

On The Lagrange Interpolation of Fibonacci Sequence

Muhammad Syifa'ul Mufid, Tahiyatul Asfihani and Lukman Hanafi

Abstract—Fibonacci sequence is one of the most common sequences in mathematics. It was first introduced by Leonardo Pisa in his book *Liber Abaci* (1202). From the first $n + 1$ terms of Fibonacci sequence, a polynomial of degree at most n can be constructed using Lagrange interpolation. In this paper, we show that this Fibonacci Lagrange Interpolation Polynomial (FLIP) can be obtained both recursively and implicitly.

Index Terms—Fibonacci sequence, Lagrange interpolation.

I. INTRODUCTION

AT first, Fibonacci sequence is the solution of the pairs of rabbits problem [1]

"In the beginning we only have one pair of rabbits (male and female). How many pairs of rabbits we have after one year if every month each pair begets a new pair which from the second month on becomes productive?"

After many years passed, Fibonacci sequence surprisingly can be found in nature such as in nautilus shells, pine cones and sunflowers [1]. In mathematics, Fibonacci occurs in many fields for instance in linear algebra, combinatorics, and discrete mathematics. There are some developments about Fibonacci sequence, for example Fibonomial (Fibonacci and Binomial) [2], [3] and Fibonacci Polynomial [4], [3].

The study of polynomials generated by Fibonacci sequence is quite interesting. Fibonacci Polynomials (FP) are generated recursively similar to Fibonacci sequence [3] while Fibonacci-Coefficient Polynomials (FCP) are generated by putting Fibonacci sequence as coefficient [4]. In this paper, we propose a novel approach to construct polynomial based on Fibonacci sequence. The idea is by locating Fibonacci sequences as points in coordinate system and then find a polynomial that passes through those points using Lagrange Interpolation.

The outline of this paper is as follows. In Section 2 the preliminary of this paper is given. Section 3 defines the Lagrange interpolation of Fibonacci sequence to get FLIP. Hereafter, we show that FLIP can be obtained both recursively and implicitly. Finally, the conclusion of this paper is given in Section 4.

Manuscript received October 18, 2016; accepted October 28, 2016. This paper was presented at the 1st International Conference on Mathematics: Pure, Applied and Computation (ICoMPAC), December 3, Surabaya, Indonesia, 2015.

The authors are with the Department of Mathematics, Institut Teknologi Sepuluh Nopember, Surabaya 60111, Indonesia. Email: {syifaul.mufid, t_asfihani, lukman}@matematika.its.ac.id

II. NOTATIONS AND PRELIMINARIES

We denote \mathbb{N} as the set natural number and $\underline{n} = \{0, 1, 2, \dots, n\}$. We also use combinatorial notations such as $n! = n \times (n - 1) \times \dots \times 1$ and $\binom{n}{i} = \frac{n!}{i!(n-i)!}$ and a binomial identity

$$(a + b)^n = \sum_{i=0}^n a^i b^{n-i} \binom{n}{i}. \quad (1)$$

A. Fibonacci Sequence

Definition 1 (Fibonacci Sequence [3]): Fibonacci sequence is a recursive sequence defined by

$$f_{n+1} = f_n + f_{n-1}, n \in \mathbb{N} \quad (2)$$

with $f_0 = f_1 = 1$.

The implicit formula of Fibonacci sequence is

$$f_n = \frac{1}{\sqrt{5}} [\varphi^{n+1} - v^{n+1}], \quad (3)$$

where $\varphi = \frac{1+\sqrt{5}}{2}$ and $\varphi + v = 1$. The irrational number φ is well known as *golden section* or *golden ratio*. Both φ and v are the roots of $x^2 - x - 1 = 0$.

B. Lagrange Interpolation

In the field of numerical analysis, interpolation is a process finding the most fitted function from some specified points. The simplest approximation of an interpolation is a polynomial [5]. It means that given some points, there exists a polynomial that passes through that points. And the polynomial is close to the function. One of the polynomial interpolation methods is Lagrange Interpolation. Suppose we have $n + 1$ different points $(x_i, y_i), i \in \underline{n}$. We need find a polynomial $L(x)$ that satisfies $L(x_i) = y_i$ for all $i \in \underline{n}$. The basic idea of Lagrange Interpolation is constructing the basis polynomials $l_{n,i}(x)$ such that

$$l_{n,i}(x_j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

for all $i, j \in \underline{n}$. By considering all factors $(x - x_i)$ for $i \in \underline{n}$, we can construct $l_{n,i}(x)$ as follows

$$l_{n,i}(x) = \prod_{j \neq i} \frac{x - x_j}{x_i - x_j}. \quad (4)$$

After constructing these basis polynomials, we can generate the Lagrange Interpolation polynomial as follows

$$L(x) = \sum_{i=0}^n y_i l_{n,i}(x) = \sum_{i=0}^n \left(y_i \prod_{j \neq i} \frac{x - x_j}{x_i - x_j} \right). \quad (5)$$

III. INTERPOLATING THE FIBONACCI SEQUENCE

Before starting the interpolation, we put the sequences in the xy -coordinate system. Let denote Fibonacci point $p_n = (n, f_n)$ as the point from the n^{th} term of Fibonacci sequence. As example, the Fibonacci points for $n = 0$ until $n = 4$ are shown in Fig. 1.

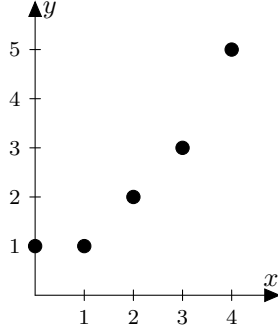


Fig. 1. Fibonacci points for $n \in \{0, 1, \dots, 4\}$.

In this paper, we define $\text{FLIP}_n(x)$ as the polynomial that generated using Lagrange Interpolation from p_i for $i \in \underline{n}$. In this case, we will interpolate from points $(x_i, y_i) = (i, f_i)$. Consequently, we have $x_i = i$ and $y_i = f_i$ in (5) and then we can write

$$\text{FLIP}_n(x) = \sum_{i=0}^n \left(f_i \prod_{j \neq i} \frac{x-j}{i-j} \right). \quad (6)$$

Surprisingly, (6) can be simplified because of $(i-j)$ factors. Let define $\Delta(n, i)$ as follows

$$\Delta(n, i) = \prod_{i \neq j} (i-j).$$

By the definition, $\Delta(n, i)$ is the product of n non-zero consecutive integers in which there are i positive integers. Therefore, we get

$$\Delta(n, i) = (-1)^{n-i} (n-i)! i! = (-1)^{n-i} \frac{n!}{\binom{n}{i}}. \quad (7)$$

By substituting (7) to (6), we have

$$\text{FLIP}_n(x) = \frac{1}{n!} \sum_{i=0}^n \left((-1)^{n-i} f_i \binom{n}{i} \prod_{j \neq i} (x-j) \right). \quad (8)$$

As example, from (8) we can derive $\text{FLIP}_n(x)$ for $n = 1, 2, 3, 4$ as follows

$$\begin{aligned} \text{FLIP}_1(x) &= 1 \\ \text{FLIP}_2(x) &= \frac{x^2 - x + 2}{2} \\ \text{FLIP}_3(x) &= \frac{-x^3 + 6x^2 - 5x + 6}{6} \\ \text{FLIP}_4(x) &= \frac{x^4 - 8x^3 + 23x^2 - 16x + 12}{12} \end{aligned}$$

The graphics of above polynomials are shown in Fig. 2.

We should remember that $\text{FLIP}_n(i) = f_i$ for $i \in \underline{n}$. Before deriving the other formulas of $\text{FLIP}_n(x)$, we will prove a theorem about its leading coefficient.

Theorem 1: The leading coefficient of $\text{FLIP}_n(x)$ is $\frac{(-1)^n f_{n-2}}{n!}$.

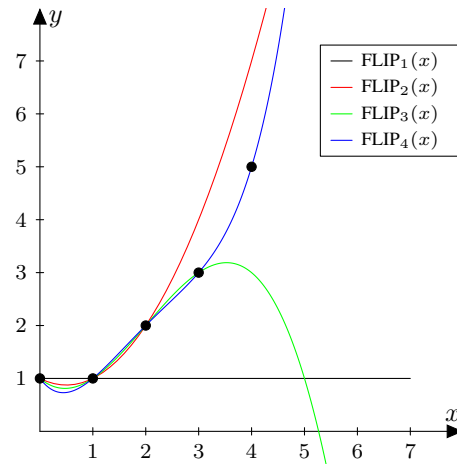


Fig. 2. $\text{FLIP}_n(x)$ for $n \in \{1, 2, 3, 4\}$.

Proof: The leading coefficient of $\text{FLIP}_n(x)$ is equal to $\frac{1}{n!} \sum_{i=0}^n (-1)^{n-i} f_i \binom{n}{i}$. Hence, we only need showing $\sum_{i=0}^n (-1)^{n-i} f_i \binom{n}{i} = (-1)^n f_{n-2}$.

$$\begin{aligned} \text{LHS} &= \frac{1}{\sqrt{5}} \sum_{i=0}^n (-1)^{n-i} [\varphi^{i+1} - v^{i+1}] \binom{n}{i} \\ &= \frac{1}{\sqrt{5}} \left(\varphi \sum_{i=0}^n (-1)^{n-i} \varphi^i \binom{n}{i} - v \sum_{i=0}^n (-1)^{n-i} v^i \binom{n}{i} \right) \\ &= \frac{1}{\sqrt{5}} [\varphi(\varphi-1)^n - v(v-1)^n] \quad (\text{identity (1)}) \\ &= \frac{1}{\sqrt{5}} [(\varphi^2 - \varphi)(\varphi-1)^{n-1} - (v^2 - v)(v-1)^{n-1}] \\ &= \frac{1}{\sqrt{5}} [(-v)^{n-1} - (-\varphi)^{n-1}] \\ &= \frac{1}{\sqrt{5}} (-1)^n [\varphi^{n-1} - v^{n-1}] \\ &= (-1)^n f_{n-2} \\ &= \text{RHS} \end{aligned}$$

■

A. Recursive and Implicit Formulas

In this part, we will derive the other formulas of $\text{FLIP}_n(x)$. It is quite impressive that $\text{FLIP}_n(x)$ can be generated recursively and implicitly, like Fibonacci sequence.

Let us consider $\text{FLIP}_{n+1}(x)$ and $\text{FLIP}_n(x)$. Each $i \in \underline{n}$ satisfies $\text{FLIP}_{n+1}(i) - \text{FLIP}_n(i) = 0$, or in other words, $\text{FLIP}_{n+1}(x)$ and $\text{FLIP}_n(x)$ pass through the $n+1$ same points. Consequently, we can write a relation

$$\text{FLIP}_{n+1}(x) - \text{FLIP}_n(x) = ax(x-1)\dots(x-n),$$

where a is the leading coefficient of $\text{FLIP}_{n+1}(x)$. By Theorem 1, we have $a = \frac{(-1)^{n+1} f_{n-1}}{(n+1)!}$. Therefore, we have a recursive formula

$$\text{FLIP}_{n+1}(x) = \text{FLIP}_n(x) + \frac{(-1)^{n+1} f_{n-1}}{(n+1)!} P_n(x) \quad (9)$$

where $P_n(x) = x(x-1)\dots(x-n)$. By rewrite recursive formula of (9) for $\text{FLIP}_n(x), \text{FLIP}_{n-1}(x), \dots, \text{FLIP}_2(x)$ we obtain the implicit formula as follows

$$\begin{aligned}\text{FLIP}_{n+1}(x) &= \text{FLIP}_1(x) + \sum_{i=1}^n \frac{(-1)^{i+1} f_{i-1}}{(i+1)!} P_i(x) \\ &= 1 + \sum_{i=1}^n \frac{(-1)^{i+1} f_{i-1}}{(i+1)!} P_i(x)\end{aligned}$$

IV. CONCLUSIONS

This paper has introduced a new method to construct polynomials based on Fibonacci sequence using Lagrange Interpolation. We already prove that those polynomials can be generated recursively and implicitly.

REFERENCES

- [1] T. Scott and P. Marketos, "On the origin of Fibonacci sequence," *MacTutor History of Mathematics*, 2014, also available at www-groups.dcs.st-and.ac.uk/history/Publications/fibonacci.pdf.
- [2] A. Benjamin and S. Plott, "A combinatorial approach to fibonomial coefficients," *Fibonacci Quarterly*, vol. 46/47, no. 1, 2009.
- [3] T. Amdeberhan, X. Chen, V. Moll, and B. Sagan, "Generalized Fibonacci polynomials and fibonomial coefficients," *Annals of Combinatorics*, vol. 18, no. 4, pp. 541–562, 2014.
- [4] D. Garth, D. Mills, and P. Mitchell, "Polynomials generated by the Fibonacci sequence," *Journal of Integer Sequences*, vol. 10, no. 2, p. 3, 2007.
- [5] J. Kiusalaas, *Numerical methods in engineering with MATLAB®*. Cambridge University Press, 2010.