Creating online learning resources:
Games, applications and interactive worksheets

Ken Wessen

This article is based on the presentation I gave at the MANSW Annual Conference 2014.

A puzzle

I'd like to start with a puzzle that I read about recently (Nishiyama, 2014) and have subsequently presented as follows to some of my classes in order to encourage mathematical thinking.

> Suppose you were to place two 20c pieces on a table in front of you so that their edges were touching and one was directly above the other. Now imagine carefully rolling the upper coin around the coin below until it is directly underneath. How do you think the lower coin is now oriented – upside down, the right way up, some other orientation?

Give it a try – you may be surprised to find that although the upper coin has only rotated along half of its circumference, it has completed a full revolution!

One way to see why this should be true is to think about what point on the outer coin's edge will end up in contact with the inner coin. (From now on I will call the rolling coin the *outer* coin, and the stationary coin the *inner* coin.) It will be a point diametrically opposite the initially touching point, and thus the point at the top of the coin. But this point will now be touching the bottom of the inner coin, so the outer coin will again be oriented upwards. Alternatively, imagine the outer coin rolling along a straight line of length half the circumference. In this case it will definitely end upside down, but then imagine dragging the line into a semicircle of the same radius. This has the effect of restoring the upright orientation of the rolling coin.

After experimenting with some coins and considering the explanations, students can see that rolling the outer coin fully around the inner coin will result in two complete revolutions of the outer coin, and it's time to extend the problem.

> If the inner coin is twice the diameter of the outer coin, how many revolutions of the outer coin would result from the journey around the inner coin? What about some other size ratios?

For the first question our earlier experience argues against the seemingly obvious answer of 2, so maybe it will rotate 4 times? This puzzle sets up a beautiful and interesting context for mathematical thinking, but we can no longer simply use coins to experiment. To maintain the accessibility of the problem and the ability to explore, visualise and explain, we need a dynamic tool. I used the framework I have built to host online mathematical activities, *The Mathenaeum* (http://thewessens.net/maths), to quickly and easily construct a visualisation of the problem for exploration in class. This specific activity can be found at http://thewessens.net/rolling and appears as shown in figure 1 below.
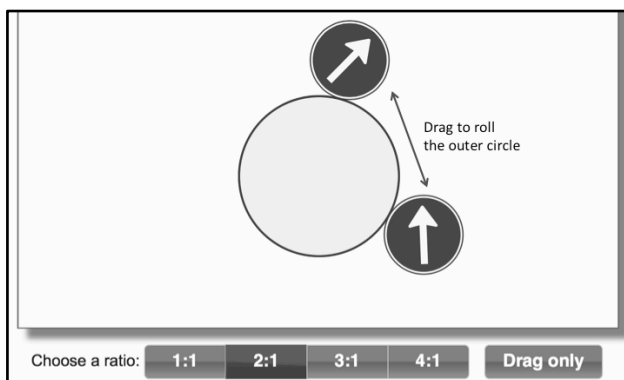

Figure 1: An activity at *The Mathenaeum* based on the coin rolling puzzle.

The activity presents the two circles in a chosen size ratio, and allows the outer circle to be rolled by mouse (or touch where available) around the inner circle while counting revolutions. Using this tool it is easily discovered that a 2:1 ratio produces 3 revolutions, a 3:1 ratio produces 4 revolutions, and a 4:1 ratio produces 5 revolutions. The pattern is clear, but where does the "extra" revolution come from? Selecting "Drag only" we can move the outer coin around the inner coin without rolling and demonstrate that this motion alone introduces an additional rotation. Eureka!

In summary, we have taken an interesting and accessible problem, explored and extended it via both physical and online manipulation. It is simple to state, hard to fathom, rewarding to understand, and is easily taken home to puzzle siblings and parents. Clearly problems such as this, coupled with appropriate tools, provide excellent opportunities for engagement and learning.

Specific requirements for resources

Online resources support the need to engage students through visualisation and interaction, provide opportunities for self-paced learning and revision, and are therefore extremely valuable. For these reasons there is an enormous and ever increasing amount of interactive material

available. Some particular sites I make regular use of are *Maths Starters* (http://mathsstarters.net) for quizzes, *Maths is Fun* (http://www.mathsisfun.com), *Cut the Knot* (http://www.cut-the-knot.org), and *Mathematics Open Reference* (http://www.mathopenref.com) for reference and interactivity. Additionally there is *Wolfram Alpha* (http://www.wolframalpha.com) for general mathematical enquiry and visualisation, *JSXGraph* (http://jsxgraph.uni-bayreuth.de/) and *GeoGebra* (http://www.geogebra.org) for geometry, statistics and more. However, my attempts to tailor and maintain something precisely matching my particular teaching needs were often frustrated, so I decided to develop my own set of activities.

Foremost in my mind were the specific requirements:

> *Portability* – the activities need to run effectively across as many different devices as practical, and once loaded, not require an ongoing Internet connection,
>
> *Consistency* – consistency of appearance and interaction means a student who is comfortable with one activity can easily feel comfortable with them all,
>
> *Accessibility* – the activities should be available to both teachers and students, at school and at home,
>
> *Customisability* – use technologies that allow quick and easy development and customisation to specific needs.

I concluded that the current web technologies (HTML5, Javascript, CSS) best allowed me to meet these requirements. They are well established, free, require only a text editor and browser as a development environment, and are cross platform through the support of all modern browsers – both desktop and mobile. Thus, in January this year, *The Mathenaeum* was born.

Building on the above practical requirements, I next needed to formalise the pedagogical requirements. I decided on an explicit set of *Guiding Principles* as follows:

> Interactive visualisation aids understanding and conceptual development.
>
> Dynamic and exploratory activities generate engagement.
>
> Historical background adds interest and helps provide context and connection for the mathematical material.
>
> Good mathematical activities promote group interaction.
>
> Good mathematical activities naturally lead to discussion and extension.

Focussing on any potential activity or learning objective through the lens of these principles helps maintain consistency of purpose and coherence in presentation.

Nature of the activities

The activities available at The Mathenaeum can be classified according to four broad types:

> *Guided activities* – directed work involving particular concepts or processes,
>
> *Explorations* – interact and discover,
>
> *Games* – practice through competition,
>
> *Visualisation* – view and understand.

Additionally, I have included a number of associated general utility applications – flipping coins (http://thewessens.net/coin), rolling dice (http://thewessens.net/dice), choosing from a list (http://thewessens.net/choose) for selecting student "volunteers" etc.

Individual activities are organised under the topics: *Number, Geometry, Algebra, Chance + Data,* and *Utilities* as just mentioned.

Some Examples

Since starting in January the site has grown to 25 pages, with something new now being added each week or two. The four examples described below illustrate the nature and direction of the activities at the site.

The *Equation Solver* algebra activity is an example of one of the guided activities, and is shown in Figure 2. A student can generate equations and inequations with specific properties or enter specific examples, and then work through their solution. Errors are caught and prevented, and a correct process reinforced.
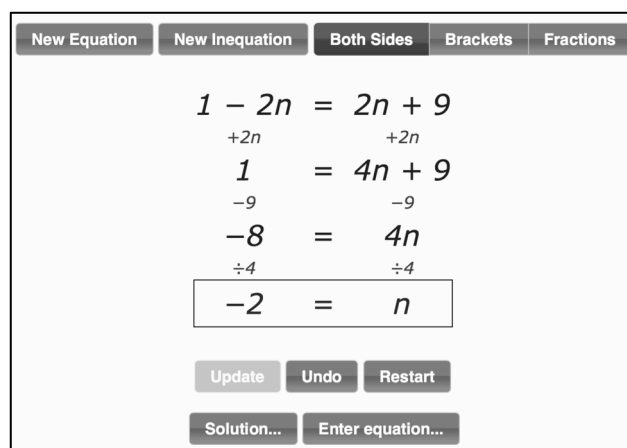

Figure 2: The Equation Solver activity.

Another of the algebra activities, and one that has proven to be very popular, is the *Sphinx* activity shown in figure 3. Here students attempt to make

larger and larger replicas of a basic "sphinx" shape, made from 6 equilateral triangles, and then study their properties. This activity can begin with paper and scissors, and the online version then provides an easy way to demonstrate solutions or extend to larger sizes.
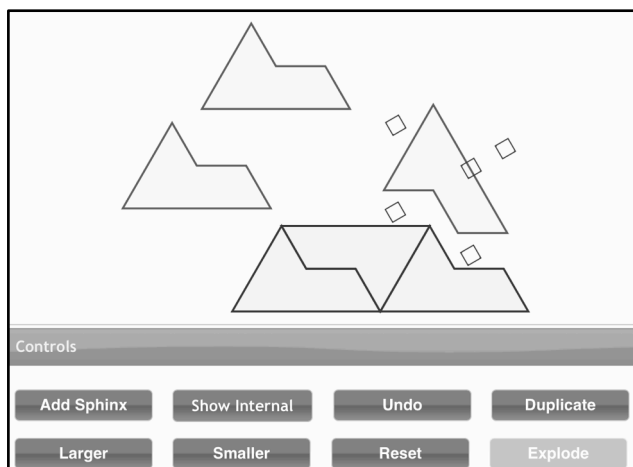

Figure 3: The Sphinx replication activity.

Properties of quadrilaterals can be explored using the *Quadrilaterals* activity shown in figure 4. By dragging vertices, students can explore standard quadrilaterals, as well as change between types. Animated folding and rotation allow discovery and visualisation of symmetries.
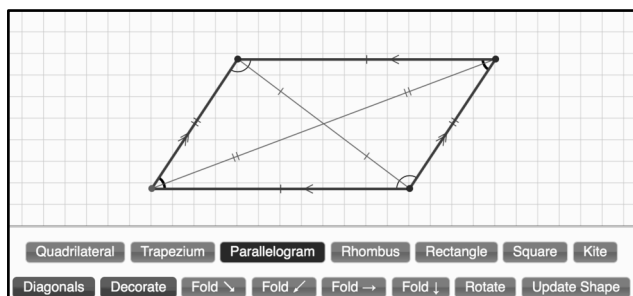

Figure 4: Explore quadrilateral side, angle and symmetry properties

As a final example, one of the games provided is *Factor Champion* (see figure 5) – a competitive factorisation activity that can be played between two players, or one player against the computer. To play successfully, a student needs to recognise possible factors quickly, and choose the pair that maximises their score at the expense of their opponent. Options exist to add time pressure and larger numbers. The factor trees generated can be examined between turns, and primality plays an important role in the optimal strategy.
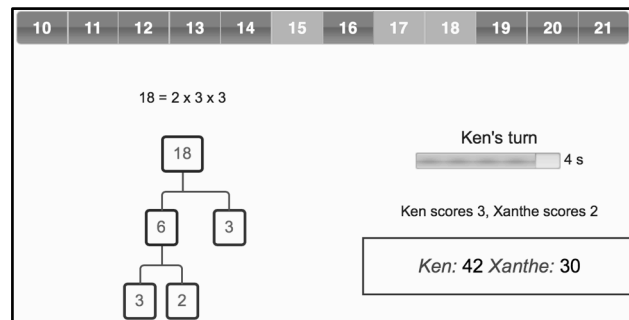

Figure 5: One turn of *Factor Champion* showing the factor tree generated during game play for a starting value of 18.

Outcomes

I have found that the structures and processes built into the site promote easy development and extension as I had hoped, leading to a rapid and satisfying growth of the site. Student engagement with the activities has been strong, and I am happy with the clarity of presentation the activities have enabled me to achieve in my teaching. Using the site has been relatively straightforward on school computers, iPads and even to a limited degree on smart phones, and it is pleasing for the material to be available at home for students who wish to investigate further.

Nevertheless, limited access to equipment in the classroom reduces the effectiveness – sometimes forcing me to make more use of the activities as presentation tools or with only selected individual involvement, rather than the full class inclusivity that I would prefer. Naturally competitive students tend to dominate the games, and this can be inhibiting for quieter or less confident students at first.

The prevalence of old and incompatible versions of Internet Explorer is a serious impediment. To make everything function on these old versions would require substantial additional work that is beyond what I am able to commit to the project. Later versions of Internet Explorer are fully supported, as are all the major browsers – Chrome, Firefox, Safari and Opera. On the other hand, the additional work required in making the activities work appropriately on both touch and desktop devices is very worthwhile, and is an important aspect of ongoing activity development.

You can do it too!

I certainly hope that *The Mathenœum* becomes a significant and widely used website supporting fun and engaging mathematical learning, and supplementing the pool of available resources with activities reflecting its unique perspective. The recently added online contact form makes it easy for students and other teachers to suggest additions and extensions that I can incorporate over time. I am also open to contributions of

activities from others, and can assist their development within the overall framework. But more generally, I hope The Mathenaeum serves as an example of the advantages and value of personal programming in teaching and learning. To explore this last point, let's see what's possible in just a few lines of code.

Returning to the original puzzle and considering the curve traced out by a point on the outer circle leads to the *Epicycloid* curves, which themselves can open up interesting explorations of locus, polar coordinates, cyclic curves and even Ptolemaic astronomy and epicycles. I will next show how we can use JSXGraph to make a simple visualisation of these curves.

First we need an HTML template – this is just a plain text file (with extension .html) that we can open in any text editor, and has the necessary skeleton HTML to define the page and load JSXGraph. The code in the appendix to this paper can be used as a general template for this kind of visualisation, with each new page only needing different content in the `<script>` section from lines 17 to 50. While keeping the file open in an editor, you can preview the output in a web browser, refreshing to view the results of each change in the file. Changing the links in the `<head>` section (lines 5–8) to local copies of the jsxgraphcore.js and jsxgraph.css files (these can be downloaded from the JSXGraph website) removes the need for an active Internet connection as well.

In the `<body>` section we define the display box and its size in a `<div>` with id 'jxgbox' (line 14), and then fill in the `<script>` section with the necessary JSXGraph code (starting from line 16).

First we define the board and its boundaries, the variable r for the outer radius, and k for the ratio.

```
board = JXG.JSXGraph.initBoard('jxgbox',
        {boundingbox:[-13,13,13,-13]});
var k = 3, r = 1;
```

To make a visualisation of the locus from the rotating circles problem we will need a few geometrical elements. We need three points: `c_R` for the center of the inner circle, `c_r` for the center of the outer circle, and `p` for the point of contact that we wish to trace for our locus (lines 32–36).

```
c_R = board.createElement('point',[0,0]…
p = board.createElement('point',
    [function () { return k * r; }, 0],…
c_r = board.createElement('point',
    [function () { return (k + 1) * r; },0],…
```

The inline functions tell JSXGraph that the values can change and need to be recalculated each update. Using these three points we construct the inner and outer circles (colouring

one green and the other red):

```
board.createElement('circle',[c_R,p],
    {strokeColor:'green'});
board.createElement('circle',[c_r,p],
    {strokeColor:'red'});
```

The equation of the curve generated by the rolling, parameterised by the rolled angle θ, is given by

$$x(\theta) = r(k+1)\cos\theta - r\cos((k+1)\theta)$$
$$y(\theta) = r(k+1)\sin\theta - r\sin((k+1)\theta)$$

and this is directly implemented in our script using a 'curve' element as follows:

```
board.create('curve', [
    function(t) { return r * (k + 1) *
Math.cos(t) - r * Math.cos((k + 1) * t);},
    function(t) { return r * (k + 1) *
Math.sin(t) - r * Math.sin((k + 1) * t);},
    0,8 * Math.PI],{strokeWidth:2});
```

The first 2 values in the array give the x and y coordinates, the last 2 give the initial and final values of the parameter t (0 to 8π).

A slider is a useful control to update the ratio k. We define and add a slider (line 44) using

```
s = board.create('slider', …
```

with additional parameters defining its size, position, range and the step size. Finally we need to update the board as the slider is dragged by providing a function that updates k and redraws the graphical elements accordingly.

```
s.on('drag', function() {
    k = s.Value();
    board.update();
});
```

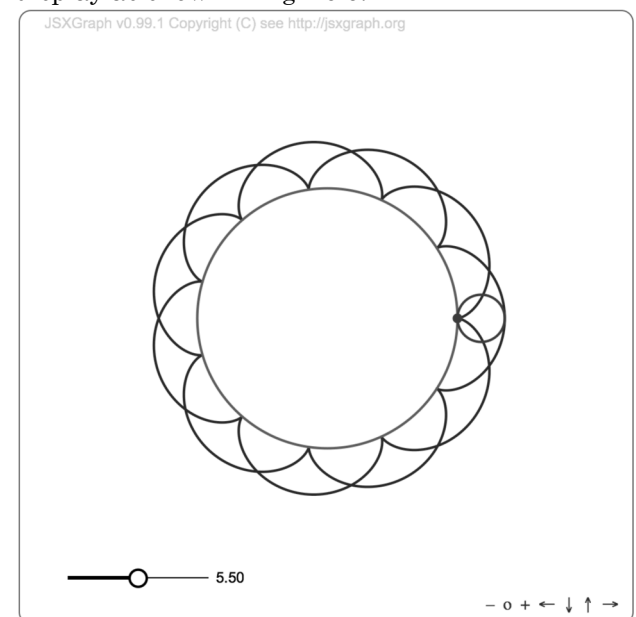Loading and running in a browser produces a display as shown in figure 6.

Figure 6: The web page coded using JSXGraph displaying a 5.5:1 ratio epicycloid.

References

Y. Nishiyama, *Circles rolling on circles*, Plus magazine, http://plus.maths.org/content/circles-rolling-circles, University of Cambridge, May 2014, retrieved Sep 23, 2014.

Black Douglas Professional Education Services, Mathematics Task Centre, http://mathematicscentre.com/taskcentre/sphinx.htm, retrieved Sep 23, 2014.

Appendix

Here is a complete listing of the HTML and javascript for visualising the epicyloid curves as discussed in the text. The line numbers are for reference only and are not part of the code. This can be downloaded from http://thewessens.net/simple_epicycle.html, and a link to a more advanced version may be found in the "About this Activity" section of the activity page http://thewessens.net/rolling.

```
1   <!DOCTYPE html>
2   <html>
3   <head>
4     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
5     <link rel='stylesheet' type='text/css'
6       href='http://jsxgraph.uni-bayreuth.de/distrib/jsxgraph.css' />
7     <script type='text/javascript'
8       src='http://jsxgraph.uni-bayreuth.de/distrib/jsxgraphcore.js'>
10    </script>
11  </head>
12
13  <body>
14    <div id='jxgbox' class='jxgbox' style='width:500px; height:500px;'></div>
15
16    <script type='text/javascript'>
17      board = JXG.JSXGraph.initBoard('jxgbox', {boundingbox:[-13,13,13,-13]});
18
19      // k is the ratio of the outer circle radius to the inner circle radius
20      //  and the inner radius, r, is fixed at 1
21      var k = 3, r = 1;
22
23      // We define the epicyloid curve parametrically, with angle parameter varying from 0 to 8 Pi
24      board.create('curve', [
25        function(t) { return r * (k + 1) * Math.cos(t) - r * Math.cos((k + 1) * t);},
26        function(t) { return r * (k + 1) * Math.sin(t) - r * Math.sin((k + 1) * t);},
27        0, 8 * Math.PI], {strokeWidth:2}
28      );
29
30      // We need three points – the inner and outer circle centres, and the point of contact
31      // Only the contact point is visible
32      var c_R = board.createElement('point',[0,0], {visible: false});
33      var c_r = board.createElement('point',[function () { return (k + 1) * r; },0],
34          {visible: false});
35      var p = board.createElement('point',[function () { return k * r; }, 0],
36          {name:'', size: 2, face: 'o'});
37
38      // Now we can use these points to define our two circles
39      board.createElement('circle', [c_R, p], {strokeColor:'green'});
40      board.createElement('circle', [c_r, p], {strokeColor:'red'});
41
42      // Finally we need a slider control to adjust the k ratio parameter
43      // We will support values are from 1 to 10 in steps of 0.25
44      var s = board.create('slider', [[-11, -11], [-5, -11], [1, 3, 10]], {snapWidth: 0.25});
45      s.on('drag', function() {
46          // as the slider is dragged, update the ratio parameter, k, and redraw the board
47          k = s.Value();
48          board.updateQuality = board.BOARD_QUALITY_HIGH;
49          board.update();
50      });
51    </script>
52  </body>
53  </html>
```