РАЗДЕЛ III.
ТЕОРИЯ И МЕТОДИКА ОБУЧЕНИЯ И ВОСПИТАНИЯ

УДК 517
DOI: 10.18384/2310-7251-2016-4-77-85

СОЗДАНИЕ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ И НОВЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ С ИСПОЛЬЗОВАНИЕМ MATLAB ПРИ ИЗУЧЕНИИ ЛИНЕЙНОЙ АЛГЕБРЫ

Власова Е.А., Попов В.С., Пугачев О.В.
Московский государственный технический университет им. Н.Э. Баумана
105005, Москва, ул. 2-я Бауманская, 5, Российская Федерация

Аннотация. Предлагаются генераторы заданий, созданные в среде MATLAB, обеспечивающие формирование фонда оценочных средств для проведения текущих и промежуточных аттестаций при обучении по дисциплине «Линейная алгебра». Показано использование созданного в MATLAB генератора матриц с целыми собственными значениями для формирования банка тестовых задач. Изложены способы наглядного геометрического представления действия линейного оператора в двухмерном и трехмерном пространствах. Приведены примеры.

Ключевые слова: MATLAB, генератор заданий, целочисленные матрицы, собственные значения, жорданова форма, симметричные матрицы, геометрическое представление линейного оператора.

CREATION OF A FUND OF ASSESSMENT TOOLS AND NEW EDUCATIONAL TECHNOLOGIES WITH THE USE OF MATLAB IN THE STUDY OF LINEAR ALGEBRA

E. Vlasova, V. Popov, O. Pugachev
Bauman Moscow State Technical University
2-ya Baumanskaya ul. 5, 105005 Moscow, Russia

Abstract. Our generators of tasks created in the MATLAB environment provide the formation of a fund of assessment tools for current and intermediate certification in teaching the subject ‘Linear algebra’. We demonstrate the use of the MATLAB-created generator of matrices with integer eigenvalues for the formation of the bank of test tasks. We describe methods for visual

© Власова Е.А., Попов В.С., Пугачев О.В., 2016.

geometric representation of the action of a linear operator in two-dimensional and three-dimensional spaces. Examples are presented.

Key words: MATLAB, generator of tasks, integer matrices, eigenvalues, Jordan canonical form, symmetric matrix, geometric action of a linear operator.

ВВЕДЕНИЕ

Одним из важных методов проверки знаний является тестирование. Этот метод имеет ряд преимуществ перед традиционными устными и письменными экзаменами, недостатками которых является высокая организационная сложность, большая трудоемкость работ, ограниченное (регламентированное) время проверки, присутствие субъективного и психологического факторов. Он также стимулирует учебно-познавательную активность студентов [1–2]. Тестирования должны быть как промежуточными, так и итоговыми.

Процесс тестирования предполагает наличие большого банка задач для каждого раздела тестируемых математических курсов. Создание банка задач – довольно сложная процедура, требующая соблюдения определённых требований к содержанию и виду таких задач: однозначность их формулировки, определенность метода решений, единственность получаемого результата. Создание «вручную» набора тестового материала во многих случаях оказывается малоэффективным занятием и отнимает довольно много времени у преподавателя. Поэтому использование современных математических пакетов (МП) существенно облегчает процесс создания такого рода тестовых наборов задач.

Для создания банка тестовых задач могут быть использованы системы MathCad и MATLAB, которые имеют встроенную матричную и комплексную арифметику, поддерживают выполнение операций с векторами, матрицами и массивами данных, используют общепринятый способ изображения математических объектов и удобную операционную среду, которая позволяет формулировать задачи и получать решения в обычной математической форме, не прибегая к рутинному программированию.

В данной работе предлагаются генераторы заданий, созданные в среде MATLAB, обеспечивающие формирование фонда оценочных средств для проведения текущих и промежуточных аттестаций при обучении по дисциплине «Линейная алгебра».

В статье представлены также способы наглядного представления действия линейного оператора в двумерном и трёхмерном пространствах. Владение этими способами позволяет успешно использовать новые образовательные технологии при изучении абстрактной и сложной для усвоения дисциплины «Линейная алгебра» [3]. Принцип наглядности и компьютерные технологии тесно взаимосвязаны, и их грамотное сочетание может привести к хорошим результатам в обучении. Наглядные и технические средства обучения способствуют не только эффективному усвоению соответствующей информации, но и активизируют познавательную деятельность обучающихся, развивают способность увязывать теорию с практикой, повышают интерес к учению, делают процесс обучения более доступным.

Генераторы целочисленных матриц с целыми собственными значениями

При составлении или обновлении задач на нахождение собственных значений и собственных векторов линейных операторов и приведение их к жордановой форме, которые студенты должны решать на контрольной работе или дома без использования компьютера и не прибегая к приближенным вычислениям, а также выполняя тестовые задания для самопроверки знаний, возникает трудность: нужно составить такую матрицу, чтобы собственные значения были целыми, и чтобы жордановы клетки имели определённый размер.

Для решения этой проблемы можем предложить преподавателям использовать следующий созданный в MATLAB генератор матриц 3×3 с целыми собственными значениями. Сначала с помощью генератора случайных чисел выбираем три целых числа – собственные значения матрицы, некоторые из них могут совпадать. Далее определяем один из трёх видов жордановой формы матрицы:

$$A1 = \begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix}, A2 = \begin{pmatrix} a & 1 & 0 \\ 0 & a & 0 \\ 0 & 0 & b \end{pmatrix}, A3 = \begin{pmatrix} a & 1 & 0 \\ 0 & a & 1 \\ 0 & 0 & a \end{pmatrix}.$$

После этого создаём случайную целочисленную матрицу T, посредством которой матрицу A преобразуем из жордановой формы в произвольную по формуле $T^{-1}AT = M$. Получающаяся матрица M имеет рациональные коэффициенты, но нам желательно получить целые. Для этого матрицу M умножаем на целые числа от 1 до 12, получается матрица N, и выбираем тот множитель, при котором элементы матрицы N наиболее близки к целым. Однако, множитель, не превышающий числа 12, может оказаться недостаточным, в этом случае пользователь видит число err, заметно отличающееся от 0, и вынужден запускать программу заново (это случается примерно в 60% случаев).

Приведём скрипт программы (рис. 1) и пример её работы с пояснениями.

```
r=round(rand(1,3)*10-5); %Три случайных целых числа от -5 до 5 будут
a=r(1); b=r(2); c=r(3); %собственными значениями (некоторые могут
совпадать).
A1=[a 0 0;0 b 0;0 0 c]; %Диагональная матрица
A2=[a 1 0;0 a 0;0 0 b]; %Матрица с жордановой клеткой 2 порядка
A3=[a 1 0;0 a 1;0 0 a]; %Матрица с жордановой клеткой 3 порядка
A=A2 %НАДО ВЫБРАТЬ ОДНУ ИЗ ТРЕХ: A1, A2 или A3.
T=round(rand(3)*10-5) %Случайная матрица из чисел от -5 до 5
M=inv(T)*A*T;   %M -- матрица, подобная A (возможен сбой, если T
вырождена). Элементы M могут оказаться дробными.
for i=1:12
 N=i*M;      %Умножаем M на числа от 1 до 12
 er(i)=max(max(N-round(N))); %и оцениваем отклонение от целых.
end
[err j]=min(er);
err   %будет близко к 0, если матрица N целочисленная;
    %если же err не 0, то надо повторно запустить программу. Сбоев
примерно 60%.
N=round(j*M) %Искомая целочисленная матрица (округлим, чтобы
избавиться от "00000" или "99999")
eig(N)  %Собственные значения матрицы N.
```

Рис. 1. Скрипт программы генерирования целочисленной матрицы с целыми собственными значениями.

Пример 1. Пусть выбраны собственные значения $a = 3$ кратности 2 и $b = 1$, тип $A2$ жардановой формы матрицы. Используем предложенный выше генератор целочисленной матрицы с целыми собственными значениями:

```
A = Матрица с жордановой клеткой 2 порядка – жорданова форма искомой матрицы
 3 1 0
 0 3 0
 0 0 1
T = Случайная матрица из чисел от –5 до 5 – матрица перехода
 -3  0  2
  4 -3  0
 -5  5  0
err = оказалось близко к 0
 1.0658e-14
N = Искомая целочисленная матрица
 18 -12 0
 16 -10 0
 22 -21 6

ans = Проверка: собственные значения матрицы N
 6
 2
 6
```

Если матрица симметрична, то для неё существует ортонормированный базис из собственных векторов. Рассмотрим задачу генерирования целочисленных симметричных матриц с целыми собственными значениями. Если действовать таким же способом, как описано выше, то придётся создавать случайную ортогональную матрицу перехода T из рациональных чисел, что очень сложно.

Здесь применим другой подход: будем генерировать случайные целочисленные симметричные матрицы и из них выбирать те, чьи собственные значения целые. Выбирать придётся из многих сотен, поэтому процедура выбора поручается системе MATLAB. Будем создавать по 500 таких матриц и выбирать ту, у которой собственные значения наиболее близки к целым (рис. 2). Примерно в 3/4 случаев они будут точно целыми хотя бы для одной матрицы из 500.

```
N=500;     %Из N матриц с вер. 75% найдется нужная.
B=round(rand(N,6)*20-10); %N строк по 6 случ. целых от -10 до 10
for i=1:N
 A=[B(i,1) B(i,2) B(i,3); B(i,2) B(i,4) B(i,5);    B(i,3) B(i,5)
B(i,6)];     %i-ая симметричная матрица
 er(i)=norm(eig(A)-round(eig(A))); %насколько собственные значения
близки к целым
end;
[err j]=min(er)       %Погрешность минимальна для j-ой матрицы.
A=[B(j,1) B(j,2) B(j,3);B(j,2) B(j,4) B(j,5);B(j,3) B(j,5) B(j,6)]
%Это j-ая матрица.
%Полученная матрица A -- искомая, ТОЛЬКО ЕСЛИ число err близко к 0.
eig(A) %Проверяем, целые ли собственные значения.
```

Рис. 2. Скрипт программы генерирования симметричной целочисленной матрицы с целыми собственными значениями.

Пример 2. Используем предложенный выше генератор целочисленной симметричной матрицы с целыми собственными значениями:

```
err = Погрешность минимальна для j-ой матрицы
 0

j =
 414

A = j-ая матрица
 -1  3 -6
  3 -5 -4
 -6 -4  1

ans = Собственные значения матрицы A
 -7
 -6
  8
```

Наглядное представление действия линейного оператора

Покажем, как в среде MATLAB можно представить геометрическую интерпретацию действия линейного оператора, заданного сначала в двумерном, а затем в трехмерном линейном пространстве.

```
A=[1 1;4 1];   %На плоскости дан линейный оператор A.
v=[1;0]; w=[0;1]; %Сначала рассмотрим стандартный базис.
%Ломаная задается матрицей: 1-я строка x, 2-я строка y. Столбец --
вершина ломаной.
L=zeros(2,5);  %1-я и 5-я вершины -- начало координат.
L(:,2)=v;L(:,3)=v+w;L(:,4)=w; %L -- квадрат, натянутый на v и w.
M=A*L;      %M -- образ квадрата L под действием A.
x=L(1,:);y=L(2,:);plot(x,y); %Рисует квадрат L,
hold on       %затем на той же картинке
x=M(1,:);y=M(2,:);plot(x,y)  %рисует параллелограмм M.
figure       %НОВАЯ КАРТИНКА
[V,D]=eig(A);v=V(:,1);w=V(:,2); %Теперь v и w собственные вектора A.
L=zeros(2,5);     %1-я и 5-я вершины -- начало коорд.
L(:,2)=v;L(:,3)=v+w;L(:,4)=w; %L -- параллелограмм, натянутый на
вектора v и w.
M=A*L;       %M -- образ L под действием A.
x=L(1,:);y=L(2,:);plot(x,y); %Рисует параллелограмм L,
hold on       %затем на той же картинке
x=M(1,:);y=M(2,:);plot(x,y)  %рисует параллелограмм M.
```

Рис. 3. Скрипт программы создания геометрической интерпретации действия линейного оператора на плоскости.

Действие линейного оператора на плоскости (матрицу задает пользователь) можно изобразить как преобразование параллелограмма, натянутого на базисные вектора. Рассмотрим сначала стандартный ортонормированный базис (1;0), (0;1), затем – базис из собственных векторов. Параллелограмм задаётся как ломаная с 5 вершинами, координаты которых записаны в матрицу из 5 столбцов и 2 строк (координаты первой вершины совпадают с координатами последней). Умножение слева на матрицу A линейного оператора преобразует вершины исходного параллелограмма в вершины его образа. Параллелограмм, построенный на стандартном базисе (единичный квадрат) преобразуется произвольно. В базисе из собственных векторов картина яснее: каждый собственный вектор умножается на своё собственное значение, при этом сохраняет направление или меняет на противоположное. Приведём скрипт программы (рис. 3) и полученные геометрические образы (рис. 4).

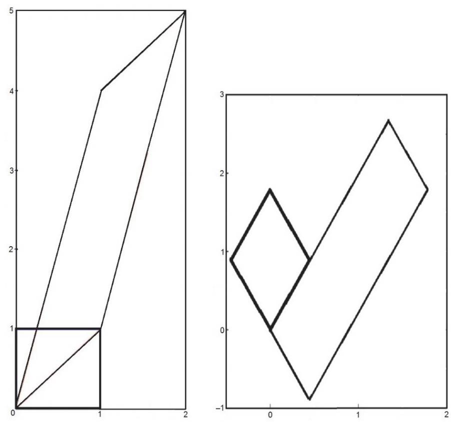

Рис. 4. Преобразование параллелограмма в стандартном и собственном базисах.

Рассмотрим аналогичный способ наглядного представления действия линейного оператора в трёхмерном пространстве. Вместо квадрата построим куб на стандартном ортонормированном базисе (1; 0; 0), (0; 1; 0), (0; 0; 1) и преобразуем его оператором с матрицей A. Затем возьмём параллелепипед, построенный на собственных векторах оператора A, и так же его преобразуем.

```
A=[-2.5 2 -2;2 0 0.5;-2 0.5 0]; %Дан линейный оператор с матрицей A.
a=rand(1,3)-0.5; b=rand(1,3)-0.5; %a и b два случайных вектора-строки.
a=a/norm(a);      %Ортонормируем a и b.
b=b-(a*b')*a;b=b/norm(b);   %Это будет базис плоскости П.
P(1,:)=a;P(2,:)=b;    %P матрица проекции на плоскость П.
%Сначала рассмотрим стандартный базис.
u=[1;0;0]; v=[0;1;0]; w=[0;0;1];
%Ломаная задаётся матрицей: 1-я строка x, 2-я строка y, 3-я строка z.
%Столбец -- вершина ломаной.
L=zeros(3,16);     %1-я и 11-я вершины -- начало координат.
L(:,2)=u;L(:,3)=u+w;L(:,4)=u;L(:,5)=u+v;L(:,6)=u+v+w;
L(:,7)=u+v;L(:,8)=v;L(:,9)=v+w;L(:,10)=v;
L(:,12)=w;L(:,13)=w+u;L(:,14)=w+u+v;L(:,15)=w+v;L(:,16)=w;
        %L -- куб, натянутый на u, v и w.
M=A*L;       %M -- образ куба L под действием A.
PL=P*L;PM=P*M;     %Проецируем L и M на П.
x=PL(1,:);y=PL(2,:);plot(x,y); %Рисует проекцию куба L,
hold on       %затем на той же картинке
x=PM(1,:);y=PM(2,:);plot(x,y) %рисует проекцию параллелепипеда M.
%Пусть теперь u, v и w -- собственные вектора A.
[V,D]=eig(A);u=V(:,1);v=V(:,2);w=V(:,3); figure %НОВАЯ КАРТИНКА
L=zeros(3,16);
L(:,2)=u;L(:,3)=u+w;L(:,4)=u;L(:,5)=u+v;L(:,6)=u+v+w;
L(:,7)=u+v;L(:,8)=v;L(:,9)=v+w;L(:,10)=v;
L(:,12)=w;L(:,13)=w+u;L(:,14)=w+u+v;L(:,15)=w+v;L(:,16)=w;
%L -- параллелепипед, натянутый на вектора u, v и w.
M=A*L;       %M -- образ L под действием A.
PL=P*L;PM=P*M;     %Проецируем L и M на П.
x=PL(1,:);y=PL(2,:);plot(x,y); %Рисует проекцию параллелепипеда L,
hold on       %затем на той же картинке
x=PM(1,:);y=PM(2,:);plot(x,y) %рисует проекцию параллелепипеда M.
```

Рис. 5. Скрипт программы действия линейного оператора в пространстве.

Поскольку параллелепипед имеет 8 вершин нечётного порядка, три ребра придётся проходить по 2 раза, и параллелепипед конструируется в виде ломаной с 16 вершинами. Поскольку мы сможем увидеть, строго говоря, не сам параллелепипед, а его проекцию на плоскость, в программе предусмотрен случайный выбор проекционной плоскости, на которой взяты ортонормированные векторы $\overline{a}$ и $\overline{b}$. Пользователь может запускать программу несколько раз, получая проекции указанных четырёх параллелепипедов на различные плоскости и выбирая оптимальный «ракурс». Приведём скрипт программы (рис. 5) и полученные геометрические образы (рис. 6). На этом рисунке цифры, расставленные по осям – координаты не самих векторов, а их проекций.

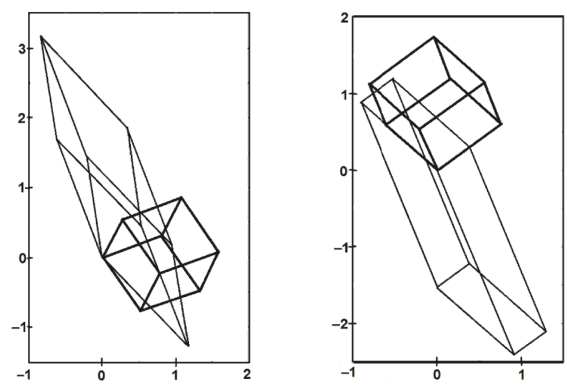

Рис. 6. Преобразование параллелепипеда в стандартном и собственном базисах.

ЗАКЛЮЧЕНИЕ

Использование новейших компьютерных технологий, в том числе интерактивных компьютерных систем и математических пакетов, в рамках дисциплин математического образования позволяет проводить мобильное обновление банка заданий, формировать методическое обеспечение индивидуальной траектории обучения студентов.

На примере применения MATLAB для составления контрольных заданий по дисциплине «Линейная алгебра» показаны конкретные алгоритмы, которые можно использовать в практической деятельности.

Изложенная в статье методика создания в среде MATLAB генераторов заданий может быть применена для обучения других математических дисциплин.

ЛИТЕРАТУРА

1. Власова Е.А., Грибов А.Ф., Попов В.С., Латышев А.В. Принципы модульно–рейтинговой системы преподавания высшей математики // Вестник Московского государственного областного университета. Серия Физика–Математика. 2013. № 3. С. 93–99.
2. Власова Е.А., Грибов А.Ф., Попов В.С., Латышев А.В. Развитие мотивационных стимулов обучения в рамках модульно–рейтинговой системы организации учебного процесса // Вестник Московского государственного областного университета. Серия: Физика–Математика. 2014. № 1. С. 48–53.

3. Власова Е.А., Попов В.С., Латышев А.В. Методические аспекты обеспечения дисциплины «Линейная алгебра» в техническом университете // Вестник Московского государственного областного университета. Серия Физика–Математика. 2015. № 3. С. 69–85.

REFERENCES

1. Printsipy modul'no–reitingovoi sistemy prepodavaniya vysshei matematiki [Principles of the module–rating system of teaching mathematics] / Vlasova E.A., Gribov A.F., Popov V.S., Latyshev A.V. // Vestnik Moskovskogo gosudarstvennogo oblastnogo universiteta. Seriya: Fizika–matematika [Bulletin of Moscow Region State University. Series: Physics and Mathematics]. 2013. no. 3. pp. 93–99.
2. Razvitie motivatsionnykh stimulov obucheniya v ramkakh modul'no–reitingovoi sistemy organizatsii uchebnogo protsessa [The development of motivational incentives for learning within the module–rating system of organization of educational process], Vlasova E.A., Gribov A.F., Popov V.S., Latyshev A.V. // Vestnik Moskovskogo gosudarstvennogo oblastnogo universiteta. Seriya: Fizika–matematika [Bulletin of Moscow Region State University. Series: Physics and Mathematics]. 2014. no. 1. pp. 48–53.
3. Vlasova E.A., Popov V.S., Latyshev A.V. Metodicheskie aspekty obespecheniya distsipliny «Lineinaya algebra» v tekhnicheskom universitete [Methodological aspects of the discipline 'Linear algebra' at the technical University] // Vestnik Moskovskogo gosudarstvennogo oblastnogo universiteta. Seriya: Fizika–matematika [Bulletin of Moscow Region State University. Series: Physics and Mathematics]. 2015. no. 3. pp. 69–85.

ИНФОРМАЦИЯ ОБ АВТОРАХ

Власова Елена Александровна – кандидат физико-математических наук, доцент кафедры прикладной математики, Московский государственный технический университет им. Н.Э. Баумана;
e-mail: elena.a.vlasova@yandex.ru

Попов Владимир Семенович – кандидат физико-математических наук, доцент кафедры прикладной математики, Московский государственный технический университет им. Н.Э. Баумана;
e-mail: vspopov@bk.ru

Пугачев Олег Всеволодович – доктор физико-математических наук, доцент кафедры прикладной математики, Московский государственный технический университет им. Н.Э. Баумана;
e-mail: opugachev@yandex.ru

INFORMATION ABOUT THE AUTHORS

Elena Vlasova – candidate of physical and mathematical sciences, associate professor of the Applied Mathematics Department at Bauman Moscow State Technical University;
e-mail: elena.a.vlasova@yandex.ru;

Vladimir Popov – candidate of physical and mathematical sciences, associate professor of the Applied Mathematics Department at Bauman Moscow State Technical University;
e-mail: vspopov@bk.ru;

Oleg Pugachev – doctor of physical and mathematical sciences, professor of the Applied Mathematics Department at Bauman Moscow State Technical University;
e-mail: opugachev@yandex.ru

БИБЛИОГРАФИЧЕСКАЯ ССЫЛКА

Власова Е.А., Попов В.С., Пугачев О.В. Создание фонда оценочных средств и новых образовательных технологий с использованием MATLAB при изучении линейной алгебры // Вестник Московского государственного областного университета. Серия: Физика-математика. 2016. № 4. С. 77–85.
DOI: 10.18384/2310-7251-2016-4-77-85.

BIBLIOGRAPHIC REFERENCE

E. Vlasova, V. Popov, O. Pugachev. Creation of a fund of assessment tools and new educational technologies with the use of MATLAB in the study of linear algebra // Bulletin of Moscow Region State University. Series: Physics and Mathematics. 2016. no. 4. pp. 77–85.
DOI: 10.18384/2310-7251-2016-4-77-85.