the population in this study.
A total amount of 70% of these 56 students accessed SIACUA more than once and 46,4% had grades in Question 1 greater or equal to 15 points.

The correlation coefficient between the grades in the written exam and the marks obtained in Question 1 is 0,75, thus showing, in spite of the subject being a non-trivial one, that the question is adjusted to the other questions in the test.

DISCUSSION

In previous years, the questions related with *Function Series and Uniform Convergence* were difficult for students, and only a minority answered them correctly. In Question 1 of the written exam in 2018/19, 46,4% of the students attained a grade greater or equal to 15 points, in a maximum of 25. From these 65,4% had a number of accesses to SIACUA bar concept id. 510 greater or equal to 5.
In addition, 15 average students, who got grades between 8 and 15 in the written test, and accessed SIACUA bar more than once, obtained a Q1≥15.
The results can be compared with the homologous of the academic year 2017/18. The first written exam was on the same subjects and only 44% of the students attained grades above 49% while in 2018/19 this number raised to 65%. The authors believe that using a frequent online assessment as well as having several exercises in SIACUA with detailed answers help students on improving their knowledge, as well as giving correct answers to questions they usually find difficult.

Creating a large number of different exercises on the same topic, that can be used in different contexts, year over year, is facilitated by the use of question generators.
There is the risk of memorizing answers without understanding the concepts, however the teacher can free time to create more generators, instead of spending it on looking for different exercises.

Some inquiries made in recent years regarding the students' opinion on the benefits of having SIACUA platform and exercises with detailed answers, they were unanimous in stating that it is very useful, for instance in (Descalço, Carvalho, & Oliveira, Motivating study before classes on Flipped Learning, 2018).

CONCLUSION

The MEGs described in this text are programmed using Python language, LaTeX and HTML, although they could be produced using other tools, as is the case on PmatE described in (Camejo, Silva, Descalço, & P., 2016).

The technical details aren't discussed here as the important issue is the algorithm to create exercise generators. Python, or another programming language, is just a matter of taste or simplicity.
It is important to make use of some CAS libraries (eg. Sagemath), as one can differentiate, integrate, etc, operations often needed either for the answer or for the multiple choice items.

The exercises generated are being used in worksheets, textbooks, online platforms for (self-)assessment as well as in exams. Although there exists already a database of hundreds of generators on Calculus topics (from one variable Calculus to functions of several real variables), the aim of the authors is to completely cover all the topics on Calculus and develop an intelligent self-assessment tool that guides students through a net of concepts.

The exercises generated are already part of a tutoring system SIACUA, described in (Siacua: Interactive Computer Learning System, University of Aveiro, 2017), where the generic Bayesian student model was implemented following the ideas presented in (Millán, Castillo, Descalço, Oliveira, & Ramos, 2013). However, this tutoring system can be improved, suggesting new exercises for each student, depending on his previous performance. Analyzing all the log data of students, the authors of the exercises can identify some common conceptual and procedural errors of learners, thus, enhancing either the system or the problems instantiated.

In (Skopljanac-Macina, Blaskovic, & Pintar, 2016), is introduced a web application that meets the fundamental prerequisites:

- ability to generate simpler and more complex question variants from a basic question template
- automatic generation of random figures
- automatic solving of the given problem
- automatic generation of step-by-step solution to the given problem as an additional feedback information for the students

in order to answer the question

> *is there an effective way to easily generate dynamic questions that would benefit the assessment process, besides ensuring that the most students get different questions of similar complexity?*

The motivation behind the creation of parametric exercises is common to all instructors authoring them:

> *Teachers give homeworks and exams, and ensure that students get enough practice. However, generating fresh problems that involve the same set of concepts and have the same difficulty level as the problems discussed in the class, is a tedious task for the instructor. Even motivated students want to have access to such fresh similar problems, when they fail to solve a given problem and had to look at the solution. (Singh, Gulwani, & Rajamani, 2012)*

Although a great investment is needed to develop exercises or problems generators, the authors believe that students take advantage of it.

> *Nevertheless, teachers and instructors must invest a lot of their time and resources in creating and maintaining teaching and learning online materials, basic learning scenarios and learning paths. Finally, the process of devising test questions and preparing testing materials is very important, but usually delicate and time-consuming. For each new question instructors must also give the correct and the wrong answers, sometimes also draw a figure, and preferably add useful hints. (Skopljanac-Macina, Blaskovic, & Pintar, 2016)*

Apart from the vast list of exercises that can be instantiated, the detailed explanation of why a solution is correct or incorrect helps students to understand the underlying concepts, thus, widening their application to different problems.

In the example of the MEG presented in this text one can have an idea of the number of exercises with similar level of difficulty it generates. This number of questions depends greatly on the functions that can be used, according to Table 1. For the functions in 0, 1, 6, 7, 8, and 11 this number is 439, for functions 2, 5 and 10, this number is 64, in function 4 one can generate 8 possibilities, for functions 5 and 12 this number is 16, one can generate 768 possibilities in the case of function 9 and the function 4 can only vary in 48 ways. The total amount of different functions generated is 3666.
Also, for the variable `domain` there are 4 different possibilities of choice, and for the variable `Introductory Text` there are 2 choices. Therefore, the number of different questions generated rises to 29328. Considering different questions Q or multiple choice questions MCQ this number increases substantially.
As referred in (Gütl, Lankmayr, Weinhofer, & Höfler, 2011), due to automatic question creation, it is possible to support personalized and self-directed learning activities by preparing appropriate and individualized test items quite easily with relatively little effort or even fully automatically.

Finally, the exercise editor has to be continuously extended to support new add-ons to the exercise system providing the authors with features that facilitate exercise authoring, that is, extensions that allow for randomizing over continuous intervals and sets of functions, and so on.

The interest on creating exercises generators can be summarized on the following items:

- It facilitates the process of creating question repositories;
- It may support the personalization of questions generated based on student's profile and personal learning goals;
- It may become a component of Intelligent Tutoring Systems, where already stored domain knowledge and pedagogic strategies may become the basis for automatic problem construction;
- It can become a crucial component in the process of automation of instructional design.

ACKNOWLEDGMENT

This research is partially supported by the Portuguese Foundation for Science and Technology FCT-Fundação para a Ciência e a Tecnologia, through the CIDMA - Center for Research and Development in Mathematics and Applications, within project UID/MAT/04106/2019.

References

Camejo, J., Silva, A., Descalço, L., & P., O. (2016). Modelmaker, a multidisciplinary web application to build question generator models from basic to higher education. *8th International Conference on Education and New Learning Technologies* (pp. 5095--5103). Barcelona: EDULEARN16 Proceedings.

Cruz, P., Oliveira, P., & Seabra, D. (2013). MEGUA package for parametrized exercises. *Proceedings ICERI2013 6th International Conference of Education, Research and Innovation* (pp. 6143-6149). Seville, Spain: Iated Digital Library.

Descalço, L., & Oliveira, P. (2018). Science Competitions: do they foster Learning? *EDULEARN18 Proceedings*, (pp. 1388-1394). Palma, Spain.

Descalço, L., Carvalho, P., & Oliveira, P. (2018). Motivating study before classes on Flipped Learning. *10th International Conference on Education and New Learning*, (pp. 6295-6300). Palma, Spain.

Descalço, L., Carvalho, P., Cruz, J., Oliveira, P., & Seabra, D. (2015). Computer-Assisted Independent Study in Multivariate Calculus. *EDULEARN 15 Conference*, (pp. 3352-3360). Barcelona.

Descalço, L., Carvalho, P., Cruz, J., Oliveira, P., & Seabra, D. (2015). Using Bayesian Networks and Parameterized Questions. *EDULEARN15 Conference*, (pp. 3361-3368). Barcelona.

DuFrene, A. (2016). Automatic Generation and Grading of Programming Exercises. Retrieved Maio 11, 2018, from url: http://digitalcommons.calpoly.edu/cscsp/95

Gantumur, T. (2015). *Power Series*. Retrieved May 11, 2018, from http://www.math.mcgill.ca/gantumur/math249w15/power.pdf

Goguadze, J., & Tsigler, G. (2017). Authoring Interative Exercises in Active Math. *Proceedings of the Mathematical User-Interfaces Worlshop* (pp. 1-10). MATHUI-2007.

Gütl, C., Lankmayr, K., Weinhofer, J., & Höfler, M. (2011). Enhanced Automatic Question Creator - {EAQC} - Concept, Development and Evaluation of an Automatic Test Item Creation Tool to Foster Modern e-Education. *The Electronic Journalof e-Learning*, 23-38.

Millán, E., Castillo, G., Descalço, L., Oliveira, P., & Ramos, S. (2013). Using Bayesian networks to improve knowledge assessment. *Computers and Education*, 436-447.

Oliveira, P., Carvalho, P., & Vieira, J. (2004). Modelo Gerador de Questões. *Proceedings IADIS Conferência Ibero-Americana WWW-Internet*, (pp. 105--113).

Projecto Matemática Ensino da Universidade de Aveiro – PmatE/UA. (2019, April 08). Retrieved from https://pmate.ua.pt

Rioja, R., Gutierrez-Santos, S., Pardo, A., & Delgado-Kloos, C. (2003). A Parametric Exercise Based Tutoring System. *33rd ASEE/IEEE Frontiers in Education Conference. 3 S1B*, pp. 20-26. Colorado: STIPES.

Sadigh, D., Seshia, S. A., & Gupta, M. (2012). Automating exercise generation: A step towards meeting the MOOC challenge for embbeded systems. *Proceedings of the Workshop on Embedded and Cyber-Physical Systems Education* (p. 2). Tampere, Finland: ACM.

Siacua: Interactive Computer Learning System, University of Aveiro. (2017). Retrieved from http://siacua.web.ua.pt.

Singh, R., Gulwani, S., & Rajamani, S. (2012). Automatically Generating Algebra Problems. *Twenty-Sixth AAAI Conference on Artificial Intelligence*, (pp. 1620-1627).

Skopljanac-Macina, F., Blaskovic, B., & Pintar, D. (2016). Automated Generation of Questions for Basic Electrical Engineering Education. *Annals of DAAAM*, (pp. 377-385).

APPENDIX 1

The grades obtained in the written exam (T1), in the online test (MT1), in Question 1 of the written exam (Q1) and the number of accesses to SIACUA platform are gathered in Table 3. The marks Q1, T1 and MT1 are in percentage.

Table 3: Marks obtained for all the students in this study, including the marks on Question 1 (Q1), written exam 1 (T1), the online test (MT1) and number of accesses to SIACUA bar on concept 510 (A)

Q1	T1	A	MT1	Q1	T1	A	MT1
16%	25%	40	80%	100%	85%	18	80%
0%	40%	1	60%	20%	30%	4	80%
0%	45%	42	100%	60%	60%	8	80%
48%	55%	17	100%	0%	45%	4	100%
76%	45%	18	80%	20%	45%	1	100%
8%	25%	11	100%	80%	80%	2	80%
12%	50%	11	80%	16%	55%	1	100%
60%	40%	7	100%	20%	55%	10	100%
0%	40%	1	100%	52%	60%	13	80%
0%	35%	17	100%	80%	85%	1	100%
92%	65%	2	100%	0%	65%	1	100%
60%	45%	14	100%	100%	60%	19	80%
0%	40%	14	100%	40%	65%	4	80%
60%	70%	12	100%	80%	70%	8	80%
20%	25%	7	60%	0%	40%	1	80%
0%	25%	1	60%	0%	30%	8	100%
0%	55%	2	100%	80%	80%	3	80%
32%	30%	3	40%	80%	50%	5	100%
0%	45%	1	100%	40%	40%	5	100%
100%	80%	7	80%	100%	90%	3	80%
0%	55%	4	80%	100%	80%	16	80%
60%	65%	193	100%	40%	80%	3	60%
68%	65%	5	100%	60%	65%	1	80%
80%	60%	3	80%	80%	70%	3	100%
80%	75%	1	80%	0%	40%	10	40%
80%	80%	101	100%	52%	55%	14	80%
60%	65%	69	100%	100%	70%	8	100%
20%	35%	28	80%	100%	80%	13	100%

APPENDIX 2

This Appendix consists on the translation of the different instantiations of the parameterized exercise used in the four contexts.

The exercise on Figure 3 is:

The function series defined on $\mathbb{R}$ by $\sum_{n=1}^{+\infty} f_n(x)$ where

$$f_n(x) = \begin{cases} \frac{7}{n^2} & if\ |x| \leq n \\ \frac{8}{x^2} & if\ |x| > n \end{cases}.$$

Choose the correct option:

- Converges pointwise but not uniformly on $\mathbb{R}$.
- Converges uniformly only if $|x| \leq 1$.
- Weierstrass's criterion can't be applied.
- Converges uniformly on $\mathbb{R}$.

- See the detailed answer without answering the question

The detailed answer to this exercise in Figure 4 is:

Answer

Notice that if $|x| \leq n$ then $|f_n(x)| \leq \frac{7}{n^2}$ and if $|x| > n$, $|f_n(x)| \leq \frac{8}{n^2}$, so we can affirm that

$$|f_n(x)| \leq \frac{8}{n^2}$$

For any $x \in \mathbb{R}$. As the series $\sum_{n=1}^{+\infty} \frac{8}{n^2}$ converges, using Weierstrass's Criterion we can state that the series $\sum_{n=1}^{+\infty} f_n(x)$ converges uniformly on $\mathbb{R}$.

The exercise in Figure 5 is

The function sequence defined on $\mathbb{R}_0^+$ by $f_n(x) = \frac{4x^n}{5n+4x^n}$

Choose the correct option:

- Converges uniformly to $f(x) = \begin{cases} 0 & if \quad x \in [0,1] \\ 1 & if \quad x > 1 \end{cases}$.
- Converges pointwise to $f(x) = \begin{cases} 0 & if \quad x \in [0,1] \\ 1 & if \quad x > 1 \end{cases}$.
- Has limit $\lim_{n\to\infty} f_n(x) = 1,\ \forall x \geq 0$.
- Has limit $\lim_{n\to\infty} f_n(x) = 0,\ \forall x \geq 0$.

The exercise included in Figure 6 is the following:

Consider the function series defined on $\mathbb{R}_0^+$ by

$$\sum_{n=1}^{+\infty} \frac{nx}{3nx+1}$$

Justify that this series doesn't converge uniformly.

Finally, the exercise in the written test (Figure 7) is:

Consider the function series

$$\sum_{n=1}^{+\infty} \frac{3}{5n^2 + 8x^2}$$

a. Find the domain of convergence of this series and justify that the convergence is uniform.

b. Justify that the function S, the sum of the series, is continuous in the domain of convergence.