TEACHING UNIVERSITY-LEVEL MATHEMATICS USING MATHEMATICA

Anna Malinova

Abstract. *This paper considers the use of the computer algebra system Mathematica for teaching university-level mathematics subjects. Outlined are basic Mathematica concepts, connected with different mathematics areas: algebra, linear algebra, geometry, calculus and analysis, complex functions, numerical analysis and scientific computing, probability and statistics. The course "Information technologies in mathematics", which involves the use of Mathematica, is also presented - discussed are the syllabus, aims, approaches and outcomes.*

Keywords: Mathematica, computer algebra systems, mathematics education, university-level mathematics
2010 Mathematics Subject Classification: 97U70, 97B70, 97D40

1. Introduction

A variety of mathematical software, such as dynamic geometry systems and computer algebra systems, emerged during the last tree decades. Few examples are Maple, Mathematica, Geometer Sketchpad, Matlab, and Derive. Many case studies have been conducted in order to investigate the actual outcomes of employing these tools in mathematics education. Extensive overviews of such studies are presented in [2], [9], [6], [7], [10], and [4]. Although there are some problems, the common understanding is that in a world with computer-based tools, communications, information, and work environment, education must adopt the new software tools and computer technologies in both teaching and learning and incorporate them in the corresponding subject's curricula.

The recent paper considers the use of the computer algebra system Mathematica [15], [5] for teaching university-level mathematics subjects. In Section two are discussed basic Mathematica concepts connected with different areas of mathematics. Section three presents the course "Information technologies in mathematics", which involves the use of Mathematica. Discussed are the syllabus, aims, and outcomes.

2. Brief overview of basic Mathematica concepts

Mathematica is a program for doing, teaching, and learning mathematics. It provides advanced algorithms for performing both symbolic and numerical calculations. This section presents a brief overview of the main Mathematica concepts in different areas of mathematics that are basic to the university mathematics curricula.

- **Algebra** – use Mathematica to teach various algebraic structures, such as sets, groups, fields, spaces and other. Any algebraic structure can be described as a set or a collection of distinct objects and a finite number of operations that can act on them. In Mathematica, a single concept, a *list* is defined, that is an ordered set of objects, separated by commas and enclosed in braces `{elements}`, or can be defined with the function `List[elements]`[12]. The list is a fundamental concept in Mathematica and could be considered as the basic data type primitive. Therefore Mathematica offers a large collection of list manipulation commands [16]. Sets, and a variety of other concepts (matrices, tables, vectors, arrays, tensors, etc.), are represented as lists. Next is discussed Mathematica's representation of infinite sets:

- Natural, integer, and rational numbers: one of the important features of symbolic mathematics is exact computations or computations with high (arbitrary) precision. Mathematica provides exact results in the arithmetical expressions with integers and automatically simplifies rational numbers $m/n\left(m,n\in\mathbb{Z}\right)$ so that numerator and denominator are relatively prime numbers, i.e. $\gcd(m,n)=1$ (gcd is the greatest common devisor) [12].

- Irrational and real numbers: if the result is an irrational number, Mathematica represents it in an unevaluated form and to obtain the value of such numbers we have to approximate using floating-point arithmetic [12]. Real numbers are represented as floating-point numbers. Mathematica can handle approximate real numbers with any number of digits. The system distinguishes two kinds of approximate real numbers: *arbitrary-precision* numbers, and *machine-precision* numbers or *machine numbers* [16]. Arbitrary precision numbers (also called software-floats) can contain any number of digits, and maintain information on their precision. Machine numbers (also called hardware-floats) always contain the same number of digits. For instance `N[Pi]` gives `3.14159` which is the machine-number approximation to π; `N[Pi,15]` gives `3.14159265358979` which is an arbitrary precision number. In Mathematica the set of real numbers $\mathbb{R}$ can be represented only as the subsets of finite dimension $\mathbb{R}_H$ (hardware-floats) and $\mathbb{R}_S$ (software-floats). Therefore important differences between $\mathbb{R}_H$, $\mathbb{R}_S$ and the field of real numbers $\mathbb{R}$ are considered [12]. For instance, associative laws are not valid if we add long signed numbers that cause overflow (or underflow), that is, whose values are greater (or less) than the maximum (or minimum) machine numbers. Mathematica automatically converts the hardware floats $\mathbb{R}_H$ to software floats $\mathbb{R}_S$ in this case.

- Algebraic and complex numbers: for representing algebraic numbers Mathematica can determine the roots of $P(x)\in\mathbb{Q}$ using the function `Root`

($P(x) = a_0x^0 + a_1x + \cdots a_nx^n$ and $\mathbb{Q}$ is the field of rational numbers). In Mathematica complex numbers are represented in the form of ordered pairs of real numbers $\mathbb{C} = \mathbb{R} \times \mathbb{R} = \{(a,b) | a \in \mathbb{R}, b \in \mathbb{R}\}$, and the imaginary unit i is denoted by I. The complex number $x + iy$ is entered as `x+I*y` or with the function `Complex[x,y]`.

Mathematica represents sets in the form of lists, as discussed above. The functions `Union`, `Intersection`, and `Complement` give a list of the distinct elements, common elements, and the elimination of the elements of sets. More functions of set theory are included in the package `Combinatorica`.

The concept of a mathematical expression is represented in Mathematica as a *symbolic mathematical expression* - a combination of constants, variables, functions, etc., that can be evaluated according to particular predefined rules. In general, Mathematica represents everything (formulas, graphics, data, programs, etc.) as expressions and thus handles the different objects in a uniform way [16]. Mathematica provides an extensive set of functions for mathematical expressions, such as simplification, factoring polynomials, combining terms, evaluating, extracting the numerator, denominator and other (`Simplify`, `FullSimplify`, `Factor`, `Expand`, `Together`, `TrigExpand`, `TrigReduce`, `TrigFactor`, etc.).

Defining functions and mappings to relate a set to another one is a frequent task in mathematics. In Mathematica we can use predefined and user-defined functions, mathematical and procedural functions, pure functions and functions defined in terms of a variable, anonymous and named function. If we define a function as a formalized, named transformation rule, the function name is also an expression. Mathematica provides operations on functions, such as composition, defining a function that takes a function name as an argument, applying functions repeatedly (`Nest`, `NestList`), applying functions to other expressions and objects, e.g. lists, (`Apply`), applying functions to each element of other expressions (`Map`), etc. The above listed features are examples of doing programming with Mathematica. An extensive discussion of Mathematica as a programming environment is given in [13].

Polynomial algebra is the base of symbolic mathematics [12], the core of computer algebra systems. Polynomials in Mathematica are represented beginning with the constant term. Univariate and multivariate polynomials and polynomials over number rings can be defined in different ways.

Solving some problems on group theory in Mathematica based on the functions in the package `Combinatorica`. There also exist other various functions and packages for operations on quotient rings, finite fields, integral domains.

- **Linear algebra** – use Mathematica to teach the algebra of linear spaces and linear transformations between these spaces [1]. In Mathematica vectors, matrices, tensors are represented with lists. A vector is a simple list and a matrix is a list of vectors. The elements of a vector or a matrix may be entered manually as a list or, more conveniently, by the use of built-in commands. Since vectors and matrices are stored as lists, all list operations apply. In addition, there are many specialized functions that are applicable specifically to vectors and matrices, such as functions for computing the matrix product of matrices, dot and cross product of vectors, determinant of matrix, inverse of matrix, transpose of matrix, etc. [3].

- **Geometry** – use Mathematica to teach the most important geometric concepts in a Euclidean space $\mathbb{R}^n$, where $\mathbb{R}^2$ represents the two dimensional plane, and $\mathbb{R}^3$, the three dimensional space. Mathematica provides functions for defining and plotting points and curves in the $\mathbb{R}^2$ and $\mathbb{R}^3$, defining and plotting surfaces in $\mathbb{R}^3$, and specifying the coordinate system to be used as well. Considered also are parametrically defined curves and surfaces, implicitly defined curves, functions for standard geometric shapes and their transformations. For this purpose a symbolic graphics language is provided - Mathematica uses the powerful idea of building up all 2D and 3D graphics from symbolic primitives - which can be manipulated using all standard Mathematica functions, and seamlessly integrated with text, math or tables. Symbolic graphics can also be used as input and can be made dynamic and interactive [14].
- **Calculus and Analysis** – use Mathematica to teach real functions of real variables, differential calculus, integral calculus, series, multivariate and vector calculus, etc. Calculus is one of the examples of using Mathematica for symbolic mathematics. For instance, we can differentiate an expression symbolically, and get a formula for the result. Getting formulas as the results of computations is usually desirable when it is possible [16]. There are many Mathematica packages which implement symbolic mathematical operations. If getting an explicit formula as the result of the computation is impossible, we must resort to numerical methods and approximations.
- **Complex Functions** – Mathematica performs complex arithmetic automatically. All operations are performed by assuming that the basic number system in Mathematica is the complex field $\mathbb{C}$ [12]. As already discussed above, the imaginary unit *i* of the complex number $x + y * I$ is denoted by *I*. In [12] are presented examples of complex functions and derivatives, complex integration, sequences and series, etc.
- **Numerical analysis and scientific computing** – teach how to use Mathematica to construct numerical algorithms, to obtain exact and numerical solutions of problems and compare the results, to evaluate computational errors, to simplify proofs and visualize the obtained solutions, etc. Mathematica tries to find exact, symbolic results for computations whenever possible. There are, however, computations where this is impossible. Then we can resort to numerical routines such as `NIntegrate`, `FindRoot`, and `NDSolve` and obtain an approximate numerical solution. The numerical routines have several options with which we can control and modify the routines. These options are related to round-off errors, precision and accuracy [11].

Approximation of functions and data is provided in Mathematica by various functions for performing interpolation, curve fitting, least square approximations (i.e. `Interpolation`, `FunctionInterpoaltion`, `Interpolating-Polynomial`, `Fit`, `FindFit`), the `FunctionApproximation` package with various functions for interpolation and approximation, and the `Splines` package [12].

- **Probability and Statistics** – as it is discussed in [11] Mathematica has a wealth of information about most standard probability distributions. For each distribution, we can ask for cumulative distribution function, probability density function, quantiles, mean, variance, random numbers, and so on. One very useful property of Mathematica is the ease of random number generation; this makes it very convenient to perform simulations. With Mathematica, we can do all kinds of basic statistical analyses, from descriptive statistics to maximum likelihood, frequencies, confidence intervals, hypothesis testing, analysis of variance (ANOVA), and linear and nonlinear regression. Additional topics include finding clusters of data, smoothing data, local regression analysis, and Bayesian statistics.

3. Real use case example

This section presents the author's experience in teaching the course "Information technologies in mathematics" for first year bachelor students in Business Information Technologies at the Faculty of Mathematics and Informatics of the University of Plovdiv "Paisii Hilendarski". The aim of this course is studying the basic principles of computer algebra systems, as well as the basic principles of creating and sharing (including on the web) of documents with mathematical content. The syllabus includes an introduction into the system Mathematica as an integrated interactive environment for computer aided solving of mathematical problems. The lectures discuss numerical and symbolic computations, expressions, transformations, patterns, lists, visualization of data and graphics, programming, import and export to other formats. Laboratory exercises include solving problems mainly form the areas of linear algebra, analytical geometry, and secondary school analysis. A short description of the laboratory syllabus, connected with using Mathematica, is given bellow:

1) Introduction to Mathematica. Solving problems form the area of algebraic calculations - numerical and symbolic calculations, transforming algebraic and logical expressions. Algebraic equations and inequalities.
2) Solving linear algebra problems using Mathematica. Vectors - entering, affine and metric operations with vectors. Matrices - entering, submatrices and elements of a matrix. Matrix manipulation - transposing, multiplication, elementary transformations.
3) Solving linear algebra problems using Mathematica. Determinants - computation, minors and cofactors. Inverse matrix, rank and basis.
4) Solving linear algebra problems with Mathematica - systems of linear equations.

5) Solving analytical geometry problems using Mathematica. Coordinate systems. Equations of a line and a plane. Intersections, distances and angle. Curves of the second order.
6) Solving analysis problems using Mathematica. Limits and derivatives. Function graph. Optimization problems – global and local optimization in Mathematica, classical optimization (using the critical points).
7) Two- and three-dimensional graphics in Mathematica. Graphics primitives and directives. Parametric graphics. Animated graphics.
8) Files and external operations in Mathematica. Importing and exporting of data and graphics in different formats.

The main goal of the laboratory syllabus is not to teach the different areas of mathematics included in the exercises (it is assumed that students have already taken the considered courses), but rather to involve the students in the computer-aided way of doing mathematics and thus to provide a prerequisite for creating Mathematica-based courses of the mathematics and some of the economics subjects that the students will be studying during the following trimesters.

During the course, students have been provided with a ready-to-use interactive document, a Mathematica notebook, for each of the laboratory exercises. The notebooks have been designed to contain explanatory text, illustrating examples and a number of problems to be solved by the students themselves. A very important feature of the teaching environment based on Mathematica was that the students were able to experiment with the presented examples on their own. In this process, students were stimulated to actively use Mathematica's help information. The Mathematica help browser provides the help information as interactive documents and thus it was also possible for the students to experiment with the help examples. Other Mathematica features, which appeared as important teaching features, were: the possibility to define format styles for the whole document or for a single cell; the possibility to switch a notebook to presentation mode and to use it directly for a multimedia presentation of a high quality; the technique to fold and unfold the sections in a notebook in order to keep the overview.

Finally, the students have become familiar with using specialized functions provided by additional packages that are not loaded when Mathematica is initially invoked. This was often needed during the laboratory exercises in solving linear algebra problems. In addition, some of the problems involved creation of user-defined functions and application of procedural, functional or rule-based programming, as presented in [8].

4. Conclusions

Modern mathematics, respectively modern mathematics education, is already impossible without the involvement of technology. The recent paper provided a brief description of basic Mathematica concepts, related to different areas of mathematics, in order to show the potential of the inclusion of a computer algebra system like Mathematica into university-level mathematics courses. An example of a teaching and learning process which is based on Mathematica was discussed –

the course “Information technologies in mathematics”. The main conclusions of this experience are connected with the enhancing of students’ understanding of mathematics and their interest in mathematical subjects in general. The use of computer algebra systems in finance, engineering, industry and commercial enterprises makes this competence necessary and provides further motivation for the use of these tools into the university mathematics courses.

5. Acknowledgements

This work was partially supported by the RS-2009-M13 project of the Scientific Fund of the University of Plovdiv “Paisii Hilendarski”, Bulgaria.

References

[1] Anton, H., R. Busby, Contemporary Linear Algebra, Willey, USA, 2003.
[2] Artigue, M., Learning mathematics in a CAS environment: The genesis of a reflection about instrumentation and the dialectics between technical and conceptual work, International Journal of Computers for Mathematical Learning, 7 (3), (2002), pp. 245-274.
[3] Don, E., Mathematica, 2nd ed., McGraw-Hill, 2009.
[4] Gocheva-Ilieva, S., A. Rahnev, New Challenges in e-Learning of Mathematics via EVLM and IDeLC Projects, in New Aspects of Engineerig Education, Proc. of the 5th WSEAS / IASME Int. Conf. on Engineering Education (EE'08), Heraklion, Crete Island, Greece, July 22-24, (2008), Published by WSEAS Press, invited paper, pp. 437-442.
[5] Gocheva-Ilieva, S., Introduction into system Mathematica, ExsPress, Gabrovo, Bulgaria, 2009, ISBN: 978-954-490-079-3. (in Bulgarian).
[6] Hoyles, S., R. Noss, What can digital technologies take from and bring to research in mathematics education?, Second International Handbook of Mathematics Education, Dordrecht, Kluwer,(2003), pp. 323-350.
[7] Lavicza, Z., Factors Influencing the Integration of Computer Algebra Systems into University-Level Mathematics Education, International Journal for Technology in Mathematics Education, vol. 14, No. 3, (2007), pp. 121-129.
[8] Malinova, A., Developing programming skills using Mathematica, Proceedings of the Anniversary International Conference “Synergetics and reflection in mathematics education”, Bachinovo, Bulgaria, September 10-12, (2010), pp. 425-435. (in Bulgarian).
[9] Monaghan, J., Computer algebra, instrumentation and the anthropological approach, International Journal for Technology in Mathematics Education, 14(2), (2007), pp. 63-72.
[10] Niess, M., Preparing teachers to teach science and mathematics with technology: Developing a technology pedagogical content knowledge, Teaching and Teacher Education: An International Journal of Research and Studies, vol. 21, No. 5, (2005), pp. 509–523.
[11] Ruskeepaa, H., Mathematica navigator: mathematics, statistics, and graphics, 3rd ed., USA: Elsevier, 2009.

[12] Shingareva, I., C. Lizarraga-Celaya, Maple and Mathematica, 2 nd ed., Springer-Verlag Vienna, 2009.
[13] Wellin, P., S. Kamin, R. Gaylord, An Introduction to Programming with Mathematica, New York: Cambridge University Press, 2005.
[14] Wolfram Documentation Center, 2010, http://reference.wolfram.com/mathematica/guide
[15] Wolfram Research, Mathematica 7.0, 2010, http://www.wolfram.com/products/mathematica.
[16] Wolfram, S., The Mathematica Book, 5th ed. USA: Wolfram Media, 2003.

Anna Atanasova Malinova
Faculty of Mathematics and Informatics
University of Plovdiv
236 Bulgaria Blvd.
4003 Plovdiv, Bulgaria
e-mail: malinova@uni-plovdiv.bg