

## REPORT ON OpenDreamKit DELIVERABLE D2.9

### Demonstrator: interactive books on Linear Algebra and Nonlinear Processes in Biology

MARCIN KOSTUR, JERZY ŁUCZKA, JAN AKSAMIT, JOLANTA MARZEC



Due on	31/08/2018 (M36)
Delivered on	31/08/2018
Lead	University of Silesia (USlaski)
Progress on and finalization of this deliverable has been tracked publicly at: <a href="https://github.com/OpenDreamKit/OpenDreamKit/issues/49">https://github.com/OpenDreamKit/OpenDreamKit/issues/49</a>	

## CONTENTS

1. Introduction	1
2. Interactive book technology	2
3. Nonlinear Processes in Biology	2
4. Lectures on Linear Algebra	3
5. Future work	4
Appendix A. Screenshots	5

## 1. INTRODUCTION

Interactive web pages have always been an attractive tool in education. It has started in the age of Adobe-Flash technology, when a lot of educational material was created. The advent of JavaScript has led to significant improvements in that field: in the first place it created a possibility to author the content using Open Source tools and secondly it increased the portability of the created documents to practically all devices equipped with a modern web browser.

The problem which was persistent in those solutions was the big gap between the authoring process of interactive documents and their usage. Usually the code behind an interactive examples was not very educational and was not supposed to be edited by the student.

Then emerged the concept of interactive widgets in the notebook, appearing first in Mathematica, then being popularized by the SAGE notebook, and recently getting huge traction with the JUPYTER notebook thanks to its wide audience. Interactive widgets, and in particular the very simple to use `@interact` decorator, delivered a versatile solution to this problem: they made it possible to turn with little efforts a mere scientific calculation into a visually appealing interactive application, all within the original environment where the mathematical explorations took place.

The need of including computer technology for teaching at University of Silesia was sparked by interdisciplinary courses. In those fields it is common to master numerical tools for modeling and at the same time to gain an intuition. Integration of modern computer technology allows students to more efficiently analyse a system without tedious symbolic calculations. This has led to systematic integration of SAGE system with science education at the Institute of Physics since 2011. In this deliverable we have created texbooks which consists of c.a. 500 pages with over hundred interactive code cells and code examples and tens of figures. The audience for those

books are students of physics, biophysics, econophysics and medical physics. This work took 25 PM of engagement of four people.

The source code of the interactive books is hosted on github within a dedicated repository of the OpenDreamKit organization: <https://github.com/OpenDreamKit/iODKbook2>.

## 2. INTERACTIVE BOOK TECHNOLOGY

We have used SPHINX as main authoring tool: SPHINX is originally a documentation generator which is popular in the Python world where it's used for the documentation of PYTHON itself, and many other projects. More generally it can be used to author large structured documents which are then exported to various formats, including HTML, PDF, EPub, and now even Jupyter notebooks. SPHINX contains a plugin system enabling to tailor it for particular applications. We used plugins for

- embedding SAGE code as live cells which can be edited, executed and interacted with (see Figure 1); under the hood the computations are run on a remote service thanks to the SageMathCell technology (<https://github.com/sagemath/sphinx-sagecell-ext>); an alternative would be to use Thebelab (see D4.7: "Full featured JUPYTER interface for GAP, PARI/GP, Singular");
- using conditional compilation of parts depending for example on whether the target medium is interactive or not, or whether solutions should be included,
- automatically including JUPYTER notebooks with embedded output;
- producing high quality pdf via  $\text{\LaTeX}$ system.

In the authoring process one can join interactive material in the form of notebooks (based on sagenb or JUPYTER) that can be embedded in full or in part into the SPHINX book. In the HTML export, the material is then rendered as live SageCellServer code that the user can edit, execute, or interact with. It has to be stressed that not only interactive widgets written with `@interact` are useful. Practically all of exercises inside the book are illustrated by such interactive material. It's our experience however that such interactive material should not require or encourage the reader to do anything but small edits to the provided code; indeed such edits are volatile by essence; they take place anonymously in the browser with no storage behind the scene.

The authoring process is very similar to writing a  $\text{\LaTeX}$ paper. It is based on edit-compile-view cycle which is known to most of physicists and mathematicians. In some cases we have used a shared account on a JUPYTER system for this process, where compilation commands were pre-coded in a JUPYTER notebook, which also included a link to web, and pdf versions. In such case it was very easy to help academics less familiar to computer science to just edit and modify the content.

## 3. NONLINEAR PROCESSES IN BIOLOGY

The book is in large extent based on coursework taught at University of Silesia. It was primarily based on sagenb system and gradually was converted into the interactive book with live examples.

The book covers the classical topics in modeling in biology. Methodologically it can be split into the following topics:

- (1) Methods of mathematical modelling of processes based on ODE and PDE.
- (2) Qualitative methods in ODE and PDE with application of Sage. It includes:
  - stationary states of the system,
  - the linear stability of fix points,
  - phase curves,
  - bifurcation diagrams,
  - time-dependent solutions,

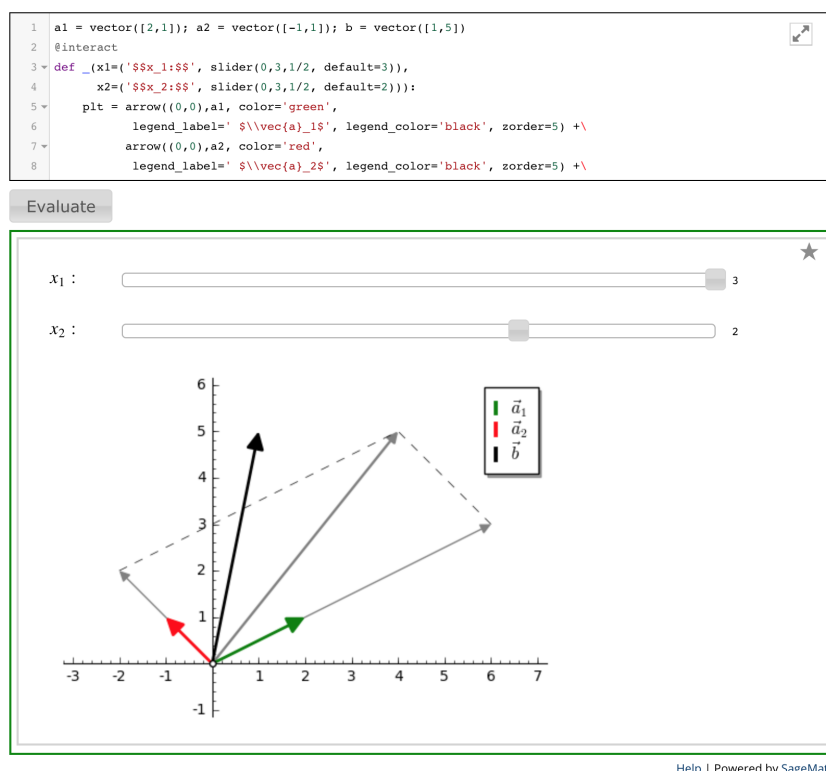


FIGURE 1. An example of `@interact` in SPHINX generated web version of interactive book. The upper box shows the code and the lower box shows the output.

- a graphical presentation of all above elements.

For each subject there is a list of related problems and supplementary tasks as homework. All requires symbolic manipulations as well as certain techniques from numerical analysis like root finding.

- (3) Numerical solving of ODEs. It is complementary to analytical and qualitative methods for ODE.
- (4) Numerical solving of PDE-s - diffusion equation, reaction-diffusion systems and similar. It requires good tools in data processing, usually uses numpy for calculations.

It turned out that conceptually simple methods - like plotting a function of one variable, which depends on a parameter (see Figure. 2), can be very useful for the analysis of models containing analytically untractable expressions. In this particular example, students construct just a simple plot of a function; then, with the help of `@interact` construction, the plot is easily turned into an applet for the analysis of the system.

The book covers the following models:

- one-dimensional systems (Malthus, Verhulst, Ludwig, Alee models);
- two-dimensional systems (Lotka-Volterra, May models);
- kinetics of chemical reactions;
- Belousov-Zhabotinsky reactions;
- models of epidemics.

#### 4. LECTURES ON LINEAR ALGEBRA

*The Linear Algebra book is dedicated to Dr Jan Aksamit for his devotion, and endless patience in work with students. He has lectured linear algebra for physics students for almost 40 years of*

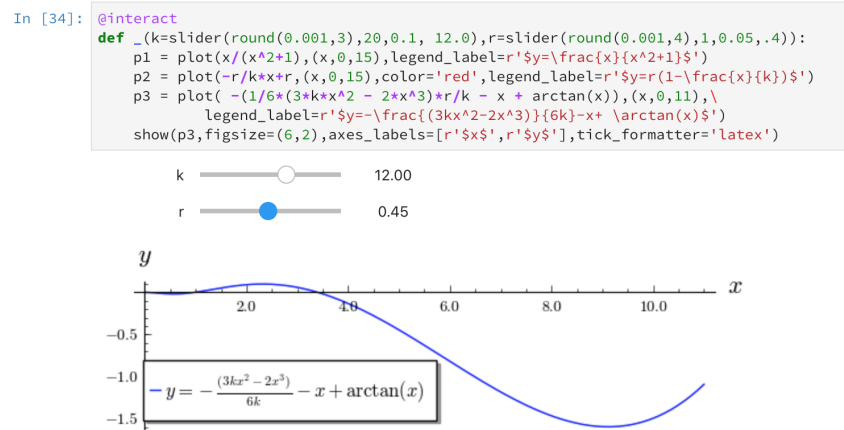


FIGURE 2. An example of `@interact` applied for graphical examination of roots of a function. It is a first step in qualitative analysis of model of tumor growth in a book.

*his life. With restless passion he spent the last years of his life preparing an electronic version of his lectures notes.*

This book is based on Linear Algebra courses for physics students taught in recent years at the University of Silesia. It has been gradually equipped with interactive materials and extended by sections illustrating application of linear algebra in economy, computer science, and other areas. The computations and the interactive part are performed by SAGE, which is embedded in the book.

Along the theory the user becomes familiar with the code used in SAGE. Their learning process is enhanced by various factors:

- numerous visualisations;
- interactive images which allows for the construction of a graphical solution to a problem; the problem may be changed in real time by the user changing the code;
- a place for experiments: the user may fill in the gaps in the code to produce graphical interpretation of a chosen operation;
- possibility to perform the calculations with SAGE within the book; in particular, computations with real data and, thus, real life conclusions are possible;
- some longer coding exercises are equipped with check points (without revealing the answer) to ensure that the discovered solution is correct.

Moreover, not only users benefit from interactive resources, but also learn how to produce them themselves.

## 5. FUTURE WORK

The two open textbooks are put to continuous use for the eponymous courses at university of Silesia; they are frequently updated with new material (new models, exercises, ...), and improved according to feedback from teachers and students. Also there are efforts being done to encourage other lecturers to use them and contribute back.

An important improvement will be inclusion of automatic testing in the compilation procedure.

Currently the material exists in Polish and English versions, which are, however, unsynchronized. In future we plan to update the English version only, mostly because of growing internationality of studies. In some cases, mostly for theoretical parts of linear algebra book, the Polish version will be kept in synchronisation.

## APPENDIX A. SCREENSHOTS

Disclaimer: this report, together with its annexes and the reports for the earlier deliverables, is self contained for auditing and reviewing purposes. Hyperlinks to external resources are meant as a convenience for casual readers wishing to follow our progress; such links have been checked for correctness at the time of submission of the deliverable, but there is no guarantee implied that they will remain valid.

---

## Contents

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Preliminaries and Notation . . . . .	1
1.2	Algebraic Structures . . . . .	4
1.3	Fundamental Concepts in Linear Algebra . . . . .	12
1.4	Group of Permutations . . . . .	18
1.5	References . . . . .	24
<b>2</b>	<b>A Primer on Linear Systems</b>	<b>25</b>
2.1	Column Vectors . . . . .	25
2.2	Geometry of Linear Equations . . . . .	28
2.3	Elimination Method for Systems of Equations . . . . .	36
2.4	Exercises and Problems . . . . .	39
<b>3</b>	<b>Algebra of Matrices</b>	<b>41</b>
3.1	Basic Operations on Matrices . . . . .	41
3.2	Properties of Matrix Operations . . . . .	46
3.3	Vectors and Matrices in Sage . . . . .	50
3.4	Exercises and Problems . . . . .	65
<b>4</b>	<b>Operations upon Matrices</b>	<b>69</b>
4.1	Matrix Formulation of the Elimination Method . . . . .	69
4.2	Matrix Transpose . . . . .	77
4.3	Matrix Inverse . . . . .	80
4.4	Elementary Matrices . . . . .	81
4.5	Matrix Inverse by the Elimination Method . . . . .	84
4.6	Permutation Matrices . . . . .	86
4.7	Appendices . . . . .	91
4.8	Exercises and Problems . . . . .	95
<b>5</b>	<b>Determinants</b>	<b>101</b>
5.1	Definition and Properties of the Determinant . . . . .	101
5.2	The Permutation Expansion . . . . .	105
5.3	Practical Calculation of Determinants . . . . .	109

FIGURE 3. Table of contents of Linear Algebra book.

5.4	Applications of Determinants . . . . .	112
5.5	Application to Hill Cipher . . . . .	120
5.6	Problems . . . . .	126
<b>6</b>	<b>Matrix Decompositions</b>	<b>129</b>
6.1	Triangular Matrices . . . . .	129
6.2	LU Factorization of a Matrix . . . . .	130
6.3	Applications of the LU Decomposition . . . . .	133
6.4	Problems and Exercises . . . . .	135
<b>7</b>	<b>Tensor (Kronecker) Product of Matrices</b>	<b>137</b>
7.1	Definition and Properties . . . . .	137
7.2	Vectorization of Matrices . . . . .	141
7.3	Kronecker Product as a Transformation Matrix . . . . .	143
7.4	Kronecker Sum of Discrete Laplacians . . . . .	144
7.5	Problems with Solutions . . . . .	146
<b>8</b>	<b>Systems of Linear Equations: Theory and Practice</b>	<b>155</b>
8.1	Theorems on Solutions of Linear Equations . . . . .	155
8.2	Example with Discussion . . . . .	160
8.3	Equilibrium of a Linear Set of Masses Connected by Springs . . . . .	166
8.4	Linear and Polynomial Regression . . . . .	168
8.5	Linear Programming . . . . .	179
8.6	Problems . . . . .	187
<b>9</b>	<b>Linear Transformations</b>	<b>191</b>
9.1	Definitions and Terminology . . . . .	191
9.2	Properties and Examples . . . . .	194
9.3	Kernel, Image and the Rank-Nullity Theorem . . . . .	197
9.4	Application to Homogeneous Systems of Equations . . . . .	201
9.5	Matrix Representation of Linear Transformations . . . . .	202
9.6	Linear Operators . . . . .	209
9.7	Change of Basis . . . . .	212
9.8	Application to Computer Graphics . . . . .	222
9.9	Problems . . . . .	235
<b>10</b>	<b>Eigenvalues and Eigenvectors</b>	<b>237</b>
10.1	Formulation of the Eigenproblem . . . . .	237
10.2	Examples of Eigenproblems . . . . .	240
10.3	Application to Systems of Linear Differential Equations . . . . .	247
10.4	Google's Application - PageRank Algorithm . . . . .	255
10.5	Exercises . . . . .	266
<b>11</b>	<b>Unitary Spaces</b>	<b>271</b>
11.1	Inner (Scalar) Product . . . . .	271
11.2	Norm of a Vector . . . . .	276
11.3	Orthogonality of Vectors . . . . .	277
11.4	Gram-Schmidt Orthonormalization and QR Decomposition of a Matrix . . . . .	281
11.5	Hermitian and Unitary Matrices . . . . .	286

FIGURE 4. Table of contents of Linear Algebra book.

11.6	Application of Sage . . . . .	293
11.7	Hermitian and Unitary Operators . . . . .	298
11.8	Normal Operators . . . . .	306
11.9	Problems . . . . .	310
<b>12</b>	<b>Diagonalization of Matrices</b>	<b>313</b>
12.1	Square Matrix as a Linear Operator . . . . .	313
12.2	Similarity Transformation . . . . .	316
12.3	Diagonalization by a Similarity Transformation . . . . .	317
12.4	Matrix Diagonalization - Examples . . . . .	320
12.5	Matrix Diagonalization - Problems . . . . .	328
<b>13</b>	<b>Proofs of Selected Theorems</b>	<b>329</b>
13.1	Fundamental Concepts in Linear Algebra . . . . .	329
13.2	Operations upon Matrices . . . . .	331
13.3	Linear Transformations . . . . .	334
13.4	Eigenvalues and Eigenvectors . . . . .	337
13.5	Unitary Spaces . . . . .	341
<b>14</b>	<b>Problems in Linear Algebra</b>	<b>345</b>
14.1	Systems of Linear Equations . . . . .	345
14.2	Matrices . . . . .	349
<b>15</b>	<b>Indices and tables</b>	<b>353</b>
	<b>Bibliography</b>	<b>355</b>

FIGURE 5. Table of contents of Linear Algebra book.



## Column Picture

The foregoing system of equations

$$\begin{aligned} 2x_1 - x_2 &= 1 \\ x_1 + x_2 &= 5 \end{aligned}$$

can be rewritten as equality of two column vectors:

$$\begin{bmatrix} 2x_1 - x_2 \\ x_1 + x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 5 \end{bmatrix}.$$

The left-hand side can be expressed as a linear combination with coefficients  $x_1, x_2$  :

$$x_1 \begin{bmatrix} 2 \\ 1 \end{bmatrix} + x_2 \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 5 \end{bmatrix}.$$

Now the unknowns  $x_1, x_2$  are weights for column vectors  $\mathbf{a}_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$ ,  $\mathbf{a}_2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$  in a linear combination, which should equal the column vector of constants  $\mathbf{b} = \begin{bmatrix} 1 \\ 5 \end{bmatrix}$ .

### Experiment with Sage:

In the following program the column vectors  $\mathbf{a}_1, \mathbf{a}_2$  and  $\mathbf{b}$  are represented by the geometric vectors  $\vec{a}_1, \vec{a}_2$  and  $\vec{b}$ , respectively. Using the sliders, set the values of coefficients  $x_1$  and  $x_2$  so that the vector  $x_1 \vec{a}_1 + x_2 \vec{a}_2$  (coloured grey) is equal to the vector  $\vec{b}$ .

```
1 a1 = vector([2,1]); a2 = vector([-1,1]); b = vector([1,5])
2
3 @interact
4
5 def _(x1=('$x_1:$$', slider(0,3,1/2, default=3)),
6       x2=('$x_2:$$', slider(0,3,1/2, default=2))):
7
8     plt = arrow((0,0),a1, color='green',
```

Evaluate

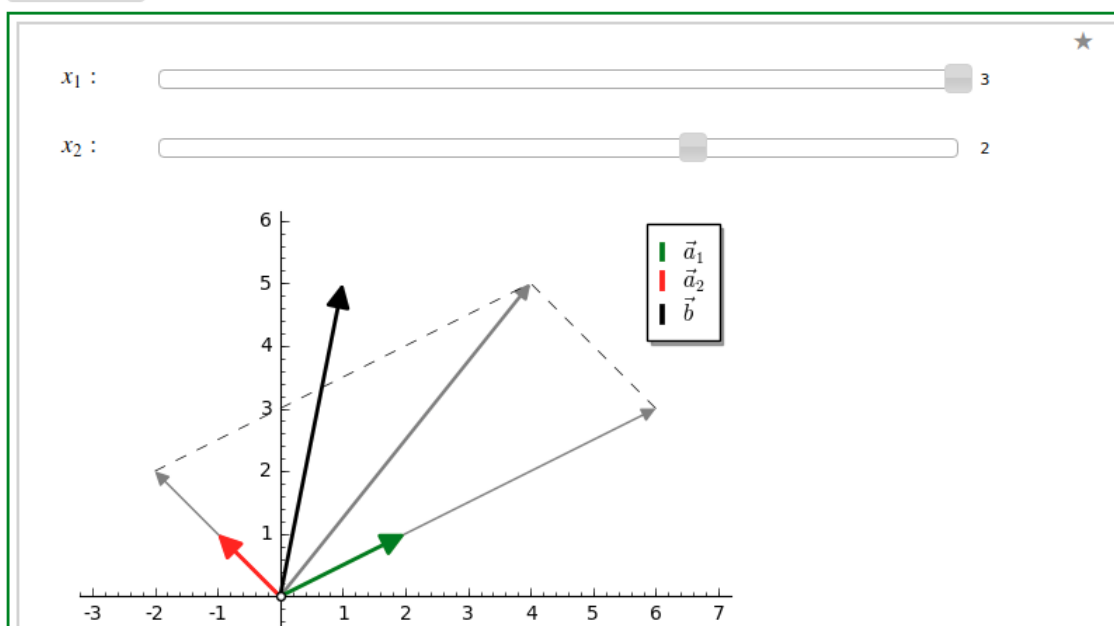


FIGURE 6. An example of interactive element in web version of Linear Algebra book.

Column Picture



The foregoing system of equations

```
.. math::
:nowrap:

\begin{alignat*}{3}
2\,x_1 & \& \{ \, , \} - \{ \, , \} & \& x_2 & \& \{ \, ; \} = \{ \, ; \} & \& 1 \\\
x_1 & \& \{ \, , \} + \{ \, , \} & \& x_2 & \& \{ \, ; \} = \{ \, ; \} & \& 5 \\
\end{alignat*}
```

can be rewritten as equality of two column vectors:

```
.. math::

\left[\begin{array}{r} 2\,x_1 \\ -\,x_2 \end{array}\right] + \begin{array}{r} x_1 \\ x_2 \end{array} = \begin{array}{r} 1 \\ 5 \end{array}
```

The left-hand side can be expressed as a linear combination with coefficients  $\mathbf{x}_1, \mathbf{x}_2$ :

```
.. math::

x_1 \begin{array}{r} 2 \\ -1 \end{array} + x_2 \begin{array}{r} 1 \\ 1 \end{array} = \begin{array}{r} 1 \\ 5 \end{array}
```

Now the unknowns  $\mathbf{x}_1, \mathbf{x}_2$  are weights for column vectors

```
:math: \mathbf{a}_1 = \begin{array}{r} 2 \\ -1 \end{array}
:math: \mathbf{a}_2 = \begin{array}{r} 1 \\ 1 \end{array}
:math: \mathbf{b} = \begin{array}{r} 1 \\ 5 \end{array}
```

in a linear combination, which should equal the column vector of constants

**Experiment with Sage:**

In the following program the column vectors  $\mathbf{a}_1, \mathbf{a}_2$  and  $\mathbf{b}$  are represented by the geometric vectors  $\vec{a}_1, \vec{a}_2$  and  $\vec{b}$  respectively. Using the sliders, set the values of coefficients  $x_1$  and  $x_2$  so that the vector  $x_1\vec{a}_1 + x_2\vec{a}_2$  is equal to the vector  $\vec{b}$ .

**sagecellserver:**

```
a1 = vector([2,1]); a2 = vector([-1,1]); b = vector([1,5])

@interact

def _(x1=slider(0,3,1/2, default=3),
      x2=slider(0,3,1/2, default=2)):

    plt = arrow((0,0),a1, color='green',
        legend_label=' $\vec{a}_1$', legend_color='black', zorder=5) +
        arrow((0,0),a2, color='red',
        legend_label=' $\vec{a}_2$', legend_color='black', zorder=5) +
        arrow((0,0),b, color='black',
        legend_label=' $\vec{b}$', legend_color='black', zorder=5) +
        arrow((0,0),x1*a1, color='gray', width=1, arrowsize=3) +
        arrow((0,0),x2*a2, color='gray', width=1, arrowsize=3) +
        arrow((0,0),x1*a1+x2*a2, color='gray',
        width=1.75, arrowsize=3) +
        line([x1*a1,x2*a2+x1*a1], color='black',
        linestyle='dashed', thickness=0.5) +
        line([x2*a2,x2*a2+x1*a1], color='black',
        linestyle='dashed', thickness=0.5) +
        point((0,0), color='white', faceted=True, size=18, zorder=7)
```

FIGURE 7. The source code of an example of interactive element in web version of Linear Algebra book.