

# Computer Science NEA 2025

Nathan Tatkowski



INVICTA GRAMMAR  
SCHOOL

Invicta Grammar School

Centre Number:

Candidate Number:

## Contents

<b>1</b>	<b>Analysis</b>	<b>3</b>
1.1	Problem Identification . . . . .	3
1.2	Identification of why this problem is solvable by computational methods . . . . .	3
1.3	Description of the Current System . . . . .	3
1.4	Identification of Stakeholders . . . . .	3
1.5	Identification of User Needs and Acceptable Limitations . . . . .	4
1.6	Existing Solutions . . . . .	4
1.6.1	Analytic Methods . . . . .	4
1.6.2	Desmos . . . . .	5
1.6.3	Kerbal Space Program . . . . .	6
1.7	Success Criteria of the Proposed System . . . . .	7
1.8	Data Source(s) . . . . .	7
1.9	Volumetrics - Data Volumes . . . . .	7
1.10	Analysis Data Dictionary . . . . .	7
1.11	Data Flow Diagrams for Existing and Proposed System . . . . .	7
1.12	Justification of Chosen Solution . . . . .	7
1.13	Hardware and Software . . . . .	7
1.14	Entity-Relationship Models . . . . .	7
1.15	Identification of Objects and Object Analysis Diagrams . . . . .	7
<b>2</b>	<b>Design</b>	<b>7</b>
2.1	Overall System Design . . . . .	7
2.2	Description of Modular Structure of System . . . . .	7
2.3	Definition of Data Requirements . . . . .	7
2.4	Identification of Appropriate Storage Media . . . . .	7
2.5	Identification of Processes and Suitable Algorithms for Data Transformation and Completion of the Solution . . . . .	7
2.6	Sample of Algorithms . . . . .	7
2.7	User Interface Design Rationale and Usability Features . . . . .	7
2.8	Security and Integrity of Data . . . . .	7
2.9	System Security (Access Control) . . . . .	7
2.10	Overall Test Strategy . . . . .	7
<b>3</b>	<b>Implementation</b>	<b>7</b>
3.1	Annotated Listing of the Program(s) . . . . .	7
3.2	Annotated "Design Views" showing details of application-generated forms, reports, queries, buttons, cross tabulations, etc. . . . .	7
3.3	Procedure and Variable List . . . . .	7
3.4	Testing to inform development - Testing at each stage . . . . .	7
3.5	Re-Testing . . . . .	7
<b>4</b>	<b>Testing</b>	<b>7</b>
4.1	Test Plan . . . . .	7
4.2	Test Data . . . . .	7
4.3	Areas to Test . . . . .	7
4.4	Tables . . . . .	7
4.5	Justification of Data Selection . . . . .	7
4.6	Evidence of Testing . . . . .	7
4.7	Program Listing . . . . .	7

<b>5</b>	<b>Evaluation</b>	<b>7</b>
5.1	Comparison of Performance of your System against Success Criteria . . . . .	7
5.2	Analysis of User Feedback . . . . .	7
5.3	Evaluation of Usability Features . . . . .	7
5.4	Maintenance Issues and Limitations of the Product . . . . .	7
5.5	Improvements and Possible Extensions . . . . .	7

# 1 Analysis

## 1.1 Problem Identification

Often times during research it is important and very helpful to be able to visualise the events that are being analysed. When working with models in two dimensions, it is easy enough to be able to draw out an accurate diagram, even if it may be tedious. However, 2D models are nowhere near as applicable or useful as considering events in three dimensions, stemming from the fact that the world we live in is three-dimensional. Something that would aid intuition and help in problem-solving would be a way to have events modelled quickly, accurately, and clearly, given a set of initial conditions.

## 1.2 Identification of why this problem is solvable by computational methods

The key requirements stated above (accuracy, haste, and clarity) lend themselves very well to using computational methods. Computers are able to make calculations orders of magnitude faster than by hand or by analogue machine, and to a virtually arbitrary degree of accuracy. Many modern central processing units (CPUs) are also able capable of making use of concurrent processing, further increasing the advantage that a computer would have over a human. Graphical processing units (GPUs) are specifically designed for parallel processing, making them especially useful for graphics, which would allow for high quality renders for the user to be able to see. Any data that you would need to consider can be displayed in a clear and user-friendly fashion, making it highly customisable to fit the individual persons needs and for many attributes to be studied at the same time.

## 1.3 Description of the Current System

Without using computer simulations, the usual process is to produce a handful of equations by hand that would model the attributes of an object, commonly the path it takes in three-dimensional space. This has the benefit of giving exact values and equations that are very useful when trying to understand the underlying reasons for an event happening. For example when considering a pendulum, it is clear from the equation

$$T = 2\pi\sqrt{\frac{L}{g}}$$

that the period of the pendulum  $T$  does not depend on the mass of the object doing the swinging. However, when running computer simulations, such relationships may not be as obvious, and as computers aren't able to analytically solve problems (i.e. through the use of rigorous mathematics), this is a drawback that I will have to consider.

## 1.4 Identification of Stakeholders

After considering the problem I identified the following groups that could use a solution to this problem, as well as having useful insight on how a program like this should function.

- **University Students** often have to deal with complex systems and a way of visualising them would be very beneficial. I have been able to contact a student at the University of Aberdeen doing a masters in electrical and mechanical engineering. Their name is Hugo, and they are 21-years-old.
- **A-Level Students**, specifically students taking physics, would be able to greatly further their understanding of core concepts and be able to explore new ideas on their own. I have been able to communicate with Daniel, a year 12 physics student, about being a stakeholder for this project.

- **Teachers** of A-Level and below could make great use of simulation software in order to make learning much easier with models and demonstrations that are clear and easy to understand. I have been able to contact Mr Waters about being a stakeholder for this project, who teaches physics at Invicta Grammar School.

## 1.5 Identification of User Needs and Acceptable Limitations

Summary of key take aways from interviews:

- My stakeholders are people who generally enjoy doing physics and find it enjoyable, although they all acknowledge how much work it can be. As such, decreasing workload without completely eliminating need for human input would be important, as that would make it less enjoyable.
- People struggle with abstract concepts, and a common topic seems to be electricity and electricals systems, as well as visualising some key concepts in physics such as waves. Considering a way to visualise circuits and the physics going on in those would be useful all of my stakeholders.
- Two out of three of my stakeholders said that they enjoyed astrophysics, so considering some sort of astro-mechanics simulation could be of use to them, as visualising large bodies moving in space is difficult.
- People find graphs very useful in visualisation and aiding intuition. Some sort of real-time graphing of attributes could be something useful to consider in the final product.
- All my stakeholders are competent in using simulation software, or don't mind spending time to learn how to use one properly. This would mean that accuracy and functionality could be prioritised over general user experience if necessary.

## 1.6 Existing Solutions

### 1.6.1 Analytic Methods

Analytic methods are very common as they require little cost or set-up and their effectiveness only depends on how well you understand the physics that you are doing. Since my stakeholders enjoy doing physics and are also quite good at it, they are all well versed in spending time going through calculations in order to achieve a set of mathematical equations that describe the system being modelled.

Handwritten mathematical derivations for a physics problem involving two masses and a spring. The equations are as follows:

$$\begin{aligned} \textcircled{1} \quad & \frac{d^2 r}{dt^2} + \frac{R_2}{r(r-R_2)} \frac{dr}{dt} \frac{dr}{dt} = 0 \\ \textcircled{2} \quad & \frac{d^2 r}{dt^2} + \frac{R_2}{r(r-R_2)} \left( \frac{dr}{dt} \right)^2 - \frac{R_2}{r(r-R_2)} \left( \frac{dr}{dt} \right)^2 + \frac{R_2}{r(r-R_2)} \left( \frac{dr}{dt} \right)^2 = 0 \\ & \text{or a radial velocity, so } d^2 r/dt^2 = 0 \text{ the second derivative is zero} \\ \textcircled{3} \quad & \frac{d^2 r}{dt^2} + \frac{R_2}{r(r-R_2)} \left( \frac{dr}{dt} \right)^2 - \frac{R_2}{r(r-R_2)} \left( \frac{dr}{dt} \right)^2 = 0 \\ \textcircled{4} \quad & E = \frac{1}{2} m \left( \frac{dr}{dt} \right)^2 + \frac{1}{2} k (r - r_0)^2 \\ & \quad + R_2 \ln \left| \frac{(r - R_2)^2 + \sqrt{(r - R_2)^2 - R_2 r}}{(r - R_2)^2 - \sqrt{(r - R_2)^2 - R_2 r}} \right| + K \\ \textcircled{5} \quad & K = \frac{\pi (r - R_2)^2}{2 R_2} \quad \left( \begin{array}{l} E = \text{total energy} \\ r = \text{distance from the centre} \end{array} \right) \\ \textcircled{6} \quad & \frac{dr}{dt} = \pm \sqrt{\frac{R_2}{r - R_2}} \left( \frac{r - R_2}{r} \right) \\ \textcircled{7} \quad & \frac{dr}{dt} = \pm \sqrt{\frac{R_2}{r - R_2}} \left( \frac{r - R_2}{r} \right) \quad \left( \begin{array}{l} R_2 = \text{Hooke's parameter (constant)} \\ R_2 = \text{the distance from the centre} \end{array} \right) \\ \textcircled{8} \quad & \frac{dr}{dt} = \pm \sqrt{\frac{R_2}{r - R_2}} \left( \frac{r - R_2}{r} \right) \quad \left( \begin{array}{l} R_2 = \text{the radial distance from the centre} \\ R_2 = \text{the distance from the centre} \end{array} \right) \end{aligned}$$

Figure 1: A sample of Daniel's working out

**Advantages:**

- Very versatile. Paper and pen allows for great customisation in the layout of the work including diagrams and annotations, which allows the user to be able to do things the way they want to do.
- Cheap and easy to use. Little equipment is required and there isn't a need to install anything.
- Writing things physically on paper usually results in the writer being able to remember it easier, which would help in remembering things when learning.
- To get effective with analytic methods you require a lot of practice, which develops core skills like manipulation of various equations to achieve a desired result.
- Ability to get exact answers and relationships between objects and attributes. Computers can only approximate exact answers and being able to see the equations of what is happening can greatly aid intuition when tackling future problems.
- Doing hard work to get to a result is rewarding and relaxing. Offloading a lot of that work to a computer would reduce the enjoyment from this process.

**Disadvantages:**

- Difficult to organise. You can't save paper notes in a way that is easily shareable and easy to organise on a computer (other than scanning them as pdf files which can take a lot of time and my clients find annoying). It may also be hard to stay consistent with formatting due to how versatile it is, as seen in Figure 1.
- Getting accurate graphs beyond simple sketches is difficult. Performing numerical methods by hand is very time-consuming which decreases the amount of time that can be spent on doing actual work.
- Scope of problems that can be approached is limited. Not all systems can be solved using analytic methods and sometimes numerical methods are necessary, depending on how simple or complex you choose to make your model.
- Some problems are much harder to approach as the effectiveness of analytic methods is dependant on your ability to manipulate and work with equations, as well as general mathematical ability. Humans also make mistakes and are far less consistent than a computer at doing the same or similar calculations repeatedly.

**1.6.2 Desmos**

In Figure 2 below, Desmos has been used in order to see how attributes of an object change (specifically the  $x$  position and velocity with respect to time of an object experiencing drag). Desmos is great in order to aid intuition and view general trends, however you still need to work through equations in order to have something to graph, and Desmos won't give exact values for solutions, but the final visualisations are still very useful.

The simplicity of Desmos allows it to be very versatile, but that also means it's more of a jack-of-all-trades and master of none, so it wouldn't be very good for a more specialised application (like the ones being considered).

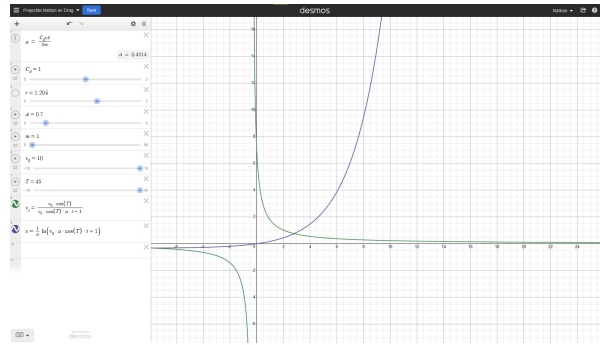


Figure 2: Desmos graphing software

**Advantages:**

- Desmos is a very versatile software which means that it can be used for a wide variety of problems.
- Free and easy to use. Desmos can be accessed by anyone with a computer or mobile device by going to their website, or by downloading the mobile app.
- Very customisable. Desmos allows for labels within the equation space, as well as changing the names of the labels on the axes. Which allows for some form of documentation within the graphs. The colours and
- Easy to share and work on on multiple devices. After you make an account, you are able to save graphs that you've done and access them on whatever device you're signed in on. Graphs can also be shared with links once they're saved which allows for collaboration on projects.
- As of quite recently, Desmos has released a beta for a 3D graphing mode, which is great for modelling some events happening in 3D, such as the trajectory of an object given some initial conditions.

**Disadvantages:**

- The versatility of Desmos means that in the case of a more specialised system with specific needs, it would be more beneficial to create custom software for that specific system. However Desmos still manages to cover a wide variety of problems.
- The accuracy of Desmos is not very customisable, usually giving answers to two or three decimal places, and it is only able to give exact answers in some circumstances (values are usually given in terms of  $\pi$  when working with trigonometric functions).

**1.6.3 Kerbal Space Program**

Kerbal Space Program here

## 1.7 Success Criteria of the Proposed System

## 1.8 Data Source(s)

## 1.9 Volumetrics - Data Volumes

## 1.10 Analysis Data Dictionary

## 1.11 Data Flow Diagrams for Existing and Proposed System

## 1.12 Justification of Chosen Solution

## 1.13 Hardware and Software

## 1.14 Entity-Relationship Models

## 1.15 Identification of Objects and Object Analysis Diagrams

# 2 Design

## 2.1 Overall System Design

## 2.2 Description of Modular Structure of System

## 2.3 Definition of Data Requirements

## 2.4 Identification of Appropriate Storage Media

## 2.5 Identification of Processes and Suitable Algorithms for Data Transformation and Completion of the Solution

## 2.6 Sample of Algorithms

## 2.7 User Interface Design Rationale and Usability Features

## 2.8 Security and Integrity of Data

## 2.9 System Security (Access Control)

## 2.10 Overall Test Strategy

# 3 Implementation

## 3.1 Annotated Listing of the Program(s)

## 3.2 Annotated "Design Views" showing details of application-generated forms, reports, queries, buttons, cross tabulations, etc.

## 3.3 Procedure and Variable List

## 3.4 Testing to inform development - Testing at each stage

## 3.5 Re-Testing

# 4 Testing

## 4.1 Test Plan

## 4.2 Test Data

## 4.3 Areas to Test

## 4.4 Tables

## 4.5 Justification of Data Selection

## 4.6 Evidence of Testing

## 4.7 Program Listing