Grant application for the ”Presidential Instructional Technology Fellows Proposal” 5/23/2004

Proposer: Oliver Knill
Department of Mathematics
One Oxford Street
Harvard University
Cambridge, MA 02138

Preceptor of Mathematics
email: knill@math.harvard.edu
tel: (617) 495 5549
URL: http://www.math.harvard.edu/ ˜knill

Proposal Summary

Project unit:	Department of Mathematics
IT Support provided by:	FAS
Project Title:	Multivariable Calculus Flux Visualization
Project Manager:	Oliver Knill, Preceptor of Mathematics
Number of students	One student fluent in Java or Flash + actionscript
Technological aim	Produce one interactive Applet or Flash tool
Reusability	For multivariable calculus courses (also applied math) and physics
Time frame	Summer 2004 (if interested student can be found), otherwise any other semester

The project

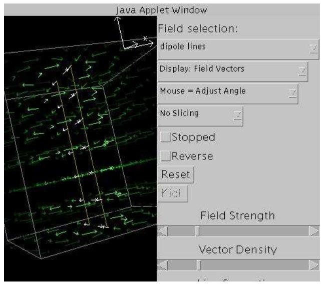
This is proposal for a small project to build a module allowing students to explore the multivariable calculus concept of **flux integrals**.

Integral theorems like Green, Stokes and Gauss theorem are among the most challenging topics in calculus. One reason is that all material comes together there: geometric objects like curves, surfaces, integrations like 2D,3D,line,flux integrals and derivatives grad,curl, div. So, this topic is also the most difficult to teach.

It is natural that educators who want to visualize something, focus on the topics which can be explained well and which are easy to realize. While one finds many applets where students can explore the concept of **rate of change** by tangents or the **Riemann sum approximation** or visualizing **slope fields**, solutions to **differential equations** or **graphing surfaces**, the available resources become rarer if more complex topics are touched and interaction in three dimensions are necessary. But there is no total vacuum neither. One particular good example are the Falstad applets:

http://www.falstad.com/vector3de/

which visualize vectorfields in 3D vector fields in an original way. It is an example, where also the source code is provided. This would be a good starting point for a student to begin coding.


Java Applet Window

Field selection: dipole lines

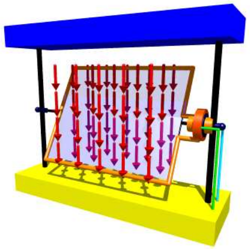
Display: Field Vectors

Mouse = Adjust Angle

No Slicing

□ Stopped

□ Reverse



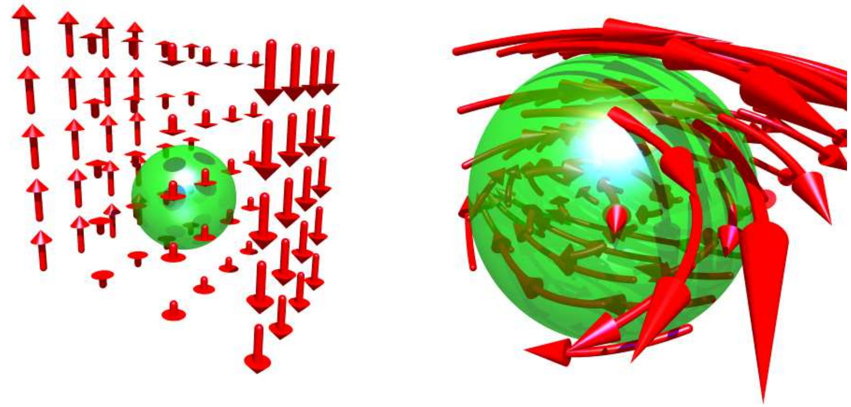
The aim is to build one single original applet enhancing the teaching of integral theorems in multivariable calculus. The applet should allow students to get a feel about flux integrals and what ”curl” means.

The module could be reused in physics for example to get a better feel about the Maxwell equations. I had realized once a dynamo in Povray for animation purposes, but this is not interactive at all.

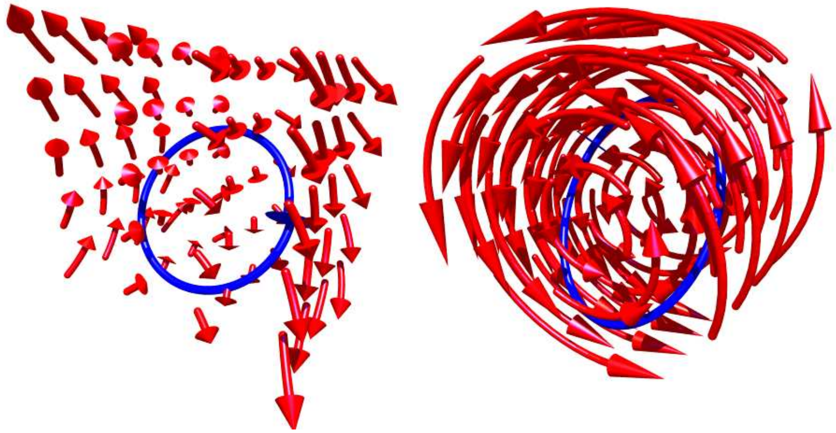
Some studies

Here are some lose ideas:

Divergence exploration:
To get a feel about **divergence** and **Gauss theorem**, one can compute the flux of a vector field through a region. If that flux is positive, then positive divergence prevails inside the region. Students could experiment with different vector fields and different surfaces.



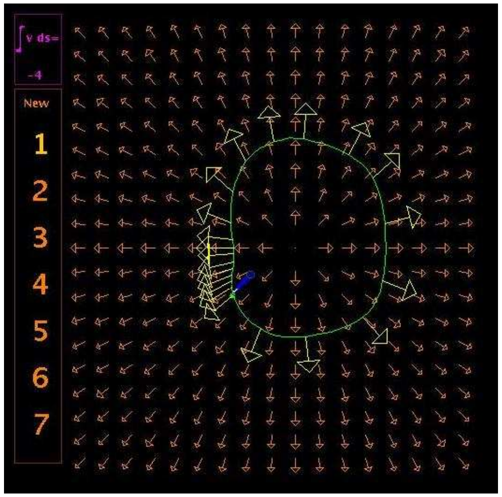
Curl exploration:
To get a feel about **curl** and **Stokes theorem**, one can compute the line integral of a vector field along closed loop. If such a line integral is positive, then some nonvanishing curl is measured along that loop. Students could experiment with different vector fields and different loops.



Previous related work by proposer

Work by myself or with a collaborator:

- In the summer of 2001, Dale Winter and I produced from scratch two modules for the **connected Curriculum project**. The units were in multivariable calculus and consisted of independent teaching modules for **line integrals** and **Greens theorem**. The modules contained Mathematica and Maple worksheets as well as two Java applets. A snapshot of one of those applets is visible below. Students can experiment with different vector fields and measure the line integral along closed curves.



http://www.math.duke.edu/education/ccp/

- In the fall of 2003, I included an exhibit of 12 pages on linear algebra. It can be found on

http://abel.math.harvard.edu/archive/21b_fall_03/exhibit.html

It has some interactive features like illustrations of sound but things are rather limited.

Work done with undergraduate students:

- At Caltech, I supervised during two different summers undergraduate research fellowship (SURF) projects with individual students. Both projects were successful and involved software development and experimentation as well as the writing of scientific papers in the area of dynamical systems theory.
- A project funded by the Provost at Harvard (which I want to test out this summer in my multivariable calculus class) has the aim to build a online chatbot. Three students were working for one year on that project. We have produced encyclopedic content (which can be used in other contexts too) and also built our own chat bot from scratch written in Perl which can interact with computer algebra systems. I currently author with the students an article on the project which produced many interesting ideas.

http://www.math.harvard.edu/˜ knill/sofia

Staff

I expect **one** student to work during the summer on a Flash or Java implementation of the module. Having done things like that in the past, I can gauge the time effort to finish this proejct. It would take me a couple of days of programming to have such a tool done by myself. A lot of thinking has to go into combining what is **technically possible** in a limited time with what is **pedagocially feasible**. Much time has also to go into optimizing for

- **attractiveness**
- **speed**.

The applet has to be

- **original**,
- **simple**
- **self-explanatory**
- **allow exploration**

and most of all to be **be fun**.

Unlike what one might think, this is not an easy task and failure to achieve all goals is possible. One has to expect that the tool has to be improved later.

As the project contact person, I would supervise the student who works on the project, but I would not want to get involved seriously into the programming part. I would prefer to have a **single** student. Splitting things up to two students with different backgrounds is difficult, reduces the chance for success because it also splits the responsibility and increases the coordination efforts.

I could advise on the mathematical and pedagogical side, assist with Java and Flash templates, help with producing art work and assist with ray-tracing, but I do not have the time resources to code things myself, nor would it be realistic to teach a student Java or Flash knowledge. I can however provide access to many of my reference books in both programming languages. The presidential fund might just be the right thing to realize such a thing.

Technology

The project is rather independent and self-contained and limited. The student will need access to the necessary software. For Java, this is no problem. For Flash, the student needs access to the Flash software.

While I think, Flash with action script would be more attractive nowadays (one can also easily include sound for example). I have not seen good prototypes which implement 3D stuff efficiently in Flash. In that respect, it would be easier to build on existing Java tools.

Here is a short overview on advantages and disadvantages for both technology.

Flash	Java
More fancy	More mature
Only PC/Mac authoring	Can also be authored in Unix/Linux
Plugins small and work	Not all plugin/browser combinations work
Much used in interactive games and multimedia content	Much used in education

For both technologies, the question is whether things developed now will still work in 10 years. Since the Java compiler source codes are available and by different companies, I have more faith in the long time survival of Java. Flash authoring is nowadays not possible without the Macromedia tools. On the plugin side, Flash definitely wins. The plugin is small and works on all browsers. So, I guess that for the next 5 years, Flash is currently the best choice to author online interactive content.