ANIMATION AS AN INTERACTIVE TOOL

Andrew Toon[1]

Open University Centre

Mathematics Department, 535A Clementi Road
Singapore 599490

Abstract

Animation can be a powerful medium for illustrating various mathematical concepts and techniques. Here we argue that animation can also be used as an interactive tool. The approach is relatively straightforward to master, requires no plug-in or sophisticated programming and is ideal for "on the spot" construction and use.

---

[1]Email: andrewtoon@sim.edu.sg

1 Introduction

Computer technologies, especially computer algebra systems, are useful and, for many, standard tools in the teaching of mathematics and other subjects. There are, however, many mathematics and statistics courses being taught without the use of such support. One reason is that within many programmes, a mathematical based course might only represent one single (elective) course out of many other courses, such as management or business programmes, for example.

Such programmes clearly do not warrant the full commitment and time required to develop and use a computer algebra system, which would of course also represent great expense. Nevertheless, such courses would still greatly benefit from some form of interactive support either via the web or other medium, but this may seem rather limited given that no computer algebra system is being exploited and the realistic assumption that many educators do not have the time and expertise to develop interactive applets.

There exists many examples of mathematics support on the web which range from the rather dull posting of lecture notes[2] to fully interactive web based mathematics that exploit either a plug-in or Java. Educators can of course refer their students to many of these sites but many teachers feel the urge to develop and use their own illustrations in order to preserve ownership and control over their teaching.

In this paper, we explicitly demonstrate that animation can be used as an interactive tool. Such an approach requires no sophisticated programming, no plug-in but clearly does require some form of software to create the animation (in this paper, Mathcad is used to create all animations). Once an animation has been constructed, we demonstrate that they can also be used as an interactive tool and hence used to mimic more sophisticated interactive applets, which require a lot more time and may require sophisticated programming to construct.

After briefly reviewing some web based and animation capabilities of some computer algebra systems (which can be used as tools by educators for constructing web pages and animation), we review the construction of animation to illustrate mathematical

---

[2]There still exists some technical problems for displaying mathematics on the web. The many GIF files created for even moderately used mathematics symbols will hopefully be superseded by something like MathML [1, 2]

concepts using a specific software, Mathcad. The specific software used in constructing the animations, however, is not important as long as users can control the animation via a control bar or button. Finally, it is argued that such animations can be used to mimic interactive applets, which are easy and fun to use.

2 Computer algebra systems

Before discussing the construction of animation, it might be useful to discuss some support and limitations of a few popular computer algebra systems [3, 4] with regard to the presentation of mathematics on the web and their capabilities for constructing animated illustrations.

The purpose of computer algebra systems is to perform both symbolic and numerical calculations. Many of them have evolved into very sophisticated computational systems, most of which can be programmed and are actively used as indispensable research tools in many diverse fields.

In recent years, many computer algebra systems have developed easy-to-use point-and-click user interfaces which greatly help to reduce the learning curve of such systems. Together with these developments has been an increase in the numbers of optional output formats such as RTF, Word, LaTeX, C, Fortran etc. Two output formats which are specific to web based presentation are html and MathML. Also, animation capabilities are also emerging as a standard tool as part of these packages.

Six popular computer algebra systems are listed below together with an indication of their web capabilities in terms of html and MathML conversion.

Computer algebra system	html	MathML	Reader	Animation
Derive	×	×	×	×
Maple	√	√	×	√
Mathcad	√	√	√	√
Mathematica	√	√	√	√
MATLAB	(Java)	(Java)	(ASCII)	√
Scientific WorkPlace	√	√	√	×

Experience as shown that some of the above conversion capabilities are far from ideal and very often one needs to go into the conversion codes to manually tidy things up. In any case, such conversions presently only generate non-interactive documents (MathML can of course be used as input and output for some computer algebra systems, which is hoped to become more popular in the future [1, 2]).

A step towards web based interactivity is the use of a free plug-in or reader that enables computer algebra files to be read. Unfortunately, some of these free readers do not perform calculations and one is therefore left with an option that is not much better than html and/or MathML conversion.

One such worthy plug-in is Mathcad's **Mathcad Explorer** which is a fully functioning web browser that also acts as a mathematics engine for Mathcad files. This represents a very cost effective and straightforward way of presenting interactive mathematics on the web. Mathcad Explorer can be downloaded from [5] and can even be placed on your own academic website.

Finally, a very useful capability of computer algebra systems is animation, which can be exported in various video or animated GIF and similar formats for direct use on the web, which are also indicated for the six popular computer algebra systems in the table above. Such animations are very useful for teaching and can illustrate specific concepts graphically, symbolically or numerically. The drawback, however is "what you see is what you get" and there is no scope to allow for full interactivity or to allow users to change parameters in the generated video or GIF files etc. (see below).

3 Animation

Animation can be a very useful tool for studying numeric output both directly and visually, as well as giving a visual illustration of symbolic manipulations and procedures. Animation can be created using some computer algebra systems (not all, see previous section) and other software packages. In this paper we use Mathcad[3] as a tool for creating animated illustrations but I would like to stress that other computer algebra systems and software packages may do just as well. The only requirement in

---

[3]Mathcad 2001 (Professional) was used to construct all animations in this presentation.

what follows is that the animation must be easily controlled via a control bar or button so that users can readily vary the direction of motion of the animation.

Apart from being a general purpose computer algebra system, Mathcad [5] is excellent for creating animated mathematics. You create animations in Mathcad using a variable called FRAME in your equations. Mathcad creates a video by shooting a sequence of snapshots or frames. One frame is generated, in order, for each value of FRAME in the specified range.

Suppose you wish to illustrate a piece of mathematics over which a parameter n is varied over some range. The Mathcad statement [5] for this is

$$n := \text{FRAME}$$

where FRAME represents a real number over some discrete specified range (the step size over which FRAME varies can also be controlled). You can now animate any region that depends on the variable FRAME. To do so [5]:

- Create a region whose appearance depends on FRAME.

- Make sure your worksheet is in automatic mode.

- Choose Animate from the View menu to open the Animate dialog box.

- Enclose the region(s) you want to animate in a selection rectangle.

- In the dialog box, specify starting and ending values of FRAME.

- Click "Animate."

You will see a miniature rendition of your selection inside the dialog box. Mathcad redraws this once for each value of FRAME. This will not necessarily match the playback speed since at this point, you are just creating the animation, you are not playing it back yet.

Note: When animating a plot, it is best to set the axis limits yourself instead of letting them be set automatically. If you do not do this, the axis limits get reset at every frame in the animation. The result: a movie in which the axis limits and not the plot change from one frame to the next.

As a simple example, suppose you wish to create an animation of the letter a varying in integer steps from 1 to 10. Within Mathcad's interface type

$$a := \text{FRAME}$$

at some suitable position. Now enter a again elsewhere, hit return and this will give $a = 0$, which is the default value of Frame. Now select Animate from the View menu and a dialogue box will appear to enter the initial and final values over which FRAME will run (in integer steps). You can also input the frames per second at which the final video animation will run.

Finally, select a region around $a = 0$, using the right mouse button, click Animate within the dialogue box, and the animated video will be generated showing a taking integer values from 1 to 10.

In a similar way, many mathematical concepts, both numerical and symbolic, can be similarly illustrated using FRAME as a parameter that varies over some chosen range. This same principle illustrated in the simple example can be straightforwardly applied to graphs (both two and three-dimensional graphs), numerical and symbolic solutions to equations, differentiation, integration, Taylor series, Fourier series, general expansions, etc. which involve a single parameter changing.

4   Animation as an interactive tool

The previous section briefly reviewed the basic animation construction procedure using Mathcad, which generates video animation. In what follows, however, animation constructed via any appropriate software can also be exploited as an interactive tool. The only requirement is that in viewing the animation, there is a control bar or button which allows users to vary the direction of animation and hence allows users to choose a specific value over which the animation parameter (for Mathcad: FRAME) is running.

Such a requirement is possible for video animation (constructed by any means) on a personal computer and may be possible for other animation formats such as animated GIF (I am assuming there exists animation viewers for animated GIF which allow users to appropriately control things?) and Flash etc. However, I stress again that I personally use Mathcad because of its ease of use in constructing video animations,

which is straightforward to master and also suitable for "on the spot" construction.

Examples

Below is a minimal list of illustrations presented during the conference. All the illustration, and many more, are permanently located at [6]. The examples indicated here are chosen to represent the usefulness of animation as an interactive tool throughout all levels of mathematics, which I hope may suit some of the reader's needs.

- The simple example of the parameter a varying from 1 to 10 is shown.

- Two and three-dimensional plots.

- Numerical and symbolic solutions to a class of equations.

- Differentiation of a class of functions.

- Integration of a class of functions.

- Series expansion of functions.

- Answers for students to a class of problems.

- Some specific teaching animations for self study.

5   Limitations

The above illustrations hopefully illustrate the power and easy in which animations can be constructed and used as an interactive teaching and learning tool. An important factor for illustrating this approach is that students do not need access to any computer algebra system (which is the case for many mathematics courses) and the ease at which educators can construct such illustrations. There are, however, a number of limitations in such an approach as a general teaching tool.

- Only problems that involve a single parameter changing can be illustrated.

One can construct examples where there is an illusion that more than one parameter is being varied but users can fundamentally only vary a single parameter. For example,

the straight line $y = mx + c$ cannot be illustrated where the users can vary **both** m and c **independently**.

- The constructed animation is fixed.

Sophisticated interactivity allows users to vary input data and hence allows users a great deal of control. Here, only a class of systems can be modelled which are related by varying a single parameter. For example, an animation illustrating the graphs of the Taylor series of, say, $\sin(x)$ as more terms are included is easily constructed, but a similar illustration for, say, $\cos(x)$ requires one to generate a new animation.

6 Conclusion

Is it hoped that the illustrations presented here have demonstrated the usefulness of animation as an interactive tool which can be easily exploited in many mathematics course. The animations are very straightforward to construct (using, for example, Mathcad) and allows users to vary a single parameter and observe the corresponding changes.

The real usefulness of the approach discussed here is the ease in which fairly sophisticated animations can be constructed and used as interactive illustrations for both teaching and independent study. Indeed, the animations can even be constructed "on the spot" which gives educators full control of their teaching even while actually teaching!

Finally, I should perhaps mention similar constructions using Geometers SketchPad [7] together with the SketchPad HTML converter, which creates webpages[4] for use with JavaSketchPad [8]. This represent another nice and relative straightforward way of putting fully interactive (Java) applets on the web (no plug-in required)[6, 9].

---

[4]The current version of this is built into Geometers SketchPad version 4.0 but sadly, many conversions are blocked (including graphs) which were active in the β-version.

References

[1] www.w3.org/Math/

[2] www.mathmlconference.org/

[3] Wester M. Computer Algebra Systems, A practical Guide, Wiley, 1999.

[4] Toon A. Applicable Mathematics with Computer Algebra, McGraw-Hill, 2001.

[5] www.mathsoft.com/mathcad/explorer/

[6] simfc1.sim.edu.sg

[7] www.keypress.com/sketchpad/

[8] www.keypress.com/sketchpad/java_gsp/index.html

[9] Toon, A. Geometric Construction and Elementary Analysis, Proceedings of the Sixth Asian Tecnology Conference in Mathematics, 187 - 198, 2001.