Generalisations of Recursive Sequences Using Diagonalizations of 2×2 Matrices

HYK028

1 Introduction

In popular mathematics, the Fibonacci Sequence, discovered by Leonardo Pisano, is often used as an elegant application of mathematics to nature and architecture, with the numbers from the sequence appearing often [Sin17]. The sequence is defined as such: $F_n = F_{n-1} + F_{n-2}$ where $F_0 = 0$ and $F_1 = 1$.

While most textbooks and courses find this definition perfectly feasible for education, it provides surprisingly little information about the actual sequence. I had 3 issues with this simplified definition:

1. The sequence lacks a generalisation of the n-th term of the Fibonacci sequence in terms of n
2. It is computationally inefficient in comparison to a generalisation.
3. The Golden Ratio, φ , is not easily derivable.

In some of my research, I read through the Oxford Discrete Math textbook, and was quite disappointed to see a mediocre proof of the ratio that defined Fibonacci. The textbook used a trial and error technique, reaching an approximation of the Golden Ratio using a spreadsheet. Later in the book, they provided a general n-th term, however, there was no derivation, and only an inductive proof. I wanted to develop a method to show what they had shown, but with understanding of why that is the case [Har+16].

Hence, I was prompted with the task of analysing the Fibonacci sequence using linear algebra. Thus began my journey to find the n-th term of the sequence, with whatever tools linear algebra could offer me.

1.1 Notation

1. $\langle x, y \rangle$ and $\begin{bmatrix} x \\ y \end{bmatrix}$ are representations of 2-dimensional vectors.
2. $M_{x,y}$ of any matrix denotes the value of the position in the x-th row and y-th column.
3. $\hat{i}$ and $\hat{j}$ represent the basis vectors $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$.
4. The identity matrix in $\mathbb{R}^2$ is $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

1.2 Operations

1. Matrix-vector multiplication occurs as so: $\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e \\ f \end{bmatrix} = \begin{bmatrix} ae + bf \\ ce + df \end{bmatrix}$
2. Matrix-matrix multiplication occurs as so: $\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} = \begin{bmatrix} ae + bg & af + bh \\ ce + dg & cf + dh \end{bmatrix}$

3. If $M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$, then $\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc$

4. The inverse of matrix, M: $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ is $\frac{1}{\det(M)} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$

2 Recurrence Relations

The definition of a recurrence relation as per the discrete math option outlines that every term, u_n, of a sequence is defined by a function, $f(u_{n-1}, \ldots, u_{n-k})$. The generalised n-th term of an arithmetic sequence is $u_n = u_1 + (n-1)d$ and for a geometric sequence, $u_n = u_1 r^{n-1}$. In contrast, the recurrence relation definition of an arithmetic sequence is described as: $u_n = u_{n-1} + d$, where $u_1 = a$; and the definition of a geometric sequence as: $u_n = r$ x u_{n-1}, where $u_1 = a$. The recurrence relation provides a recurring algorithmic calculation of values, often simple to understand for students, and easy to teach with. However, with the goal of mathematical elegance and extraction of crucial information about the sequence, a switch to find the n-th term is required.

For simple arithmetic and geometric sequences, the conversion is simple, and the results are equally as simple. The same cannot be said about the Fibonacci sequence that shows no simple pattern to calculate the generalisation. This is because the Fibonacci sequence is known as a second degree recurrence relation, where u_n is defined in terms of u_{n-1} and u_{n-2}. It is harder to calculate the n-th term from here.

The techniques which I found useful, I found in university lecture notes. These techniques will be systematically discussed through the paper, however they revolved around representing the Fibonacci numbers as vectors, and applying a linear transformation upon them n number of times. This is similar to how each next term of the sequence would be calculated with the recurrence. [NA16]

3 The Matrix

The first step in the method of applying an abstract vector space was to find a linear transformation matrix that could map the vector of terms F_n and F_{n+1} to the vector of terms F_{n+1} and F_{n+2} :

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} F_{n+1} \\ F_n \end{bmatrix} = \begin{bmatrix} F_{n+2} \\ F_{n+1} \end{bmatrix} = \begin{bmatrix} F_n + F_{n+1} \\ F_{n+1} \end{bmatrix}$$

Using the definition of matrix multiplication by a vector, we can derive the equations:

$$aF_{n+1} + bF_n = F_n + F_{n+1} \tag{1}$$
$$cF_n + dF_{n+1} = F_{n+1} \tag{2}$$

By inspection, the solution for Equation 1 occurs when $a = 1$ and $b = 1$. Since a and b only appear once in Equation 1 only, any values that satisfy the first equation must satisfy both equations. The same argument can be made for Equation 2, with values

$c = 0$ and $d = 1$. This shows that the matrix, M, which outputs values of the Fibonacci sequence is:

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

4 Attempting to Generalise

Noticing that repeatedly applying this matrix to a vector would provide me with the Fibonacci sequence, I first attempted to find a pattern within the actual matrix. If the vector with the n-th and $(n+1)$th term is V, then applying MV once, produces the next term; applying M again produces the next term and so forth. For simplicity, we denote applying a matrix n times as M^n. Assuming $F_0 = 0$ and $F_1 = 1$, applying M^n to the vector $\langle 1, 0 \rangle$ would provide the output vector with the $(n+1)$th and $(n+2)$th term of the Fibonacci sequence.
First, I calculated the matrix multiplications to get an idea of what M^n might look like:

$$M^1 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \Rightarrow M^2 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \Rightarrow M^3 = \begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix} \Rightarrow M^4 = \begin{bmatrix} 5 & 3 \\ 3 & 2 \end{bmatrix}$$

The pattern I noticed was that $M^n_{2,2}$ provides the $(n-1)$th term of the sequence. $M^n_{1,2}$ and $M^n_{2,1}$ provide the n-th term of the sequence. $M^n_{1,1}$ provides the $(n+1)$th term. This can be proved by induction:

Theorem 1.

$$M^n = \begin{bmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{bmatrix} \textit{ for all } n \in \mathbb{N}$$

Proof. Let $n = 1$:

$$M^1 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Let $k \in \mathbb{N}$ be given and suppose M^n is true for $n = k$. Then:

$$M^k = \begin{bmatrix} F_{k+1} & F_k \\ F_k & F_{k-1} \end{bmatrix}$$

Inductive step for when $n = k + 1$:

$$M^{k+1} = \begin{bmatrix} F_{k+2} & F_{k+1} \\ F_{k+1} & F_k \end{bmatrix}$$

$$M^{k+1} \tag{3}$$
$$= M \cdot M^k \tag{4}$$
$$= \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} F_{k+1} & F_k \\ F_k & F_{k-1} \end{bmatrix} \tag{5}$$
$$= \begin{bmatrix} F_{k+1} + F_k & F_k + F_{k-1} \\ F_{k+1} & F_k \end{bmatrix} \tag{6}$$
$$= \begin{bmatrix} F_{k+2} & F_{k+1} \\ F_{k+1} & F_k \end{bmatrix} \tag{7}$$

Since $n = 1$ is true, and when $n = k$ is assumed to be true, $n = k + 1$ is also true, Theorem 1 holds. QED

The information provided by these findings was crucial in my process. I found that M^n's elements were the Fibonacci sequence itself. At first, it seemed like this was a dead end, since I thought I had gone into an inevitable circle. In hopes to find a more interesting pattern behind the Fibonacci sequence, I just found the Fibonacci sequence hidden underneath. I spent a few days pondering the problem after which I found that this result was actually mathematically useful. I only needed to generalise the matrix and that would solve the problem immediately since the Fibonacci sequence is hidden inside the matrix itself.

5 Linear Transformations

Before understanding the methodology used to diagonalize the matrix, it is important to understand how applying a matrix to any vector in vector space, $\mathbb{R}^2$, will change the vector. A vector, V is traditionally represented with:

$$\begin{bmatrix} a \\ b \end{bmatrix}$$

which can be written as $V = a\hat{\imath} + b\hat{\jmath}$. This transforms our understanding of a vector. Traditionally, a and b represent the x and y coordinates of a vector, however this alternate notation of a vector allows us to understand that the vector is a combination of multiples of the basis vectors, $\hat{\imath}$ and $\hat{\jmath}$. We can represent this definition as:

$$a \begin{bmatrix} 1 \\ 0 \end{bmatrix} + b \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

It is important to notice that this is the same as:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

What I understood from this was that a matrix is simply a method for describing the basis in which the vector is functioning within. For a vector in our traditional system these basis vectors are $\hat{\imath}$ and $\hat{\jmath}$ which show themselves in the first and second column of the matrix respectively. This also told me that any vector space spanning $\mathbb{R}^2$ would have a matrix defining its basis vectors with dimensions 2 x 2.

Using the definition that matrices have control over the basis vectors, we can understand what a matrix does to a vector. If any matrix is applied to a vector, it takes the coordinate system that the matrix symbolises and the vector that exists in that coordinate system and puts it into our coordinate system.

$$\text{Given the matrix: } B = \begin{bmatrix} i_x & j_x \\ i_y & j_y \end{bmatrix} \text{ and the vector: } \begin{bmatrix} a \\ b \end{bmatrix}$$

When the matrix, B is applied to the vector the output of the multiplication is the vector, $ai + bj$. This means that the vector output of their multiplication is the addition of i, a

number of times, and j, b number of times. This means that i and j are the new basis vectors for this vector. In our coordinate space, $\langle a, b\rangle$ would represent a vector addition of $\hat{\imath}$, a number of times, and $\hat{\jmath}$, b number of times. The matrix redefines the coordinate space for us. It gives us the answer to the question: "If $\langle a, b\rangle$ was defined in a different coordinate space, with basis vectors i and j, how would we describe that vector in our own coordinate space?". Since the matrices are able to make changes to vectors by changing the basis vectors of the matrix, we can define a matrix as a linear transformation of the entire vector space [NA16].

6 Further Results

Further discovery showed me this result which can is obvious by inspection: Just as B transforms the vector from another "language" with alternate bases into our language with the regular bases, B^{-1}, the inverse of B, will transform the vector in our language back into the language of alternate bases.

Using that result, I then found that any linear transformation itself can undergo this process. For any matrix M that defines a linear transformation to a vector space, I can change the language of the linear transformation to satisfy the language of the vectors that I want to change. For example, I have the vector $\langle a, b\rangle$, that is defined in a language I have never seen before, and I need to apply the linear transformation, M, that is in my language. Naturally I would first convert the vector to my own language by applying the change of language matrix like so:

$$B \begin{bmatrix} a \\ b \end{bmatrix}$$

Now that the result of this matrix is in my language, I can apply the linear transformation, M:

$$MB \begin{bmatrix} a \\ b \end{bmatrix}$$

The result of the linear transformation is now a vector in my language, but to change it into the alternative language would require the inverse conversion matrix:

$$B^{-1}MB \begin{bmatrix} a \\ b \end{bmatrix}$$

This means that any linear transformation in our language can be applied to a vector of any language by sandwiching the linear transformation in between the conversion matrix and the inverse of the conversion matrix [NA16].

This is a very useful conclusion since it means that we can deal with any linear transformation matrix including the Fibonacci one that we created.

7 Eigenvectors

Now we will look into Eigenvectors, and their properties which will come in handy for diagonalizing the matrix later on. Before delving in, there are some key definitions:

Span: The span of a vector is the line on which the vector exists in vector space. A vector that lies on the same span as another vector is a scalar multiple of the original vector.

Determinant: The determinant of any given matrix is the area of the parallelogram that is created by the associated basis vectors of that matrix, i and j.

As we discussed earlier, matrices are linear transformations that transform the vector space. For most vectors, the vector will be transformed off of the vector's original span. For example, the vector $\langle 1, 0 \rangle$, being transformed by matrix, N:

$$\begin{bmatrix} 2 & 0 \\ 1 & 1 \end{bmatrix}$$

The original vector lies on the span of $x = 0$, and the transformed vector, $\langle 2, 1 \rangle$, lies on the span $y = \frac{1}{2}x$. This vector, and most vectors get thrown off of their span, but there are a few vectors that stay on their span during the transformation. For the above matrix, it happens to be that the vector, $\langle 1, 1 \rangle$, on the span $y = x$, transforms to the vector, $\langle 2, 2 \rangle$, that also lies on the span $y = x$. This vector, $\langle 1, 1 \rangle$, is known as an Eigenvector for this matrix, since it stays on its own span. This Eigenvector has a related Eigenvalue which is the scalar by which the vector is multiplied to get the transformed vector. In this example, the Eigenvalue of the Eigenvector, $\langle 1, 1 \rangle$, is 2, since the vector is being scaled by 2.

The method for calculating Eigenvectors of a matrix M of form $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ can be split into 2 parts: finding the eigenvalues and finding the eigenvectors.

7.1 Calculating Eigenvalues

The method for calculating the Eigenvalues for a matrix M is by using the equation:

$$Mv = \lambda v$$

where M is the matrix in question, v is any given vector in the vector space, and λ represents the Eigenvalue. Since the equation is equating a matrix-vector product to a scalar-vector product, we cannot continue with algebra. The value of λ can easily be represented as the matrix λI since the identity matrix has no effect upon the matrix itself. This is the property of the identity of any set. λI would look like so:

$$\begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}$$

Thus, we can continue with calculations. The above equation can be manipulated as so:

$$Mv = \lambda I v \tag{8}$$
$$Mv - \lambda I v = 0 \tag{9}$$
$$(M - \lambda I)v = 0 \tag{10}$$

This is an important step, since it tells us that whatever the matrix $M - \lambda I$ is, when it is applied to any vector in the vector space, it collapses that vector to 0. The only way that this can occur is if the determinant of that matrix is 0, which is a result that follows from linear algebra that are unimportant to the content of this IA. Thus, solve for when the determinant of $M - \lambda I$ is 0:

$$\det \begin{bmatrix} a-\lambda & b \\ c & d-\lambda \end{bmatrix} = 0 \tag{11}$$

$$(a-\lambda)(d-\lambda) - bc = 0 \tag{12}$$

After this, simple algebra can solve for the Eigenvalues. Note that since this is a quadratic, there can exist 0, 1, or 2 real Eigenvalues. For the sake of simplicity, complex Eigenvalues will be ignored.

7.2 Calculating Eigenvectors

For however many real Eigenvalue solutions exist, those amount of real Eigenvectors usually exist. There are scenarios where the matrix has an infinite number of Eigenvectors, but they do not concern the aim at hand, and do not appear in this IA. The calculation of the Eigenvector associated with an Eigenvalue, λ_i, is shown below:

$$(M - \lambda_i I)v = 0 \tag{13}$$

$$(M - \lambda_i I)v = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{14}$$

Since the values of M and λ_i are known, the vector, v can be calculated. This vector is the Eigenvector for one Eigenvalue and there exist more Eigenvectors for every Eigenvalue.

7.3 Application to Eigenbases

So far, the work into Eigenvectors is somewhat abstract and far deviated from actually solving the problem at hand, which is to generalise the Fibonacci matrix. However, the Eigenvectors are very useful, especially when combined with another technique discussed earlier involving change of basis.

It is possible to redefine any given linear transformation in the bases of its Eigenvectors. This would mean creating the change of basis matrix, B, for the two Eigenvectors of the transformation matrix. I found that this would change the matrix in a way that could make it much easier to generalise. Let us try doing this for the matrix above, N:

$$N = \begin{bmatrix} 2 & 0 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad \lambda_1 = 1, \quad \lambda_2 = 2$$

$$v_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}, \quad B^{-1} = \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix}$$

$$B^{-1}NB = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$$

The interesting part of this matrix is that it is diagonal. There only exist values in the matrix along the diagonal from the top left to the bottom right. This is a special result when the bases are changed to the Eigenvectors to create Eigenbases. This is because the transformation is now defined by a stretching and pulling of the basis vectors. At first glance, this isn't very useful, but the useful thing about diagonalising the matrix is the following property:

$$\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}^n = \begin{bmatrix} a^n & 0 \\ 0 & b^n \end{bmatrix}$$

This means that once a matrix is diagonalised, it can be generalised for n multiplications of the matrix. Once the matrix has been operated upon in its diagonalised state, it can be converted back to regular form by applying the same conversion matrices from earlier, in reverse order:

$$B \begin{bmatrix} a^n & 0 \\ 0 & b^n \end{bmatrix} B^{-1}$$

8 Generalising the Fibonacci Matrix

All the above techniques only need to be applied to the Fibonacci matrix, M:

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

8.1 Calculating Eigenvalues

The Eigenvalues of M are calculated as follows:

$$\det \begin{bmatrix} 1-\lambda & 1 \\ 1 & 0-\lambda \end{bmatrix} = 0 \tag{15}$$

$$(1-\lambda)(-\lambda) - (1)(1) = 0 \tag{16}$$

$$\lambda^2 - \lambda - 1 = 0 \tag{17}$$

$$\lambda_1 = \frac{1+\sqrt{5}}{2}, \quad \lambda_2 = \frac{1-\sqrt{5}}{2}$$

Notice that the eigenvectors are the the Golden Ratio, φ_1, and Golden Ratio conjugate, φ_2. Thus, we can continue to refer to the Eigenvalues as φ_1 and φ_2 for the sake of convenience and neat calculations.

8.2 Calculating Eigenvectors

8.2.1 For $\lambda = \varphi_1$:

$$(M - \varphi_1 I)v = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{18}$$

$$\begin{bmatrix} 1-\varphi_1 & 1 \\ 1 & -\varphi_1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{19}$$

Note that $1 - \varphi_1 = \varphi_2$. Use Gaussian Elimination to solve:

$$\left[\begin{array}{cc|c} \varphi_2 & 1 & 0 \\ 1 & -\varphi_1 & 0 \end{array}\right] \tag{20}$$

$$\left[\begin{array}{cc|c} -\varphi_1\varphi_2 & -\varphi_1 & 0 \\ 1 & -\varphi_1 & 0 \end{array}\right] \tag{21}$$

$$\left[\begin{array}{cc|c} 1 & -\varphi_1 & 0 \\ 1 & -\varphi_1 & 0 \end{array}\right] \tag{22}$$

$$\left[\begin{array}{cc|c} 1 & -\varphi_1 & 0 \\ 0 & 0 & 0 \end{array}\right] \tag{23}$$

This produces the system of equations:

$$v_x - \varphi_1 v_y = 0 \tag{24}$$

$$v_y = v_y \tag{25}$$

If $v_y = 1$, which can be plugged in arbitrarily since v_y is an independent variable, then, $v_x = \varphi_1$ and the Eigenvector, v_1, is:

$$\begin{bmatrix} \varphi_1 \\ 1 \end{bmatrix}$$

8.2.2 For $\lambda = \varphi_2$:

Similar operations can be carried out and the second Eigenvector, v_2, is :

$$\begin{bmatrix} \varphi_2 \\ 1 \end{bmatrix}$$

8.3 Diagonalisation

The two Eigenvectors can be constructed into a change of basis matrix, B:

$$\begin{bmatrix} \varphi_1 & \varphi_2 \\ 1 & 1 \end{bmatrix}$$

The inverse of the change of basis matrix, B^{-1}:

$$\begin{bmatrix} 1 & -\varphi_2 \\ -1 & \varphi_1 \end{bmatrix}$$

Then, we can apply these matrices following the technique described earlier:

$$\frac{1}{\sqrt{5}} \begin{bmatrix} 1 & -\varphi_2 \\ -1 & \varphi_1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \varphi_1 & \varphi_2 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} \varphi_1 & 0 \\ 0 & \varphi_2 \end{bmatrix}$$

Such a matrix is easy to generalise for n applications as :

$$\begin{bmatrix} \varphi_1^n & 0 \\ 0 & \varphi_2^n \end{bmatrix}$$

The only issue with this generalisation is that it describes a language where the Eigenvectors are the bases of the vector space, and the transformation needs to be converted back to the language where the basis vectors are $\hat{i}$ and $\hat{j}$. The technique is discussed above, the same conversion matrix and inverse of the conversion matrix can be applied in reverse order.

$$\frac{1}{\sqrt{5}}\begin{bmatrix}\varphi_1 & \varphi_2\\ 1 & 1\end{bmatrix}\begin{bmatrix}\varphi_2^n & 0\\ 0 & \varphi_1^n\end{bmatrix}\begin{bmatrix}1 & -\varphi_2\\ -1 & \varphi_1\end{bmatrix}$$

8.4 n-th Term of Fibonacci

At first I attempted to calculate this by hand and go through the calculations, but I realised that for whatever matrix, M^n, these set of matrices gave, $M^n_{1,1}$ would provide F_{n+1} of the Fibonacci sequence, as proven by induction earlier. Therefore, I simply calculated $M^n_{1,1}$ to save space since the final matrix is very large and complex. Therefore I found that:

$$F_{n+1} = \frac{\varphi_1^{n+1} - \varphi_2^{n+1}}{\sqrt{5}}$$

and it follows that:

$$F_n = \frac{\varphi_1^n - \varphi_2^n}{\sqrt{5}}$$

This is the final solution to the very problem that was being investigated. This is an established formula known as Binet's formula, since it was first derived by Jacques Philippe Marie Binet in 1812 using the same concept of matrix multiplication [SR06]. Using this equation, the first two problems discussed in the introduction have been solved. Firstly, a generalisation has been achieved. For any value of n, F_n can be found. Secondly, the value of F_n can be calculated instantly, in a single operation. In the previous recurrence relation algorithm, finding the n-th term would take n operations. A generalisation and computational efficiency have been reached. Finally, the ratio between Fibonacci terms becomes apparent as will be seen in the next section.

9 Fibonacci Ratio

The ratios for the first 10 terms are shown in the table below, and a clear pattern is seen. The ratios converge upon a values 1.618, or φ_1.

As n grows larger, the value of the ratio converges to φ_1. This can be proven by computing the limit as n approaches ∞ of the ratio between F_{n+1} and F_n.

Theorem 2. $\lim_{x\to\infty} \frac{F_{n+1}}{F_n} = \varphi_1$

Proof.

$$\lim_{n\to\infty} \frac{\frac{\varphi_1^{n+1}-\varphi_2^{n+1}}{\sqrt{5}}}{\frac{\varphi_1^{n}-\varphi_2^{n}}{\sqrt{5}}} \tag{26}$$

$$= \lim_{n\to\infty} \frac{\varphi_1^{n+1} - \varphi_2^{n+1}}{\varphi_1^n - \varphi_2^n} \tag{27}$$

n	F_n	F_{n+1}	Ratio
1	1	1	1.0000
2	1	2	2.0000
3	2	3	1.5000
4	3	5	1.6667
5	5	8	1.6000
6	8	13	1.6250
7	13	21	1.6154
8	21	34	1.6190
9	34	55	1.6176
10	55	89	1.6182

Note that $\varphi_2 = -\frac{1}{\varphi_1}$ since the product of the roots φ_1 and φ_2 is -1, derived from the formula for the product of roots in the equation, $x^2 - x - 1$.

$$= \lim_{n\to\infty} \frac{\varphi_1^{n+1} - (-\frac{1}{\varphi_1})^{n+1}}{\varphi_1^n - (-\frac{1}{\varphi_1})^n} \tag{28}$$

$$= \lim_{n\to\infty} \frac{\varphi_1^{n+1} - \frac{(-1)^{n+1}}{\varphi_1^{n+1}}}{\varphi_1^n - \frac{(-1)^n}{\varphi_1^n}} \tag{29}$$

$$= \lim_{n\to\infty} \frac{\frac{\varphi_1^{2n+2} - (-1)^{n+1}}{\varphi_1^{n+1}}}{\frac{\varphi_1^{2n} - (-1)^n}{\varphi_1^n}} \tag{30}$$

$$= \lim_{n\to\infty} \frac{\frac{\varphi_1^{2n+2} - (-1)^{n+1}}{\varphi_1}}{\varphi_1^{2n} - (-1)^n} \tag{31}$$

$$= \lim_{n\to\infty} \frac{\varphi_1^{2n+2} - (-1)^{n+1}}{\varphi_1(\varphi_1^{2n} - (-1)^n)} \tag{32}$$

$$= \lim_{n\to\infty} \frac{\varphi_1^{2n+2} - (-1)^{n+1}}{\varphi_1^{2n+1} - \varphi_1(-1)^n} \tag{33}$$

Note that the terms $(-1)^{n+1}$ and $\varphi_1(-1)^n$ are constant even as n approaches ∞, allowing them to be removed from the limit problem since their sizes are infinitesimal compared to φ_1^{2n+2} and φ_1^{2n+1}.

$$= \lim_{n\to\infty} \frac{\varphi_1^{2n+2}}{\varphi_1^{2n+1}} \tag{34}$$

$$= \lim_{n\to\infty} \varphi_1 = \varphi_1 \tag{35}$$

QED

This was the final problem of the introduction of deriving the Golden Ratio. Using this variation of calculating Fibonacci terms, the ratio is easily seen with solving a simple limit.

10 Extension

The initial problem was solved, with a straight forward application of a technique utilising key properties in linear algebra. While the techniques served a functional purpose, and that is the main focus of this IA till now, I developed an interest for the original transformation matrix that defined the recurrence relation. I then conducted 2 mini-investigations that would exceed the contents of this IA and stray from the aim significantly, however, they are worth mentioning as points of further discovery.

10.1 Tribonacci Sequence

Since this technique worked for a 2-dimensional vector space, I wondered how the technique would function in a 3-dimensional vector space showing the progression of Tribonacci terms, which followed a form of $T_n = T_{n-1} + T_{n-2} + T_{n-3}$. Interestingly, I was able to devise a matrix that represented the transformation of this matrix as well:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

My first thought was to investigate the possibility of diagonalisation. Therefore, I computed an equation to find the number of eigenvalues with the same determinant technique as above, which produced the equation: $-\lambda^3 + \lambda^2 + \lambda + 1$. My first reaction was that this was a challenge. The equation only has 1 real Eigenvalue, and therefore only one real Eigenvectors. Soon, I realised that this didn't matter since I could use complex roots as Eigenvalues, and have complex Eigenvectors [NA05]. I didn't carry on with the work to generalise the Tribonacci sequence, since that would require a significant further amount of work which would exceed the contents of this IA.

However, before moving on from this idea, I thought about k-bonacci sequences, and found a matrix for the 4-bonacci sequence:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

This is where I noticed a pattern. For any k x k matrix describing a k-bonacci sequence, the first row of elements are 1's and the remaining matrix is a diagonal of 1's in a field of 0's. This is a direction in which I can continue working on my IA. The generalisation for the n-th term of the k-bonacci sequence is the next step for such an investigation.

10.2 Other Recurrence Relations

The recurrence relation in this IA was a simple one, with two term addition. However, through experimentation and playing around with the matrices I found that a similar matrix that describes the Fibonacci recurrence relation can describe other Recurrence relations of the form: $a_n = xa_{n-1} + ya_{n-2}$. The change in this matrix is that the terms are being multiplied by constants, x and y. In the Fibonacci sequence, x and y are both 1. I found empirically that for such a recurrence relation, the transformation matrix would be:

$$\begin{bmatrix} x & y \\ y & 0 \end{bmatrix}$$

I found that looking into multiples of each term was an interesting direction to take the investigation. I didn't do as much investigating in this direction, but it is definitely a possibility, and surely something that could be fascinating.

11 Conclusion

I started off this IA with a goal to generalise a simple recurrence relation that exists in mathematics, using an unorthodox methodology. I saw that there was a lack of proper derivation in most high school textbooks for second degree recurrence relations, and I sought for an elegant solution. I was able to use key properties which I learned through exploring linear algebra on an entirely different field of mathematics, discrete mathematics. Eventually, I was able to reach a derivation which was identical to the one described in the textbook, and in the literature. This showed me that my aim was achieved. I had found an elegant, and rigorous proof to the Fibonacci sequence.

The usage of linear algebra in this manner revealed its versatility in mathematics. Regarding the original problem of recurrence relations, there were a few extensions, all using linear algebra. However, the more I ponder the concept of using linear algebra, I realise the potential of using linear algebra in nearly every field of mathematics. If a simple sequence can be represented by vectors and a matrix, then it follows that more mathematical concepts can be represented in this way. Perhaps linear algebra can be applied to integral calculus (which is also linear). Mathematically speaking, linear algebra can now be used as a tool where other tools don't work anywhere near as well.

This also leads me to think about it computationally. In computer science, I often see the concept of arrays, which are mathematically vectors, and also 2-dimensional arrays, which are matrices. Ultimately, by representing mathematical problems as vectors and matrices, a computer can solve them much faster. This only strengthens my conclusion that linear algebra is an extremely useful tool.

References

[Har+16] Josip Harcet et al. “Mathematics Higher Level: Discrete Mathematics”. In: Great Clarendon Street, Oxford, OX2 6DP, United Kingdom: Oxford University Press, 2016, pp. 78–80.

[NA05] N.A. *COMPLEX EIGENVALUES OF REAL MATRICES*. Tech. rep. University of Tennessee, Knoxville, 2005.

[NA16] N.A. *Designing Information Devices and Systems I*. Lecture Notes 21. University of California, Berkely, 2016.

[Sin17] Sudipta Sinha. “The Fibonacci Numbers and Its Amazing Applications”. In: *International Journal of Engineering Science Invention* 6.9 (Oct. 2017), pp. 7–14.

[SR06] Alexey Stakhov and Boris Rozin. “Theory of Binet formulas for Fibonacci and Lucas p-numbers”. In: *Chaos, Solitons and Fractals* 27 (Apr. 2006), pp. 1162–1177.