Investigation Into The Representation Of 4D Shapes

13/01/2022

Christmas Progress Report

- Rotors Work besides 1 thing…
- Grab ball was unsuccessful (was not intuitive) so will leave that alone
  - Although I could use it specifically for 3D and not 4D, but still may not make sense
- Recorded tutorial video and built in-game video player
  - needs minor improvements
- Reorganised project alot (refactoring and directory structure)
- Created controllers
  - Load the different scenes
  - Setup random tests
  - Record data from test
    - Time, selected/loaded shape, texture, selected/loaded rotation etc.
  - Record survey data between tests
    - Confidence and understanding
- Attempted but failed to obtain angle between rotors
  - unsure why what i have tried didn't work

Questions

- Ideas of what could be wrong with Rotors (again)
- Opinion on graphs in between each representation
- Angle between Bivectors?
  - `Rotor.Scalar = Cos(theta/2)`
  - To “Add” rotations -> `Rotor4 total *= r`
  - To “subtract” rotations -> Rotor division = `Rotor4 total = total * r.Reverse()`
  - Taking the `Acos()` of the scalar of this does not yield the correct value…
  - Tried obtaining the *Dual* of a bivector (its normal vector), but cannot get that using GAlgebra in Vector form
  - With the normal Vectors I could use the dot product to get the angle between them

Plan Ahead

Week 1 & Week 2
- Finish implementing test system
  - graphs between representations
  - fix rotor rotation
  - add time limit

Week 3
- Begin developing data analysis tools, most likely with a jupyter notebook
- run some preliminary tests to decide if there is any data or visualisations I want to cut

Week 4 & Week 5 & Week 6
- Run experiments
- Begin dissertation
- Further develop data analysis tools

Week 7
- Evaluation of experiments

Week 8 & Week 9 & Week 10
- Write up draft dissertation

Am I on schedule

- Problem with Rotor may interrupt things
- Problem with angle between rotors may slow down things, but it is something that is not required for experiments
- Look to be on schedule