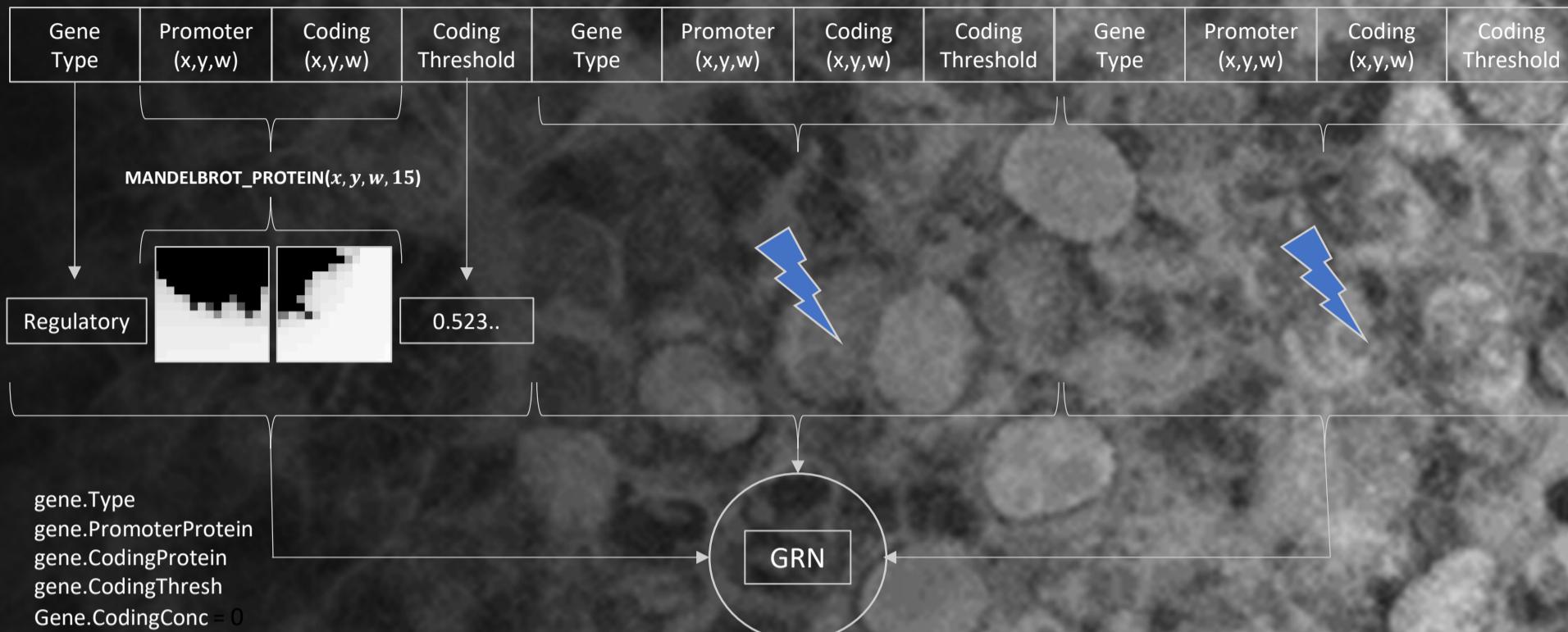


Constructing the GRN from the genetic code (Genotype-Phenotype Mapping)

Here we visualize the procedure through which a GRN is initialized using a sequence of real values representing the genetic code



Pseudocode

Mandelbrot Protein

To generate our digital proteins, we sample a square region from the Mandelbrot set, using 3 coordinates. This is done at a fixed resolution, so that the output is a 15x15 256-bitmap:

Algorithm 1: Mandelbrot Protein

```

procedure MANDELBROT_ITERATION(c)
    n ← 0
    z ← 0 + 0i
    while |z| < threshold && n < max_iteration
        z ← z2 + c
        n ← n + 1
    end while
    return n
end procedure

procedure MANDELBROT_PROTEIN(x, y, w, n)
    protein = IntegerArray(n, n)
    grid = DiscreteMesh(center = (x, y), step_size =  $\frac{w}{n}$ )
    for i in protein.Size.X
        for j in protein.Size.Y
            c = Complex( grid[i, j] )
            protein[i, j] = MANDELBROT_ITERATION(c)
    end procedure
    return protein
end procedure

```

Algorithm 2: Sense Environment

```

procedure SENSE_ENVIRONMENT(GRN, signal)
    for gene in GRN if gene.Type == Environmental
        gene.CodingConc ← signal
    end
end procedure

```



Protein Chemistry

Here we define the procedure through which two or more proteins 'react' and create another protein product

Algorithm 3: Fractal Chemistry

```

procedure FRACTAL_CHEMISTRY(ListOfProteins,ListOfConcentrations)
    protein = IntegerArray(n, n)
    protein_concentration ← 0
    for i in protein.Size.X
        for j in protein.Size.Y
            winner ← Max(reactant[i, j] for reactant in ListOfProteins)
            winner_conc ← GetConcentration(winner, ListOfConcentrations)
            protein[i, j] ← winner
            protein_concentration ← protein_concentration + winner_conc
        end
    end
    protein_concentration ← Normalize(protein_concentration)
    return protein, protein_concentration
end procedure

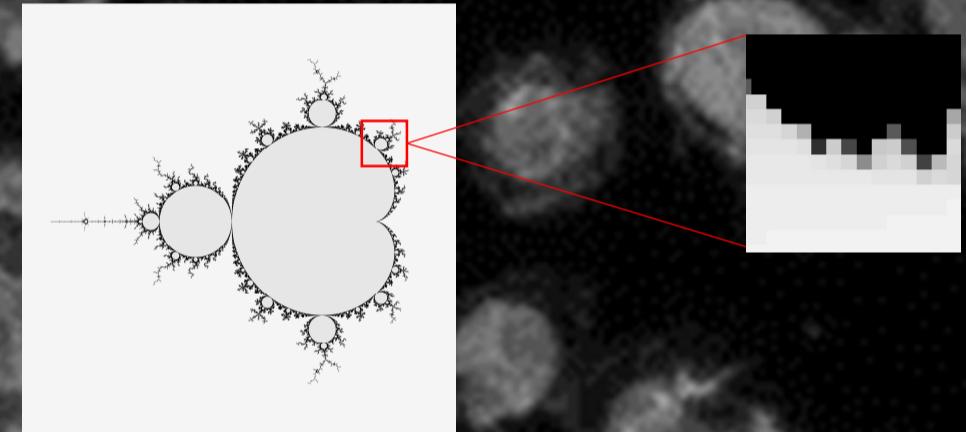
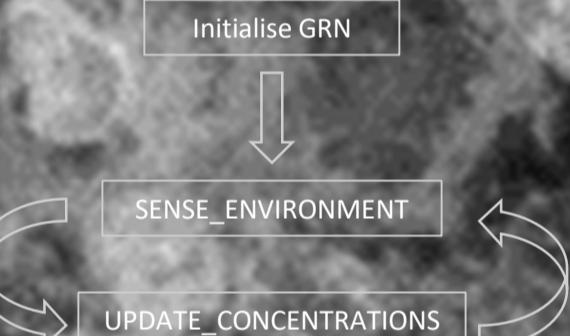
```

Algorithm 4: Protein Affinity

```

procedure PROTEIN_AFFINITY(protein_1, protein_2)
    affinity ← SUM(ElementWiseAbsoluteDifference(protein_1, protein_2))
    return affinity
end procedure

```



GRN Dynamics

Here we define how protein concentrations are updated using our gene regulatory network

Algorithm 5: Update GRN Concentrations

```

procedure CONC_CHANGE(gene, productProtein, productConc)
    affinity ← PROTEIN_AFFINITY(gene.PromoterProtein, productProtein)
    reactionStrength ← (1-affinity)*productConc
    dc ← reactionStrength*Sigmoid(reactionStrength) - gene.CodingConc/Cp
    return dc
end procedure

procedure UPDATE_CONCENTRATIONS(GRN)
    Cytoplasm = ListOfProteins
    for gene in GRN if gene.Type == Regulatory, Environmental
        if gene.CodingConc > gene.CodingThresh
            CytoplasmProteins.Add(gene.CodingProtein)
    end
    productProtein, productConc ← FRACTAL_CHEMISTRY(CytoplasmProteins)
    for gene in GRN if gene.Type == Regulatory, Behavioral
        gene.CodingConc ← CONC_CHANGE(gene, productProtein, productConc)
    end
end procedure

```

Project theme: Interpretation of Environment Through Fractal Gene Regulatory Networks (Real Time Data Feedback).

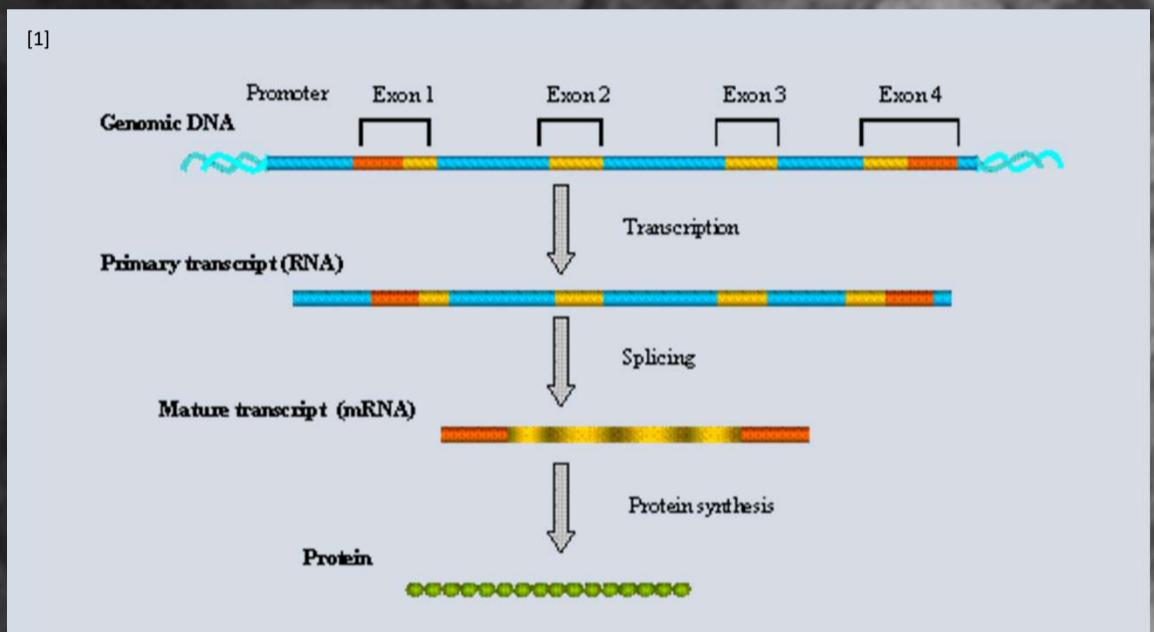
Proteins represent the language of cells. They are instrumental in the development process of all organism's observed in the biological world, including the remarkable transformation of the single-celled zygote to the fully developed human.

If proteins are words, then gene-regulatory networks are vocal cords. They are responsible for the control of the flow of certain proteins, using signals from the cell's environment.

Gene expression

Genes refer to specific regions of DNA that code for a specific protein. 'Gene expression' refers to the process in which specialist cell machinery interprets the DNA instruction set of a gene and produces the corresponding protein.

Sometimes the proteins produced directly contribute to the level of expression of other genes. It is through this type of feedback loop that gene expression is regulated, and it is a collection of such mechanisms which forms a Gene Regulatory Network.

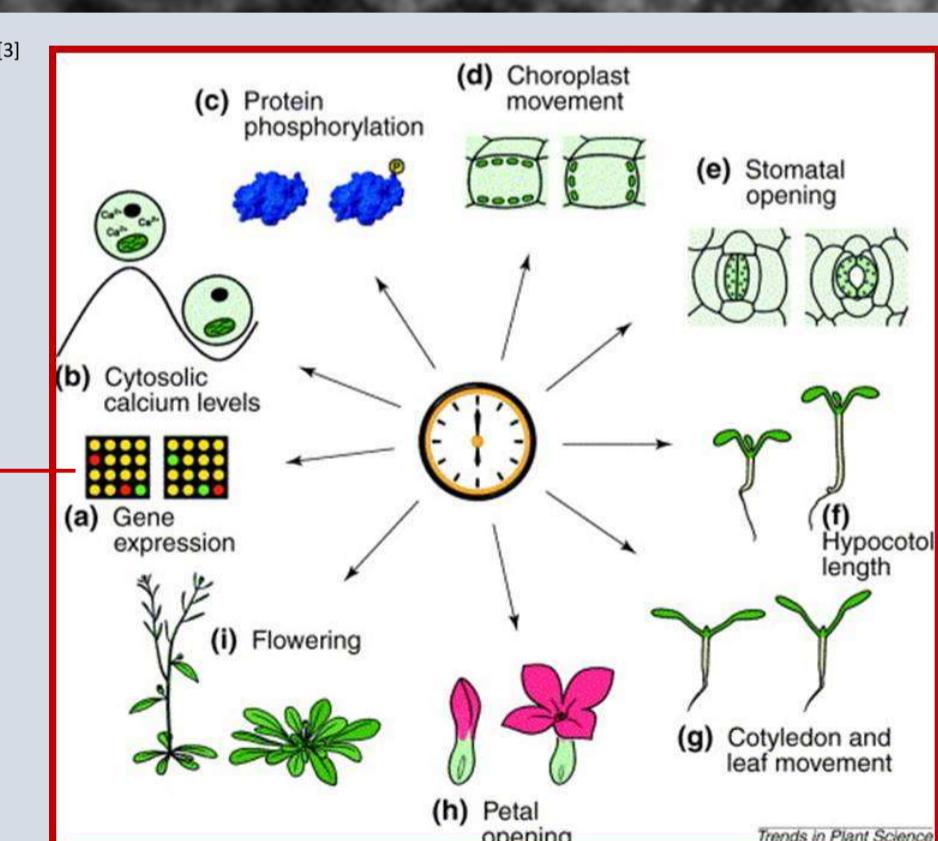
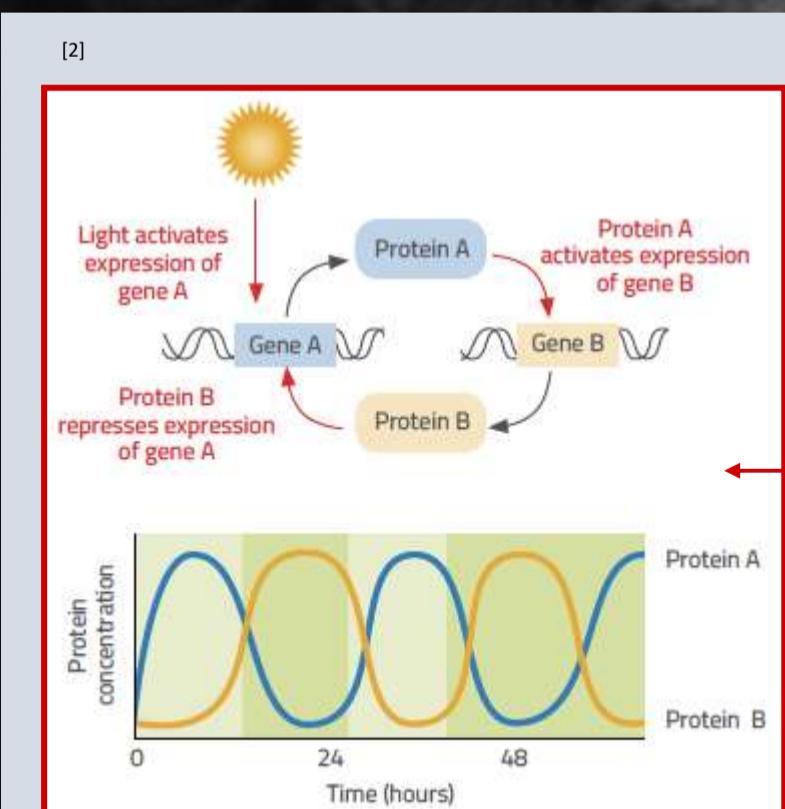


Gene expression and the Environment

Since the concentrations of certain proteins within an organism are determined by environmental factors, and certain proteins regulate gene expression, gene regulatory networks are fundamental in the mechanism of adaption organisms show in response to environmental cues.

Case study: Plant circadian rhythm's

Throughout the course of a summer's day, the structure of a plant will change. This is directly a consequence of changing protein concentrations within the plant's cells, where the changes are initiated by a gene regulatory network and sunlight.



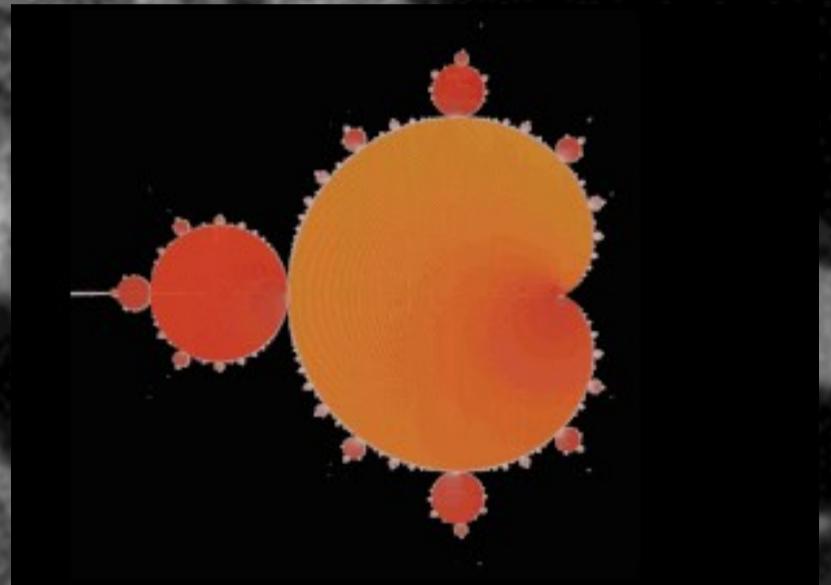
In our grasshopper script, the external signal is defined as the dot product between a normal vector in the direction of the z-axis, and a vector whose direction points from the centre of the visualization to a large rotating sun. The input is hence sinusoidal; however, it is floored at zero (this represents the darkness). This setup is meant to stimulate the pattern of light a plant or building might receive throughout the day.

Process of adaption

Fractal Gene Regulatory Networks

In 2004, Peter Bentley published a paper describing an implementation of an artificial GRN based on fractals. It is based on the following idea: In nature, the defining feature of a protein and its subsequent complex activity is its shape. Shape determines its chemistry with other proteins, amongst other things. Hence, a key requirement for any artificial GRN is an algorithmic representation of shape, and ideally one in which this notion of shape affords a compact parametrization.

Enter the Mandelbrot set:



This is a fractal, and its literal infinite amount of spatial complexity is generated using a simple recursive formula. Bentley's idea was the following – by sampling regions of the Mandelbrot set and converting to bitmaps, we have an unlimited source of digital proteins.

These regions can be parameterized using 3 values – real and imaginary coordinates, plus a window size which determines the zoom factor. If we then define a 'fractal' chemistry, in other words an algorithmic process defining how these fractal proteins can interact, we have the basis for an artificial gene regulatory network.

Algorithmic implementations of GRN's

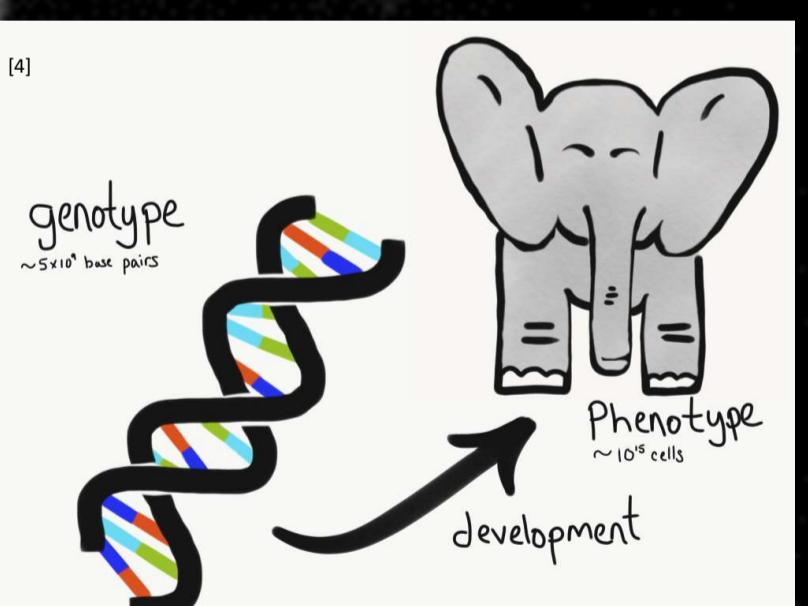
Many implementations of artificial GRN's summarise the degree of activation/inhibition between two genes through a weight matrix.

The values would represent the degree of positive reinforcement versus negative that the presence of one gene had on the expression of another.

Within the context of evolutionary computing, one downside of this is the following: the number of parameters determining the behaviour of the algorithm scales quadratically with the number of genes.

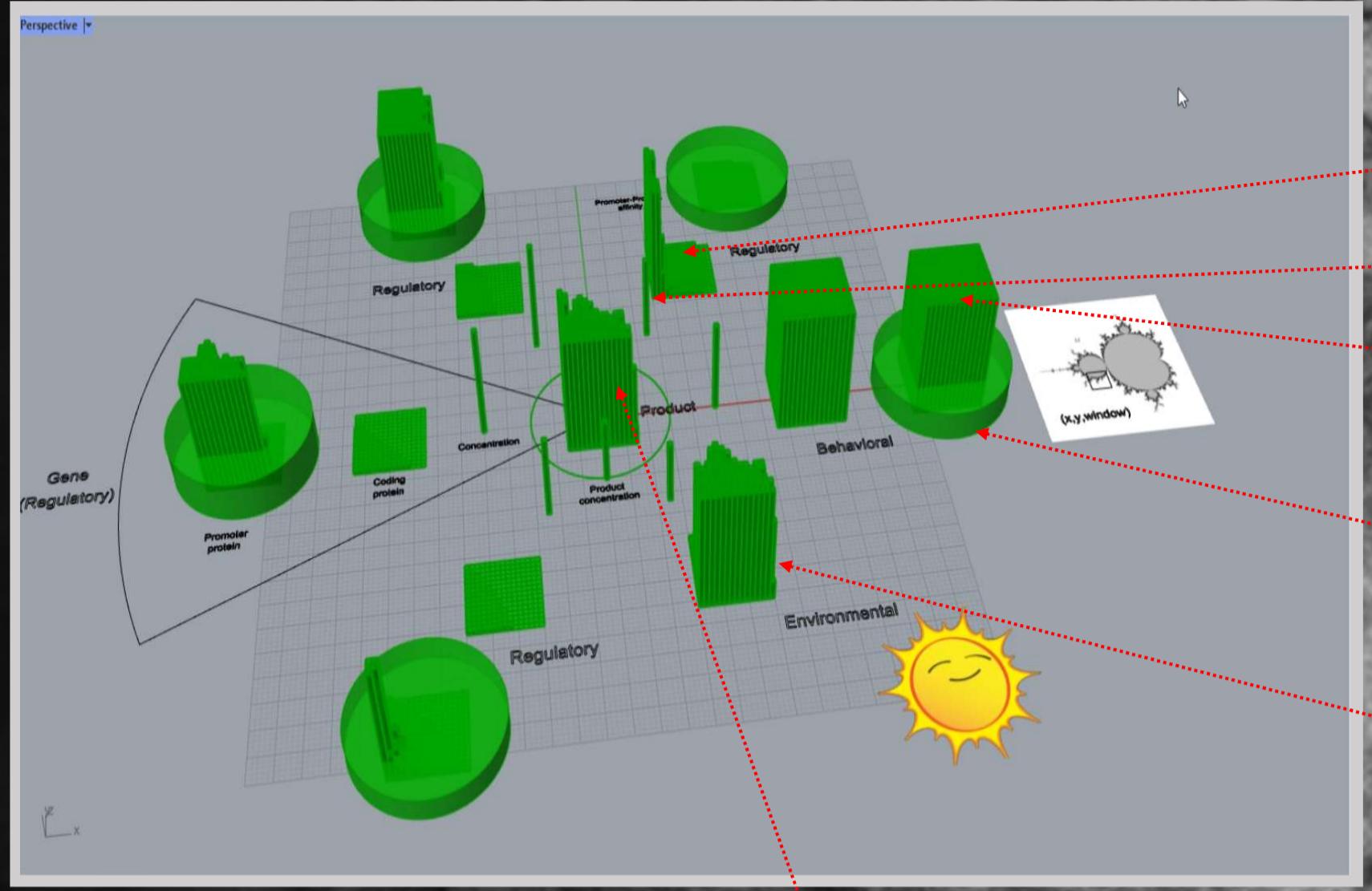
Using Bentley's fractal proteins, the number of evolvable parameters increases linearly with the number of genes.

This allows more complex networks to be evolved, and is a result of the compact genotype-phenotype mapping provided by the Mandelbrot set. This study of such compact mappings is sometimes referred to as Embryogeny.



Sources:

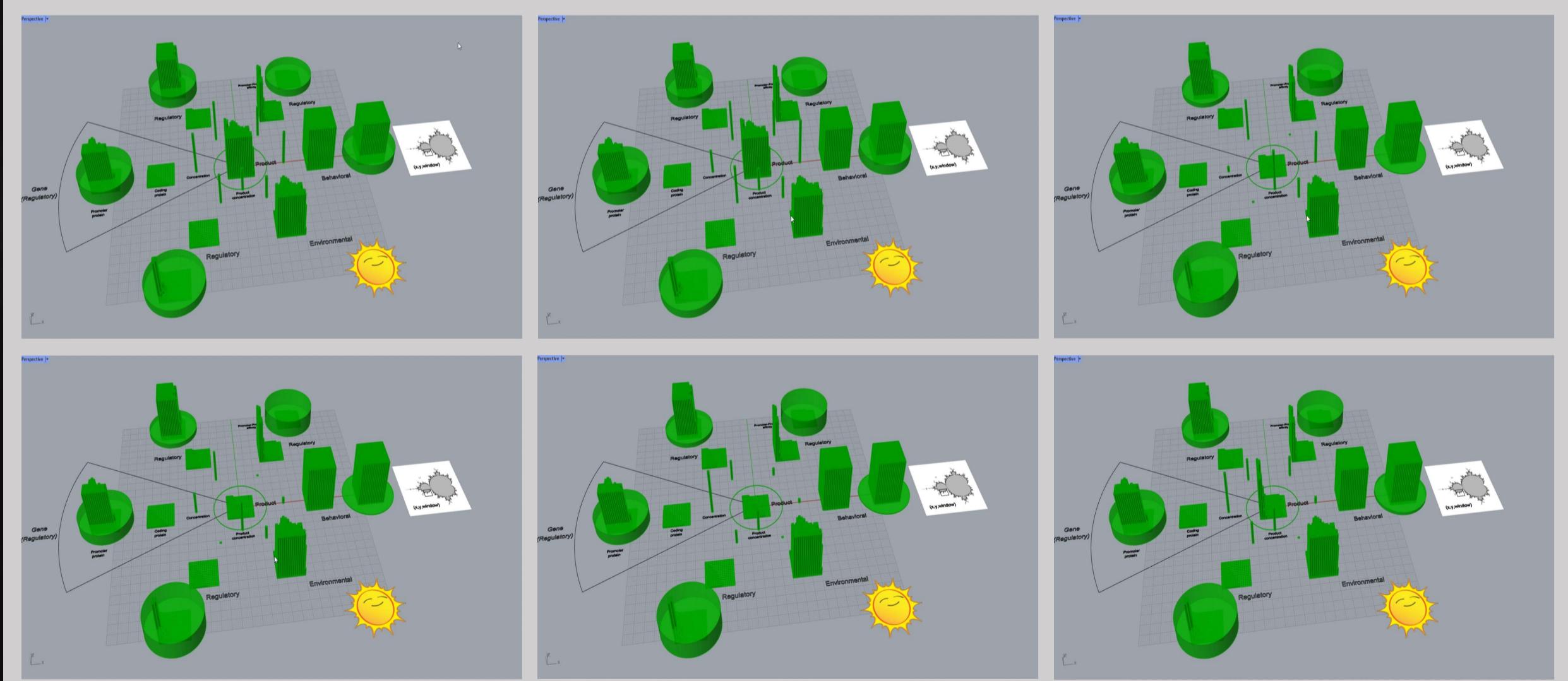
- [1] <https://www.ncbi.nlm.nih.gov/probe/docs/applexpression/>
- [2] <https://www.scienceinschool.org/content/how-plants-beat-jet-lag>
- [3] [https://www.cell.com/trends/plant-science/comments/S1360-1385\(00\)01785-4](https://www.cell.com/trends/plant-science/comments/S1360-1385(00)01785-4)
- [4] <https://devolab.org/learning-an-evolvable-genotype-phenotype-mapping/>
- [5] https://en.wikipedia.org/wiki/Mandelbrot_set



Results

Video Roll

The frames below show the changing dynamics of the GRN in response to sunlight. We can see changing coding protein concentrations and the corresponding changes in the protein product found in the cytoplasm.



Contributions:

- Removed stochastic binding probability, replaced with 'average affinity'
- Introduced concentration threshold for coding proteins, to allow for different combinations of coding proteins within the cytoplasm and hence more varied protein products
- Using Rhino3D, provided visual tool for education and further research.
- Provided C# Fractal GRN implementation for education and further research.

A gene's coding protein

The concentration of the gene's coding protein

A gene's promoter protein

The affinity between the gene's promoter protein and the cytoplasm protein product

The environmental gene and its coding protein

The protein product of the reaction between the coding proteins present in the cytoplasm

Protein concentration curves

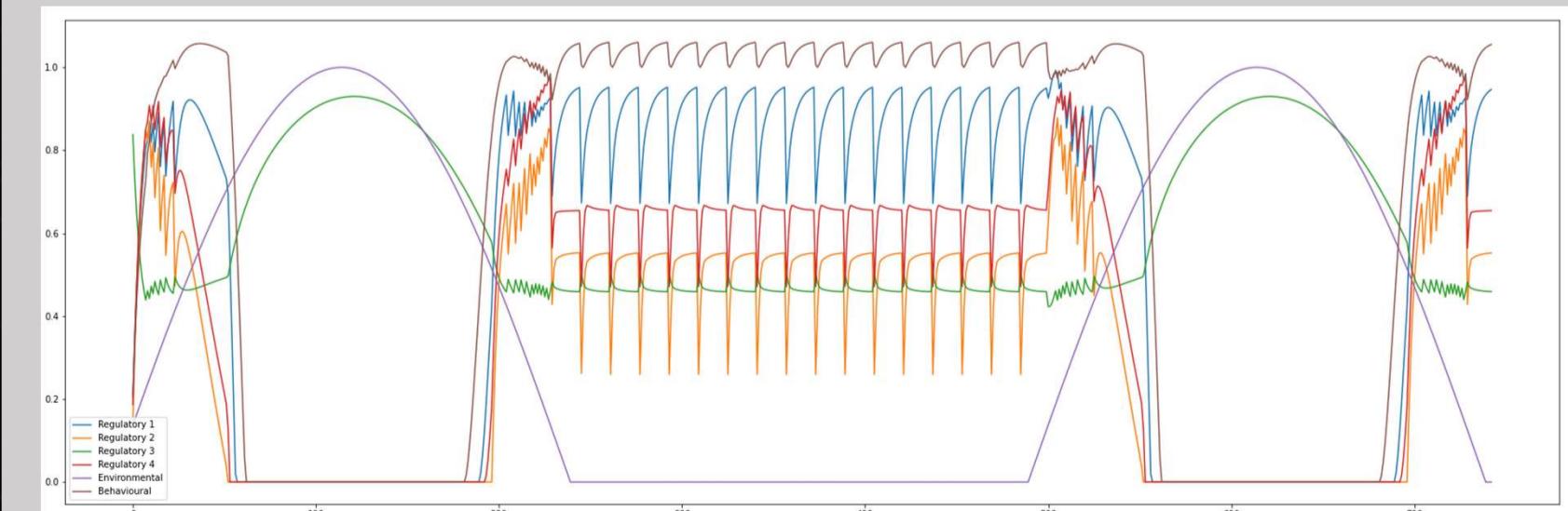
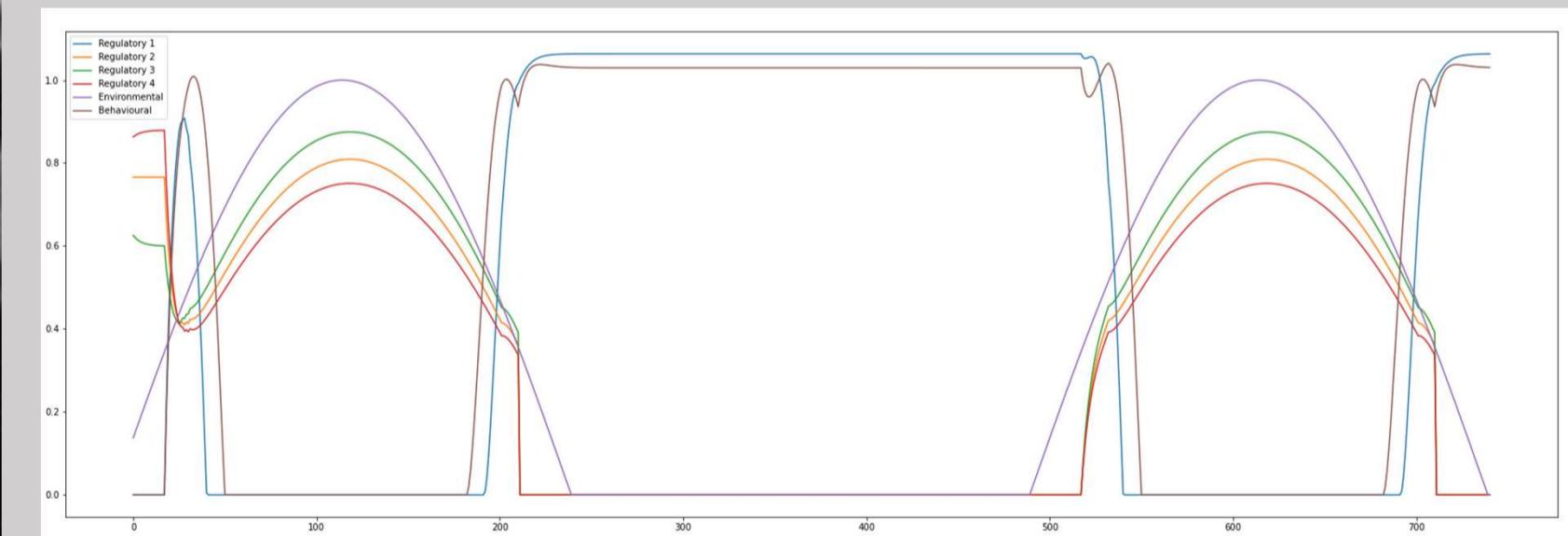
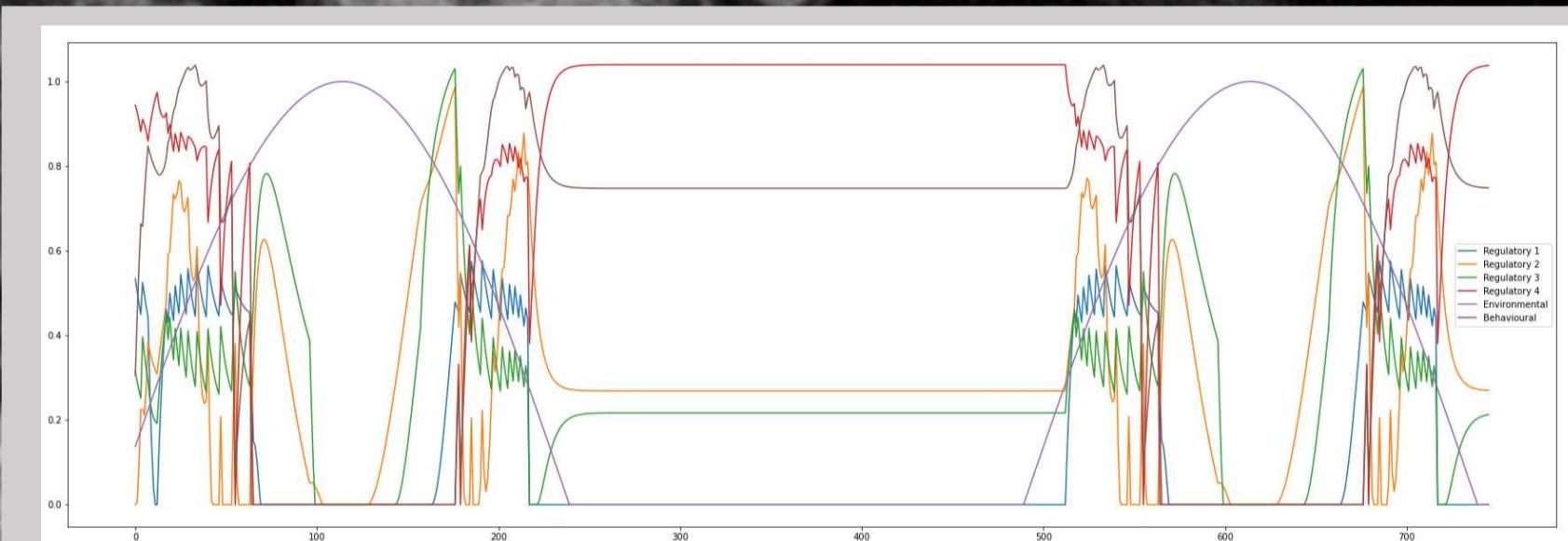
The charts below show some example concentration trajectories which are produced by our fractal gene regulatory networks, using sunlight strength as the environmental signal. Common to all three therefore is the purple line, which represents the trajectory of the sun and the corresponding light intensity. We can see that there is a period of darkness, and two sunrises.

The brown line represents the concentration of the behavioral gene's coding protein. This would therefore be the 'output' of the network, and hence determine some aspect of the organism's phenotype or behavior.

All cases show an inversion between the environmental input and the behavioral output.

Chart 1 and 3 demonstrate oscillatory behavior at a higher frequency than the environmental input, suggesting our GRN can derive new schedules/clocks.

By looking at the peaks of the curves in Chart 1, we can see various degrees of phase shift amongst the different regulatory protein's responses to the environment. This indicates that the GRN can introduce different delays into its dynamical response to the environment.



Slide Content and Code by Harry Booth, except where referenced.

Email: harrybooth@live.com