

# Multiple-choice questions in Mathematics: automatic generation, revisited

*Kosaku Nagasaka*

nagasaka@main.h.kobe-u.ac.jp

Kobe University, Japan

## Abstract

*The multiple-choice question is one of very common assessment tools in the e-learning environment. Especially in mathematics learning, the multiple-choice question is used mainly for assessing lower-level cognitive processing (e.g. definitions, simple computations and so on), and there are not so many known studies on the multiple-choice question in mathematics learning: benefit, guiding principle, automatic question generation and so on. In this paper, we show our guiding principle, our whole system based on Moodle and Python (with Jupyter notebook), and some examples of automatic question generators in linear algebra and fundamental calculus.*

## 1 Introduction

The multiple-choice and fill-in-the-blank questions are very common in the e-learning environment (including blended and mobile learnings) since they can be scored automatically. Especially in mathematics learning (except the elementary school mathematics), the multiple-choice question has been used for assessing lower-level cognitive processing (e.g. definitions, simple computations and so on), and the fill-in-the-blank with some mathematical expression input system has been used for assessing more higher-level cognitive skills (e.g. procedural computation problems). In fact, Yang [11] noted that there already exists a question generator in mathematics in the middle 90s (the example is a multiple-choice question).

From the technological point of view, “with mathematical expression input” is very interesting and sounds great hence we have tended to focus on the fill-in-the-blank question. This is very natural since many people think that step by step computations and proofs are important in mathematics learning. For this direction, we have to use some CAS (computer algebra system) or something like that since in general any mathematical object may have multiple representations (e.g.  $2 + \cos^2 \theta = 1 + \sin^2 \theta + 2 \cos^2 \theta = 2 \sin^2 \theta + 3 \cos^2 \theta$ ) and we have to handle them. For this purpose, webMathematica, Möbius Assessment (formerly Maple T.A.) and STACK are widely used, and also there is a new e-learning system using JavaScript (run over browsers but no server required) proposed by Kitamoto et al. [4]. Moreover, preparing a set of good questions of this kind is not easy. Therefore, for example, Nakamura et al. [7] proposed a question sharing system with STACK.

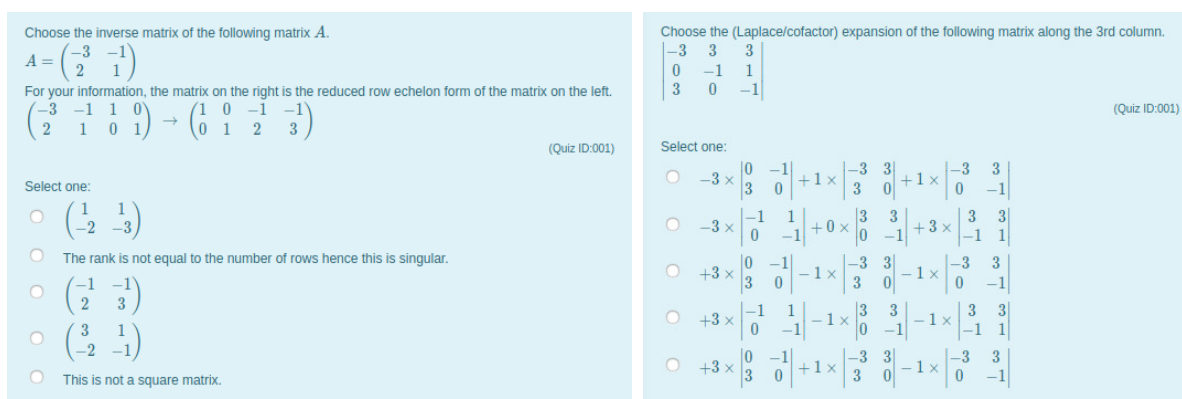


Figure 1: Multiple-choice questions in linear algebra: previewed by Moodle

On the other hand, it seems that “the multiple-choice question” in mathematics learning has been studied very limited. For example, in [1], some very simple generator for the multiple-choice question based on Perl and LaTeX is proposed. They prepared a question text, a set of correct answers and a set of incorrect answers in advance, and their software generates randomly different combinations. In the author’s opinion, this is because that 1) it does not require any frontier technology hence away from the main stream of research topics (e.g. many people generated and used the multiple-choice questions but thought it is trivial hence has not been published), and 2) many people think that it can be replaced with the fill-in-the-blank question or the conventional paper-based assessment (including submitting photo images, PDF files and so on).

However, “the multiple-choice question” has its own useful property and cannot be replaced with others. In fact, Douglas et al. [2] studied the useful situation where the multiple-choice questions are very suitable. Ng et al. [8] reported that some weekly online quizzes equipped in a flipped learning course were very helpful for students (see also [10] for further general information on multiple-choice questions).

In this paper, we are interested in “the multiple-choice question” only in mathematics learning (Figure 1 shows our questions in use) and show its benefit, our guiding principle and the whole system based on Moodle and Python (including automatic question generator), following the well-known drill & practice methodology. This paper is formed as follows. At first, we show our concept and principle in the next section (Section 2), we introduce the KaTeX filter for Moodle in Section 3 and our question generator in Section 4. In the last section (Section 5), we give some remarks.

## 2 Framework with Multiple-choice question

In general, writing a good multiple-choice question is not easy. For example, there are so many rules in the guidelines by Zimmaro [12] (see also [9] for more examples in mathematics). For your information, we briefly summarize some of them as Table 1 (we selected rules that are still important for questions in mathematics). We note that we use the technical terms “question text”, “correct answer (item)” and “incorrect answer (item)” while sometimes they are also called “stem”, “key” and “distractor”, respectively.

Table 1: Multiple-choice question writing rules (summarized)

<p><b>Simple</b> Use the best answer or the correct answer format with simple question text.</p> <p><b>Single</b> Focus on a single problem or idea (skill/knowledge) for each question.</p> <p><b>Honesty</b> Avoid “None of the above” and other trick answer items (be not a puzzle).</p> <p><b>Homogeneous</b> Keep all answer items homogeneous and plausible (no shortcut).</p> <ul style="list-style-type: none"> <li>• Use common errors of students as incorrect answers.</li> <li>• Avoid clueing one item with another (keep items independent).</li> </ul>
--

Select one that is the factorization of $x^2 - 2x - 8$ .			
1. $(x+8)(x+1)$	3. $(x-8)(x+1)$	5. $(x+4)(x+2)$	7. $(x-4)(x+2)$
2. $(x+8)(x-1)$	4. $(x-8)(x-1)$	6. $(x+4)(x-2)$	8. $(x-4)(x-2)$

Figure 2: MCQ example: factoring a polynomial

## 2.1 Mathematics specific issues

One may think that we understand the rules hence just follow them, and there will not be any problem. However, there exist issues that are specific to mathematics. Let us consider a multiple-choice question for “factoring a polynomial” in Figure 2 where we note that all the items should be vertically listed in the actual assessment environment while we use 4 columns format to save the space. Actually, this question is not adequate to assess the skill to factor a polynomial since this question can be solved by its inverse method (i.e. expansion). For example, it can be solved by a student who can expand a polynomial expression but cannot factor a polynomial hence this breaks the “Single” rule. Moreover, the question in Figure 3 seems no problem. However, even in this short question, that there are only four items (10, 16, 24 and 36) may be able to clue students who are not confident of multiplications. If we cover these cases, the items will be 2, 4, 6, 8, 10, 16, 20, 24, 28, 32, 36 and so on. But this breaks the “Homogeneous” rule again since this rule is self-contradictory in terms of miscalculation that is one of common errors in mathematics.

Consequently, we need to prepare a set of multiple-choice questions preventing the following two issues, if we want to use the multiple-choice question as the direct assessment tool for grading (i.e. the final examination) in mathematics learning.

**Fact 1 (inverse method and miscalculation issues)** *Let  $Q$  be a multiple-choice question with its correct and incorrect items, in mathematics. Then, 1)  $Q$  may have inherently the **inverse method issue** that solving  $Q$  by analyzing or using correct and incorrect items is much easier than solving  $Q$  directly without the items, and 2)  $Q$  may have inherently the **miscalculation issue** that there are so many potential incorrect items corresponding to common miscalculation errors for each (correct and incorrect) items.*

We bought 4 packs of apples where each pack has 6 apples. How many apples we bought in total?			
1. 10	2. 16	3. 24	4. 36

Figure 3: MCQ example: arithmetic word problem

However, we can realize from the questions in Figures 2 and 3, that it is explicitly difficult to make all the multiple-choice questions in mathematics, preventing the inverse method and miscalculation issues. In fact, the inverse method issue can be considered as “the P versus NP problem” in terms of the complexity class theory. For most of people, understanding/constructing a deterministic way that we solve the question directly, is more difficult than doing a nondeterministic way (equivalently) that we check only whether each item is correct or not. In the rest of this section, to resolve these issues, we consider this difficulty from the view point of the difference between the multiple-choice question (MCQ) and fill-in-the-blank question (FBQ).

## 2.2 MCQ and FBQ in Mathematics – strengths and limitations –

The inverse method issue is raised by the existence of correct and incorrect items, and the miscalculation issue is raised by the nonexistence of potential incorrect items while other items exist. Hence these issues are inevitable for the multiple-choice question in mathematics. To overcome this situation, basically we have to use the fill-in-the-blank question instead.

For the fill-in-the-blank question, we must prepare some additional system (CAS): web-Mathematica, Möbius Assessment (formerly Maple T.A.), STACK and so on, since we have to check the correctness of students’ inputs in 2D mathematical text expressions (e.g.  $1/2$ ,  $\cos(x)^2 + \sin(x)^2$  and so on). However, this type of platforms has the following weak points from the practical point of view: 1) **less scalability** that on demand computations by CAS for generating questions and rating answers require large computing resources, 2) **less portability** that porting codes between different systems is not easy task, and 3) **less usability** that writing any 2D mathematical expression on mobile devices is painful (or at least more difficult than just selecting items).

**Fact 2 (strengths and limitations of FBQ)** *The fill-in-the-blank question solves the inverse method and miscalculation issues of the multiple-choice question, and is much flexible though it is less scalable, less portable, and less usable from the administrative point of view.*

On the other hand, the multiple-choice question does not require any specific computing resource hence it has strong points: 1) **more scalability** that the multiple-choice question does not require any on demand computation by CAS, 2) **more portability** that questions can be represented in the Moodle XML format (or other common formats) hence we can use any conventional XML tool to convert, edit and port them, and 3) **more usability** that the multiple-choice question is very easy to answer definitely.

**Fact 3 (strengths and limitations of MCQ)** *The multiple-choice question is more scalable, more portable, and more usable from the administrative point of view though it has the inverse method and miscalculation issues, and is less flexible.*

Consequently, the multiple-choice question and the fill-in-the-blank question should be used complementary: MCQ for the place where the quantity of questions is required, and FBQ for the place where the quality of questions is required. Moreover, following the blended assessment [3] is recommended if the system is not compatible with FBQ (hence MCQ is only the option).

## 2.3 MCQ Guideline in Mathematics learning as self-assessment

By the discussions in the previous subsections, we propose the following framework of multiple-choice questions for self-assessment that is compatible with the well-known drill & practice methodology, and the result reported by Ng et al. [8] (weekly online quizzes).

### Fundamentals

- Use for drills and self-assessment (i.e. quantity  $\succ$  quality).
- Avoid any prevention of inverse method and miscalculation issues.
- Avoid any prevention of cheating (since self-assessment and not cost-effective).

### Questions

- Use a simple question text in the best (correct) single answer format.
- Focus on a single problem (idea/skill/knowledge) for each question.
- Ensure informative but no strong assumption.

### Items

- Be honest and Avoid meaningless items (each one corresponds each common error).
- Avoid non-homogeneous (not plausible) items even if they are common errors.
- Include symmetric and uniform items (i.e. items generated by a structured way).

The rules of fundamentals and some of others just follow the discussions in the previous subsections. The “single answer” rule of questions follows the “single focus” rule since in general it is easier to recognize a mistake and learn by the mistake than the multiple answers/selections format (could be a final examination, but not suitable for self-assessment). The “informative” rule of questions follows that any strong assumption may assess the reading comprehension but just many information may require the mathematical deeper thinking (i.e. a higher-level cognitive processing). The “homogeneous” rule of items follows that any incorrect answer far away from the correct one cannot give students a good implicit feedback. Moreover, any incorrect item must be a good educational material (not only a wrong choice). Therefore, we have the “symmetric” rule of items.

### 2.3.1 example: eigenvector

The formal definition of eigenvector is as follows.  $\vec{x} \in V$  is an eigenvector of the linear transformation  $f$  in the vector space  $V$  over the field  $K$  if  $\vec{x}$  is a nonzero vector and satisfy  $f(\vec{x}) = \lambda\vec{x}$  for some scalar in  $K$ . Asking the correct definition or the correct word for some blanks cannot be used as drills since they do not have any variety of questions. In this case, the question pattern “choose one satisfying the specified property” is useful. The knowledge of the property is required and students can take different instances<sup>1</sup> repeatedly.

The example in Figure 4 basically follows the guideline. We can construct many instances of this (question) pattern (i.e. drills) since there are quite many possibilities of linear transformations and vectors even if we restrict the elements to tiny integers. The question is very

---

<sup>1</sup>To distinguish each actual quiz that students try to solve from its question pattern, we call each quiz “an instance quiz of question pattern” (instance in short) here and hereafter.

Choose an eigenvector of the linear transformation: $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2, \vec{x} \mapsto \begin{pmatrix} 3 & 0 \\ 1 & 2 \end{pmatrix} \vec{x}$ .				
1. $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$	2. $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$	3. $\begin{pmatrix} 1 \\ -2 \end{pmatrix}$	4. $\begin{pmatrix} -2 \\ 1 \end{pmatrix}$	5. $\begin{pmatrix} -2 \\ -2 \end{pmatrix}$

Figure 4: MCQ example: assessing eigenvector

Choose the set of coefficients appeared in the factors of the factorization of $x^2 - 2x - 8$ .			
1. $\{1, 8\}$	3. $\{1, -8\}$	5. $\{1, 2, 4\}$	7. $\{-4, 1, 2\}$
2. $\{-1, 1, 8\}$	4. $\{-1, 1, -8\}$	6. $\{-2, 1, 4\}$	8. $\{-4, -2, 1\}$

Figure 5: MCQ negative example: assessing factoring a polynomial

simple and the items are homogeneous and symmetric. Moreover, the zero vector is one of common errors since students sometimes do not understand that the zero vector is included in eigen spaces but not an eigenvector.

### 2.3.2 negative example: factoring a polynomial

In the beginning of this section, we described the inverse method issue with the example in Figure 2. To prevent the inverse method about factoring a polynomial, it is required not to give any factor in any format. For example, choosing a set of factors (e.g.  $\{x - 4, x + 2\}$ ) is equivalent to the item  $(x - 4)(x + 2)$ . Even if we give one of factors (e.g.  $x - 4$ ), the question can be solved by dividing the given polynomial by each item. Moreover, one may reach the unusual question as in Figure 5, however, this is not a good question since this question does not follow any mathematical (traditional/conventional) format hence it misleads the learner. Therefore, we have proposed the fundamental rules, and it is better to use the question in Figure 2 as is.

## 3 Moodle KaTeX filter

In this section, we consider the students' experience with the multiple-choice question from the viewpoint of the speed of rendering mathematical expressions. To make the experience better we want to use KaTeX which is a JavaScript library for TeX math rendering on the web, and is initially developed by Khan Academy (see <https://katex.org/> for further information). One may think that the Moodle has the MathJax filter hence every mathematical TeX expression will be rendered on the client-side, and any other TeX rendering libraries is redundant. This is partially correct and not correct for our framework since the rendering speed is very important for the students' experience especially with many questions at once. We note that the first rule of our guideline is "Use for **drills** and self-assessment (i.e. quantity  $\succ$  quality)".

To use KaTeX, in general we just insert some link and script tags into the header of each page (see the auto-render extension in the documentation of KaTeX). However, in our case, our mathematical courses in Moodle are only some tiny parts of the whole courses (corresponding to all the lectures in Kobe university). Hence it is not easy to disable the MathJax filter and insert the link and script tags of KaTeX into the header of each page on the server-side. Therefore, we had to write our Moodle KaTeX filter [5] that basically follows the Moodle MathJax filter and approximately 50 lines are only different.

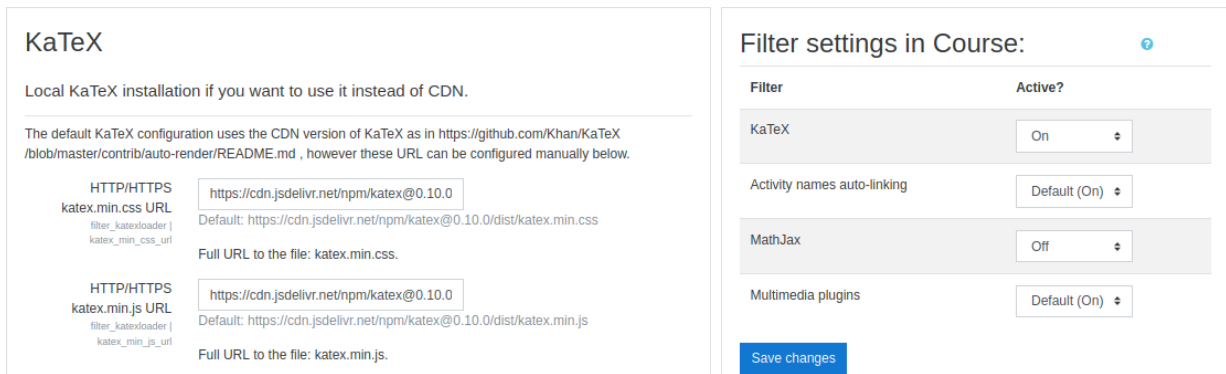


Figure 6: KaTeX configuration in site and filter settings in course

With this filter we can enable the KaTeX only for each course of ours as in Figure 6. By this filter and the power of KaTeX, we can assign a set of multiple-choice questions with many mathematical expressions to the students very smoothly.

## 4 Moodle XML Question Generator for Python

In our framework, we need many multiple-choice questions to give the students a set of periodic drills and/or self-assessment activities. For this purpose, we had been looking for a suitable question generator but could not find anything we wanted. Therefore, we implemented our desired multiple-choice question generator in Mathematica 9 in 2017, and ported it to Python (with Jupyter notebook) in 2019 [6]. We note that our module has 99 sample generators for linear algebra and 40 sample generators for fundamental calculus (as of summer, 2020) but they are in Japanese (only 10 or less examples in English).

In the rest of this paper, we describe the generator and its examples. We note that our module requires some external libraries (e.g. `sympy`, `lxml` and so on). Mostly they are included in the Anaconda individual distribution (<https://www.anaconda.com/>) but some example generator may require additional modules (e.g. `timeout_decorator`).

### 4.1 Getting started

Our module name is `moodle_xqg` hence loading the module by `moodle_xqg.core` is required mandatory, and `moodle_xqgb.qbank.common` is not required but recommended since this includes some useful functions (e.g. deleting duplicate answers).

```
import moodle_xqg.core as mxqg
import moodle_xqgb.qbank.common as mxqg_common
import random # for random integer generations
import sympy # for manipulating mathematical expressions
```

With this module, to generate multiple-choice questions, we have to done the following 3 steps: 1) implement a question-specific generator as a child class of the module's class, 2) generate questions by an instance of the generator, and 3) export it into the Moodle XML format and import it in Moodle.

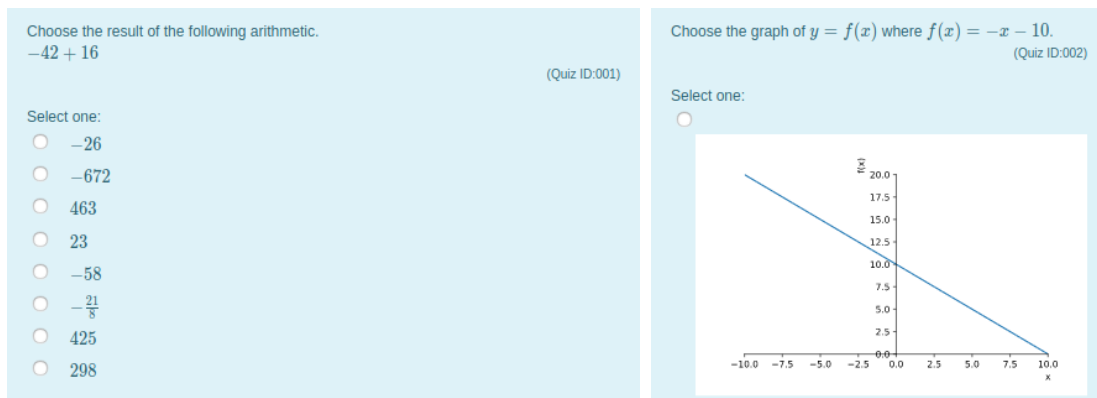


Figure 7: Getting started examples by Moodle XML Question Generator for Python

#### 4.1.1 minimum example of question generator

At first, we implement a question generator (corresponding to each question category/pattern) that is defined as a child class of `Question`, and the child class must implement the following 5 methods: `question_generate`, `correct_answers_generate`, `incorrect_answers_generate`, `question_text` and `answer_text`. `question_generate` must return an instance of `Quiz` class. `correct_answers_generate` and `incorrect_answers_generate` must return a list of answers (each answer must be a dict with the keys: `fraction`, `data` and `feedback`). `question_text` and `answer_text` convert each instance of `Quiz` and answer to be in the HTML format. Moreover, each question generator class has to generate the identifier of each quiz instance (each multiple-choice question in the same question category/pattern) and store it in the instance variable `Quiz.quiz_identifier`. The module distinguishes quizzes (instances) by this value. Therefore, the module generates only one quiz instance if you forget to implement this part. The following program defines a child class.

```
class integer_addition_simple(mxqg.Question):
    def __init__(self, lower=-10, upper=10):
        # the range of integers to be generated.
        self.lower = lower
        self.upper = upper
    def question_generate(self, _quiz_number=0):
        quiz = mxqg.Quiz(name='addition of integers', quiz_number=_quiz_number)
        # generates a quiz data (exclude 0)
        quiz.data = [random.choice(list(
            set(range(self.lower, self.upper + 1))-{0})) for i in range(2)]
        quiz.quiz_identifier = hash('{}+{}'.format(quiz.data[0], quiz.data[1]))
        # generates the correct answer
        quiz.data.append(quiz.data[0] + quiz.data[1])
        # generates the correct choice
        ans = { 'fraction': 100, 'data': quiz.data[2] }
        quiz.answers.append(ans)
        return quiz
```



```

def correct_answers_generate(self, quiz, size=1):
    # nothing to do since we already have the correct answer generated
    pass
def incorrect_answers_generate(self, quiz, size=4):
    # generates incorrect choices randomly (not recommended though)
    answers = []
    ans = { 'fraction': 0, 'data': quiz.data[2] }
    while len(answers) < size:
        ans['data'] += random.randint(-9,9)
        ans['feedback'] =
            'think and compute carefully. it is just a simple addition.'
        if ans['data'] != quiz.data[2]:
            answers.append(dict(ans)) # need to copy the object if reuse it
            answers = mxqg_common.answer_union(answers)
    return answers
def question_text(self, quiz):
    return 'Choose the result of the following arithmetic.<br />' +
        mxqg_common.sympy_str_to_text('{ }+{ }'.format(quiz.data[0],
                                                    quiz.data[1]))
def answer_text(self, ans):
    return mxqg_common.sympy_expr_to_text(ans['data'])

```

To generate actual quizzes, the `generate` method is used. However, before calling the method, we have to generate an instance of the question generator. The number of quizzes to be generated is specified by the `size` option of `generate` (in the default, 10 quizzes will be generated).

```

ias = integer_addition_simple()
ias_quizzes = mxqg.generate(ias, category='minimum example', size=5)

```

The result of question generation is an instance of the `Quizzes` class. We can confirm the resulting questions in the Jupyter notebook by the `preview` method.

```

ias_quizzes.preview(size=1)

```

In the following preview output, the scores and feedbacks will be appeared as mouse over text.

#### output on Jupyter notebook

Choose the result of the following arithmetic.

$-3 + 3$

(Quiz ID:005)

- 0
- 9
- 8
- 17
- 15

Finally, the `save` method writes the quizzes in the specified file in the Moodle XML format that can be imported to Moodle.

```
ias_quizzes.save('integer_addition_simple_in_english.xml')
```

Moreover, the left example of Figure 7 is generated by the following definition where new generator `integer_addition` is implemented as a child class of `integer_addition_simple` and only the `incorrect_answers_generate` is overridden.

```
class integer_addition(integer_addition_simple):
    def incorrect_answers_generate(self, quiz, size=7):
        answers = []
        ans = { 'fraction': 0 }
        # using incorrect binary operator
        ans['data'] = quiz.data[0] - quiz.data[1]
        ans['feedback'] = 'The given arithmetic is not a subtraction.'
        if ans['data'] != quiz.data[2]:
            answers.append(dict(ans))
        ans['data'] = quiz.data[0] * quiz.data[1]
        ans['feedback'] = 'The given arithmetic is not a multiplication.'
        if ans['data'] != quiz.data[2]:
            answers.append(dict(ans))
        if quiz.data[1] != 0:
            ans['data'] = sympy.Rational(quiz.data[0], quiz.data[1])
            ans['feedback'] = 'The given arithmetic is not a division.'
            if ans['data'] != quiz.data[2]:
                answers.append(dict(ans))
        answers = mxqg_common.answer_union(answers)
        if len(answers) >= size:
            return random.sample(answers, k=size)
        # generates incorrect choices randomly (not recommended though)
        count = 0
        ans['data'] = quiz.data[0] + quiz.data[1]
        while len(answers) < size and count < 10:
            count += 1
            ans['data'] += random.randint(self.lower, self.upper)
            ans['feedback'] =
                'think and compute carefully. it is just a simple addition.'
            if ans['data'] != quiz.data[2]:
                answers.append(dict(ans))
            answers = mxqg_common.answer_union(answers)
        return answers
```

## 4.2 Further examples

### 4.2.1 elementary row operations

This is very simple example to assess the knowledge of elementary row operations that may be usually assessed via computation of the reduced row echelon form in a paper-based assessment.

Which elementary row operation is required to get the following matrix  $B$  from the following matrix  $A$ .

$$A = \begin{pmatrix} -1 & 1 & 1 & 3 \\ -1 & 0 & -2 & -2 \\ 0 & -2 & 1 & 3 \end{pmatrix}, B = \begin{pmatrix} -3 & 1 & -3 & -1 \\ -1 & 0 & -2 & -2 \\ 0 & -2 & 1 & 3 \end{pmatrix}$$

(Quiz ID:003)

Select one:

- multiply the 1st row by 2.
- interchange the 2nd and 1st rows.
- add the 1st row multiplied by 2 to the 3rd row.
- interchange the 3rd and 1st rows.
- add the 2nd row multiplied by 2 to the 1st row.

Choose the adjugate matrix of the following  $A$ .

$$A = \begin{pmatrix} -2 & 1 & -3 \\ 0 & 2 & 3 \\ 1 & 1 & -3 \end{pmatrix}$$

For your information, we have the minors:  
 $|A_{1,1}| = -9, |A_{1,2}| = -3, |A_{1,3}| = -2, |A_{2,1}| = 0, |A_{2,2}| = 9, |A_{2,3}| = -3, |A_{3,1}| = 9, |A_{3,2}| = -6, |A_{3,3}| = -4$

(Quiz ID:003)

Select one:

- $\begin{pmatrix} -9 & -3 & -2 \\ 0 & 9 & -3 \\ 9 & -6 & -4 \end{pmatrix}$
- $\begin{pmatrix} -9 & 0 & 9 \\ -3 & 9 & -6 \\ -2 & -3 & -4 \end{pmatrix}$

- $\begin{pmatrix} -9 & 0 & 9 \\ 3 & 9 & 6 \\ -2 & 3 & -4 \end{pmatrix}$
- $\begin{pmatrix} -9 & 3 & -2 \\ 0 & 9 & 3 \\ 9 & 6 & -4 \end{pmatrix}$

Figure 8: Elementary row operations and adjugate matrix examples

By our simple short multiple-choice questions, students can have many exercises step by step (independently of computing the echelon form). In this example, to follow the homogeneous rule, all the incorrect items use the same integer for scalar multiplications (i.e. 2) with row and column indices that run through all the possibilities. One of other possible incorrect answer generations is to list the elementary row operations by which at least the first column is equal to that of matrix  $B$ .

#### 4.2.2 adjugate matrix

In textbooks of linear algebra, we may have the question “compute the adjugate matrix of the following matrix”. To solve this problem, students may have to learn the definitions of the adjugate matrix and minors, and the skill to compute the determinants (minors). It means that this question cannot be used directly for a multiple-choice question. At least we must select one knowledge or skill. In our course, students learn the definition of determinant, the rule of Sarrus, the (Laplacian) determinant expansion by minors, in this order. By this fact, we suppose that we focus on the definition of adjugate matrix, and assessing other knowledge and skill (minors and determinants) are already done in advance. As the result, we have the question on the right in Figure 8 where we give all the minors explicitly in the question text.

To follow the guideline (especially, the homogeneous and symmetric rules), there are only five possible incorrect items:  $c_{ij} = (-1)^{i+j}|A_{i,j}|$ ,  $c_{ij} = (-1)^{i+j+1}|A_{i,j}|$ ,  $c_{ij} = (-1)^{i+j+1}|A_{j,i}|$ ,  $c_{ij} = (+1)^{i+j}|A_{i,j}|$  and  $c_{ij} = (+1)^{i+j}|A_{j,i}|$  where  $c_{ij}$  denotes the  $(i, j)$  element of candidate of adjugate matrix of  $A$  and  $|A_{i,j}|$  denotes the  $(i, j)$  minor of  $A$ . For example, its implementation with our Python module is as follows (we only quote the `incorrect_answers_generate` method).

```
def incorrect_answers_generate(self, quiz, size=5):
    answers = []
    ans = { 'fraction': 0 }
    [_dim, _matrix, _adjugate] = quiz.data
    ans['feedback'] = 'be careful on subscript indices of elements of
                        adjugate matrix.'
    _incorrect_ans = [[sympy.Matrix(self._make_minor(_matrix, _i, _j)).det()
```

```

        *(-1)**(_i+_j) for _j in range(_dim)] for _i in range(_dim)]
if _incorrect_ans != _adjugate:
    ans['data'] = _incorrect_ans
    answers.append(dict(ans))
ans['feedback'] = 'be careful on signs and subscript indices of
                    elements of adjugate matrix.'
_incorrect_ans = [[sympy.Matrix(self._make_minor(_matrix,_i,_j)).det()
                    *(-1)**(_i+_j+1) for _j in range(_dim)] for _i in range(_dim)]
if _incorrect_ans != _adjugate:
    ans['data'] = _incorrect_ans
    answers.append(dict(ans))
_incorrect_ans = [[sympy.Matrix(self._make_minor(_matrix,_i,_j)).det()
                    *(1)**(_i+_j) for _j in range(_dim)] for _i in range(_dim)]
if _incorrect_ans != _adjugate:
    ans['data'] = _incorrect_ans
    answers.append(dict(ans))
ans['feedback'] = 'be careful on signs of elements of adjugate matrix.'
_incorrect_ans = [[sympy.Matrix(self._make_minor(_matrix,_j,_i)).det()
                    *(-1)**(_i+_j+1) for _j in range(_dim)] for _i in range(_dim)]
if _incorrect_ans != _adjugate:
    ans['data'] = _incorrect_ans
    answers.append(dict(ans))
_incorrect_ans = [[sympy.Matrix(self._make_minor(_matrix,_j,_i)).det()
                    *(1)**(_i+_j) for _j in range(_dim)] for _i in range(_dim)]
if _incorrect_ans != _adjugate:
    ans['data'] = _incorrect_ans
    answers.append(dict(ans))
answers = common.answer_union(answers)
if len(answers) >= size:
    return random.sample(answers,k=size)
return answers

```

### 4.2.3 power series expansion

In a calculus course, we may have the question “find the Taylor polynomial of degree 2 approximating the function at the point  $x = 1$ ” and so on. To solve this problem, students may have to learn the definitions of the Taylor expansion, higher derivatives and factorials. It means that this question cannot be used directly for a multiple-choice question. In general, we can suppose that students already learnt the definition of higher derivatives and factorials. By this fact, we suppose that we focus on the definition of the Taylor expansion, and assessing other knowledge and skill are already done in advance. Moreover, differentiating non-simple functions does not contribute anything to assess the Taylor expansion since it is included in the skill of higher derivatives.

As the result, we have the question on the left in Figure 9. To follow the guideline (especially, the homogeneous and symmetric rules), there are only five possible incorrect items:  $\sum_{n=0}^{\infty} f^{(n)}(a) \cdot (x - a)^n$ ,  $\sum_{n=0}^{\infty} f^{(n)}(a) \cdot (x + a)^n$ ,  $\sum_{n=0}^{\infty} \frac{1}{n!} f^{(n)}(a) \cdot (x + a)^n$ ,  $\sum_{n=0}^{\infty} \frac{1}{n} f^{(n)}(a) \cdot (x - a)^n$

<p>Choose the approximation as a polynomial of degree 3 (based on Maclaurin series) of the following function <math>f(x)</math>.</p> $f(x) = \frac{1}{3x-4}$ <p style="text-align: right;">(Quiz ID:158)</p> <p>Select one:</p> <p><input type="radio"/> <math>-\frac{9}{64}x^2 - \frac{3}{16}x - \frac{1}{4}</math></p> <p><input type="radio"/> <math>-\frac{27}{128}x^3 - \frac{9}{64}x^2 - \frac{3}{16}x - \frac{1}{4}</math></p> <p><input type="radio"/> <math>-\frac{81}{128}x^3 - \frac{9}{32}x^2 - \frac{3}{16}x - \frac{1}{4}</math></p> <p><input type="radio"/> <math>-\frac{27}{256}x^3 - \frac{9}{64}x^2 - \frac{3}{16}x - \frac{1}{4}</math></p> <p><input type="radio"/> <math>-\frac{9}{32}x^2 - \frac{3}{16}x - \frac{1}{4}</math></p>	<p><math>f(x, y)</math> is a <math>C^2</math> function and satisfies the following conditions at <math>(a, b) = (0, -3)</math>. Choose the best one in terms of extremum of <math>f(x, y)</math> at <math>(a, b)</math>.</p> <p><math>f_x(a, b) = 0, f_y(a, b) = 0, f_{xx}(a, b) = 2, f_{yy}(a, b) = 7, f_{xy}(a, b) = -1</math></p> <p style="text-align: right;">(Quiz ID:001)</p> <p>Select one:</p> <p><input type="radio"/> With only the information given, it is undecidable whether the function <math>f(x, y)</math> takes any extremum or not at <math>(a, b) = (0, -3)</math>.</p> <p><input type="radio"/> The function <math>f(x, y)</math> does not take any extremum at <math>(a, b) = (0, -3)</math>.</p> <p><input type="radio"/> The function <math>f(x, y)</math> takes a local minimum at <math>(a, b) = (0, -3)</math>.</p> <p><input type="radio"/> The function <math>f(x, y)</math> takes a local maximum at <math>(a, b) = (0, -3)</math>.</p>
--	---

Figure 9: power series expansion and extremum condition examples

and  $\sum_{n=0}^{\infty} \frac{1}{n} f^{(n)}(a) \cdot (x + a)^n$ . Additionally, other possibilities are the above five items with  $a = 0$  if  $a \neq 0$ , and with different degree from the specified.

#### 4.2.4 extremum condition of bivariate function

In a calculus course, computing local extrema is one of important problems hence in general we want to assess this knowledge. In this case, the most difficulty for generating questions automatically is that there are not so many functions whose extrema can be calculated by hands. However, by the “Single” rule and considering the most important knowledge for computing local extrema, we have that partial differentiating is not required in this assessment, and the important thing is the necessary and sufficient condition:

for the given  $C^2$  function  $f(x, y)$  and the point  $(a, b)$ ,

- $f_x(a, b) = 0$  and  $f_y(a, b) = 0$  if  $(a, b)$  is a stationary point,
- $(a, b)$  is a saddle point if  $f_{xx}(a, b)f_{yy}(a, b) - f_{xy}(a, b)^2 < 0$ ,
- $(a, b)$  is a local extremum point if  $f_{xx}(a, b)f_{yy}(a, b) - f_{xy}(a, b)^2 > 0$ ,
  - $(a, b)$  is a local maximum point if  $f_{xx}(a, b) < 0$  and  $f_{yy}(a, b) < 0$ ,
  - $(a, b)$  is a local minimum point if  $f_{xx}(a, b) > 0$  and  $f_{yy}(a, b) > 0$ ,
- anything possible if  $f_{xx}(a, b)f_{yy}(a, b) - f_{xy}(a, b)^2 = 0$ .

As the result, we have the question on the right in Figure 9 where we simply give some virtual information on the values  $f_x(a, b)$ ,  $f_y(a, b)$ ,  $f_{xx}(a, b)$ ,  $f_{yy}(a, b)$  and  $f_{xy}(a, b)$ , and we do not have to take care of the actual function.

## 5 Conclusion

Recently we are interested in the evaluation of our multiple-choice questions instead of its technological background since we already have made our system (guideline, KaTeX filter, Moodle XML question generator). Especially in the first semester of 2020, we have four courses with our format below: linear algebra 1 (1 credit, 46 students), linear algebra 2 (1 credit, 46 students), fundamental calculus 1 (1 credit, 16 students), fundamental calculus 2 (1 credit, 16

students). Our format is a kind of the flipped learning (full online due to COVID-19) where the activity outside the classroom consists of online materials on Moodle (on-demand PDF slides and videos) and our multiple-choice questions as “weekly self-assessment quiz” on Moodle, and the in-class activity consists of paper-based assignments (submitting their photos to Moodle and responding by image files with marks and comments by hands on tablets) and online video meetings. A brief report on the students’ evaluation of the courses said that my course was selected as one of the best 4 linear algebra 1 courses (out of 18 courses). However, the survey has not yet done hence we quote another result in 2017 as follows.

## 5.1 Short report on students’ responses

In 2017, we used some sets of multiple-choice questions as “weekly self-assessment quiz” on Moodle for two courses: linear algebra 1 (1 credit, 32 students) and linear algebra 2 (1 credit, 23 students) where the scores of these quizzes contributed to 10% of the final course assessment. We assigned 7 self-assessments for linear algebra 1 and 6 for linear algebra 2. Each self-assessment consisted of 10 multiple-choice questions, and the students were able to take them several times (the highest score will be only counted). We note that we had generated 200 instances (quizzes) for each generator (more than 30 generators in total) by the Mathematica version of Moodle XML question generator (more than 6000 instances in total). 61.8% of students (85.0% of respondents) answered that the self-assessments had been helpful or moderately helpful. Moreover, we quote some of positive and negative comments as follows.

### Positive

- we can try the same examination several times with different instances.
- randomness of questions is useful for exercise.
- we can try it even on buses, trains and so on.
- online quizzes do not waste papers.
- weekly quizzes keep our motivation of learning.

### Negative

- we really want to have a feedback/exposition if incorrect.
- the volume should be adjusted to the topic.
- it is preferable to display all the questions on the same page.
- paper based is preferable since the final paper examination is paper based.
- there may exist a question not solvable by hand.

## Acknowledgements

This work was supported by JSPS KAKENHI Grant Number 18K02941 that is the research project “Development of learning materials suitable for mobile devices which induce students’ awareness and deepen understanding” by K. Yoshitomi with M. Kameda, K. Kanenishi and I.

## References

- [1] BRITO, I., ALMEIDA, J. J., AND MACHADO, G. J. Mvgen: Multi Version Question Generation for Math Courses. *Open Education Studies* 1, 1 (2019), 146–150.
- [2] DOUGLAS, M., WILSON, J., AND ENNIS, S. Multiple-choice question tests: a convenient, flexible and effective learning tool? a case study. *Innovations in Education and Teaching International* 49, 2 (2012), 111–121.
- [3] JANIER, J. B., AND SHAFIE, A. B. Blended assessment: A strategy for classroom management. In *Proceedings of the 14th Asian Technology Conference in Mathematics* (2009), pp. 193–199.
- [4] KITAMOTO, T., KANEKO, M., AND TAKATO, S. E-learning system with computer algebra based on javascript programming language. In *Proceedings of the 23th Asian Technology Conference in Mathematics* (2018), pp. 123–132.
- [5] NAGASAKA, K. Moodle Katex filter. [https://wwwmain.h.kobe-u.ac.jp/~nagasaka/research/xml\\_quiz/](https://wwwmain.h.kobe-u.ac.jp/~nagasaka/research/xml_quiz/), 2018. ver. 2019021313 (2019).
- [6] NAGASAKA, K. Moodle XML Question Generator for python. [https://wwwmain.h.kobe-u.ac.jp/~nagasaka/research/xml\\_quiz/](https://wwwmain.h.kobe-u.ac.jp/~nagasaka/research/xml_quiz/), 2019. ver. 0.3 (2020).
- [7] NAKAMURA, Y., TANIGUCHI, T., AND NAKAHARA, T. Development of an item bank system for the mathematics e-learning system stack. In *Proceedings of the 18th Asian Technology Conference in Mathematics* (2013).
- [8] NG, W. L., TEO, K. M., WONG, K. Y., AND KWAN, K. L. M. The delivery role and assessment role of computer-based technology in a flipped university mathematics course. In *Proceedings of the 24th Asian Technology Conference in Mathematics* (2019), pp. 185–194.
- [9] TORRES, C., LOPES, A., BABO, L., AND AZEVEDO, J. Developing multiple - choice questions in mathematics. In *ICERI2009 Proceedings* (16-18 November 2009), 2nd International Conference of Education, Research and Innovation, IATED, pp. 6218–6229.
- [10] XU, X., KAUER, S., AND TUPY, S. Multiple-choice questions: Tips for optimizing assessment in-seat and online. *Scholarsh. Teach. Learn. Psychol.* 2, 2 (2016), 147–158.
- [11] YANG, W.-C. Technological tools have enhanced our teaching, learning and doing mathematics, what is next? In *Proceedings of the 19th Asian Technology Conference in Mathematics* (2014).
- [12] ZIMMARO, D. Writing good multiple-choice exams. <https://facultyinnovate.utexas.edu/sites/default/files/writing-good-multiple-choice-exams-fic-120116.pdf>, 2004. workshop material.