IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

ADAPTATION OF NEWTON-GREGORY FORWARD INTERPOLATION METHOD IN PYTHON IDLE 3.10 FOR THE PLANAR MOTION ANALYSIS OF 1-D BAR ELEMENT OF UNIT MASS

[1]Chandramouli Kotta Kota,[2]Kshitijarun Bidari
[1]Ph.D Scholar at NIT, Andhra Pradesh,[2]Intern at AnalytixLabs
[1]Department of Mathematics
[1]AnalytixLabs, Bangalore, India

Abstract: The field of Theoretical Mathematics has a number of methods and techniques to obtain specific parameters for a given set of data that relates to frequency, time, acceleration and velocity. Newton-Gregory Forward Interpolation Method is one such technique that can be applied for the real-time analysis of motion of physical objects through mathematical modeling. The physical motion of objects involves trajectory path analysis, projectile motion analysis that can relate to the jet plane taking off or a rocket being shot or the motion of a stone or rod in a specific projectile path. The motion of the particle or member can be for a given period of time with certain parameters like positions known and the corresponding velocity and acceleration can be obtained.
This novel research idea involves performing the motion analysis of a physical member or particle that moves in a circular path for a specific period of time using Newton-Gregory Forward Interpolation Method and validating the same through the software application Python IDLE 3.10.
The study basically involves giving the input as time periods and the corresponding position of the member to obtain the velocity and accelerations. The same parameters are given as input to the python and the required mathematical code is developed to be executed to obtain the solution.

Index Terms **– Newton-Gregory Forward Interpolation Formula, Python 3.10 IDLE, Motion of Objects.**

I. INTRODUCTION

The modern world is experiencing a huge burst of data which if left un-analyzed can lead to detrimental losses to the environment and life. The field of data science makes use of mathematical models which, when applied to software tools like Python, R, Java, C can give highly beneficial results that can help increase the efficiency of performance of a system by leveraging the data. This research paper involves applying the Newton-Gregory Forward Interpolation technique to create a mathematical model to analyze a given set of data by the classical method and by Python 3.10 IDLE, thereby obtaining the values for certain unknown parameters. There are a number of tools that can analyze data in the form of known values to obtain new unknown parameters, but this research uses Python 3.10 software application to apply the mathematical model, interpolate it and obtain the unknown data.

II. CONCEPT BEHIND THE RESEARCH

The field of data science basically deals with what conclusions we can arrive at by a given set of data. Analyzing the data can be done in any field such as engineering, business, mathematics, aeronautics, space, geography wherein a conclusion needs to be arrived at by visualizing the same. Fields of engineering and economics will require the use of mathematical models built by the help of existing or new data to obtain the unknown parameters for analysis of concepts like motion, flow direction, selling, growth rate, increasing profits, leveraging businesses and enhancing performances.

This research works on similar lines of obtaining unknown parameters by building a mathematical model around a given data set and finding out the unknown parameters velocity and acceleration of a given member that rotates in a plane.

The field of aeronautics, space and mechanical engineering see a lot of conditions where it is required to obtain or find certain data or parameters that are unknown by utilizing a given set of known data. For, example reciprocating motion of a piston, rotation of a crankshaft, rotary motion of turbine blades are required to be analyzed for their positions, velocities and accelerations to understand their behavior. This can help increase the performance efficiency of the machine or device.

In this research we aim at analysis one such motion of a member in a circular path to obtain is velocity and acceleration for a given position at a given period of time. A set of data is analyzed and implemented in an interpolation method to obtain certain unknown parameters.

III. NEED FOR THE ANALYSIS OF MOTION OF OBJECTS

Real world problems that involve rotating and reciprocating members need to be analyzed for the displacements, velocities and accelerations for developing a safe operating system. A system not designed according to the safety standards can result in causing highly catastrophic damages to life and property. Hence analyzing and designing a system stands as a matter of huge concern.

The IC Engine Piston reciprocates at varying speeds and thus rotates the crank rod at similar angular velocities that, if not analyzed can result in detrimental situations causing fracture of the crankshaft, crank rod or connecting rod when it absorbs the huge impact of the combustion from the piston. This research paper aims at performing such an analysis of a machine component (1-D rod member) that takes into consideration the different time periods and angular positions of the member.

IV. PROBLEM DEFINITION

Fig.1 – Crank Rod in an IC Engine

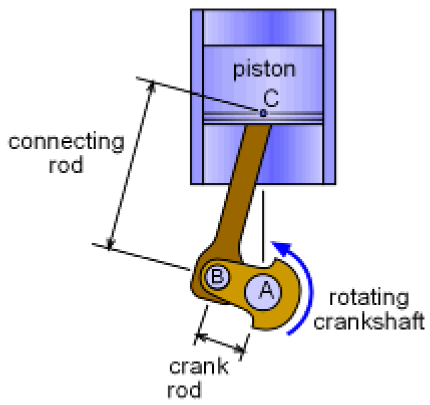
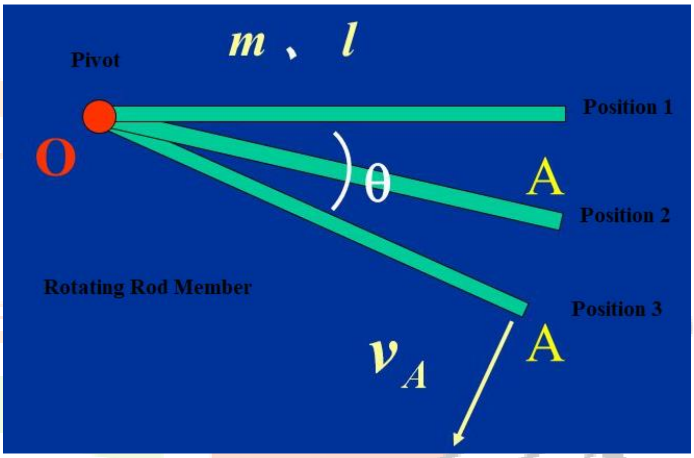


Fig.2 – Crank Rod Modeled as a 1D Bar Element to analyze rotational motion



The crank rod rotates at varying velocities based on how the acceleration is given to the vehicle according to the needs of the driver. Designing the crank rod without analyzing the velocities and accelerations can lead to failure of the same.
The crank rod is modeled here as a 1-D rod element of unit mass and unit length and the analysis is done. The problem involves finding the velocity and acceleration of given specific displacements/positions of a rod member(crank rod) that rotates around a fixed point. The problem definition goes as follows:

A one-dimensional rod is rotating in a 2-Dimensional plane. The table below gives the angle of rotation ϴ in radians which the rod has turned in different time periods denoted as "t". The angular velocity and Angular Acceleration at different intervals has to be obtained.

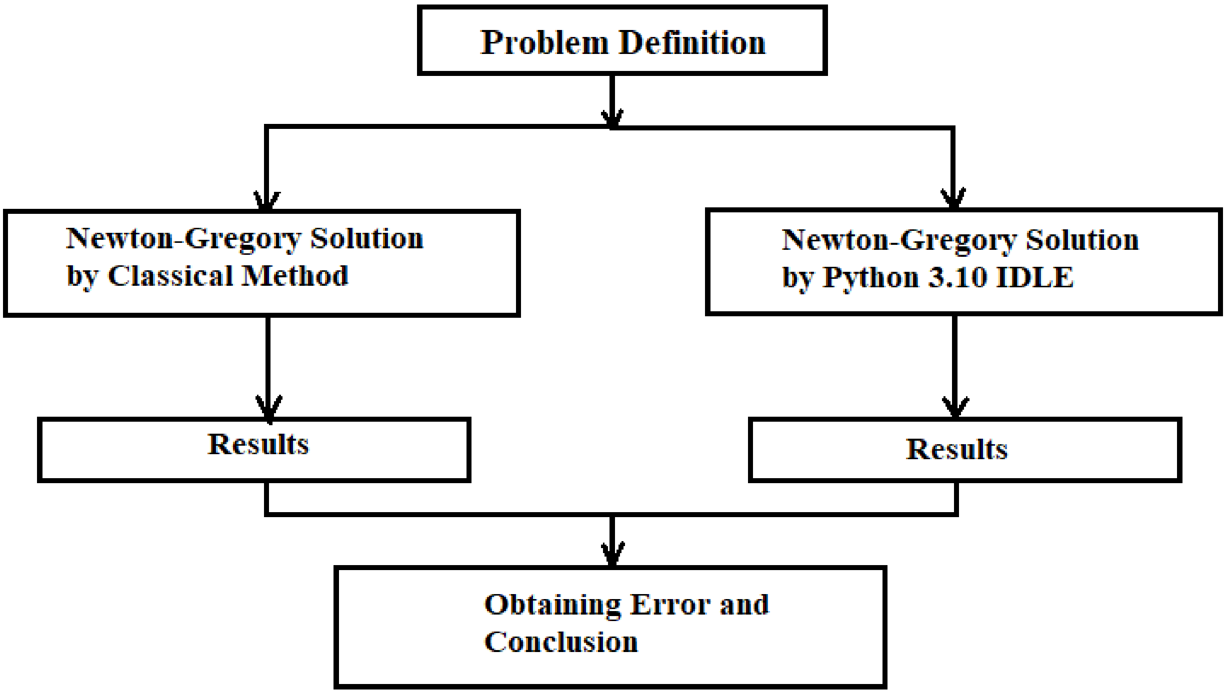
time = t	**0**	**0.2**	**0.4**	**0.6**	**0.8**	**1**	**1.2**
Angle = ϴ	**0**	**0.12**	**0.49**	**1.12**	**2.02**	**3.2**	**4.67**

V. RESEARCH METHODOLOGY

The methodology consists of 2 parts, initial part being to obtain the solution of the above mentioned problem by using the classical method wherein the Newton-Gregory Forward Interpolation formula is applied to the data to obtain the velocities and accelerations at different positions of the member. The Newton-Gregory Interpolation Formula interpolates over the whole set of data (i.e., position points) and by the use of differentiation principles obtains the velocities and accelerations.

The second part of the research study involves applying the same to a programming language Python 3.10 IDLE. Here the Python code is developed for the same conditions and data set above. The code is then executed which yields the values of the required velocities and accelerations. The Python code is developed in its core format without utilizing any external libraries for the mathematical calculations. All the computations are performed by the core python code fed to the system.
After the results of both the methods are obtained they are compared and the error is found out to check the amount of corroboration between them.

Figure.3 - Flow chart indicating the methodology of the research study



IV. MATHEMATICAL SOLUTION USING NEWTON-GREGORY INTERPOLATION FORMULA

The mathematical method and solution developed by Newton is used for this analysis wherein the multiple time and positional points are given as the input values and subsequently the forward interpolation table is developed by performing the iterations as developed in the generalized method of forward interpolation. The iterated values of $\Delta^2\Theta$, $\Delta^3\Theta$, $\Delta^4\Theta$, $\Delta^5\Theta$, $\Delta^6\Theta$ are obtained by using the stepwise obtained solutions in the method.

The Newton-Gregory Forward Interpolation method makes use of differentiation to obtain the angular velocity and angular acceleration. The same method will be repeated for obtaining the values of velocity and acceleration at different points.

The major advantage of utilizing this method is that it doesn't require the equation of the displacement curve to obtain the velocity and acceleration as required by the conventional method of differentiation.

The added part in this method is the forward/backward difference table which enables to obtain the iterated parameters as mentioned above.

V. THE APPLICATION OF PYTHON LANGUAGE FOR THE PROBLEM

Python is one of the most preferred languages for obtaining solutions to mathematical problems of various kinds. The problem at discussion here involves the analysis of rotary motion of a member in a 2-Dimensional plane so Python being a language that is comparatively easy to use and develop code, is being used. The Python language comes with various libraries and packages that can help solve complex mathematical problems in a considerably short period of time. Considering other languages for the development of the code would prove to be a tedious task.

VI. DEVELOPMENT OF THE PYTHON CODE

The Python Code for the above problem was developed in its core-programming format without the use of any libraries or packages for the solution.

Input For the Code:

The code involved taking the input parameters as the different time and positional points. The basic integer input command is used to take input from the user in terms of seconds for time and radians for rotational position.

Body of the code:

The main body of the code involves, in the first part, building the forward difference table with the respective iterated values of $\Delta^2\Theta$, $\Delta^3\Theta$, $\Delta^4\Theta$, $\Delta^5\Theta$, $\Delta^6\Theta$ being obtained sequentially to help find the values of the needed velocity and acceleration further on.

The second part involves writing the code for the main solutions for angular velocity and angular acceleration wherein the inputs for these will be the previously obtained iterated values of the (delta parameters) $\Delta^2\Theta$, $\Delta^3\Theta$, $\Delta^4\Theta$, $\Delta^5\Theta$, $\Delta^6\Theta$. With this as the base the python formula code calculates the angular velocity and acceleration.

VII. RESULTS OBTAINED THROUGH CLASSICAL METHOD

After following the steps as mentioned in the previous head for the other values of time and position, the different angular velocities and accelerations obtained are as listed below:

Table.1 – Shows the Velocity and Acceleration values as obtained by the Classical Method

Sl.No	Time(t)	Displacement(ϴ)	Velocity	Acceleration
1	0	0	0	0
2	0.2	0.12	1.3	6.3
3	0.4	0.49	2.5	6.5
4	0.6	1.12	3.82	6.75
5	0.8	2.02	5.2	7
6	1	3.2	6.62	7.25
7	1.2	4.67	8.1	7.5

VIII. RESULTS OBTAINED THROUGH PYTHON METHOD

The results obtained by the Python 3.10 IDLE method are as follows: -

Table.2 – Shows the Velocity and Acceleration values as obtained by the Python Method

Sl.No	Time(t)	Displacement(ϴ)	Velocity	Acceleration
1	0	0	0	0
2	0.2	0.12	1.216667	6.24999
3	0.4	0.49	2.491667	6.5000
4	0.6	1.12	3.816667	6.7500
5	0.8	2.02	5.191667	7.0000
6	1	3.2	6.616667	7.2500
7	1.2	4.67	8.091667	7.5000

It is observed in the above results that the values of angular velocities and accelerations are almost close to the values as obtained by the classical method. The advantage being that the values are obtained in precise decimal values thus reducing the chances of error.

IX. CONCLUSION

The solution is obtained by both the methods, thus obtaining the angular velocity and acceleration of the motion of a particle at given specific time periods. The comparison of both the methods shows that the difference is negligible and hence both the methods corroborate each other. The average error is seen to be a mere 1.62 percent for the velocity and 0.72 percent for the acceleration, hence it can be concluded that the analysis method applied is acceptable.

Sl.No	Methods				% Error	
	Classical Method		Python 3.10 IDLE			
	Velocity	Acceleration	Velocity	Acceleration	Velocity	Acceleration
1	0	0	0	0	0	0
2	1.3	6.3	1.216667	6.24999	8.33%	5.001%
3	2.5	6.5	2.491667	6.5000	0.8%	0
4	3.82	6.75	3.816667	6.7500	0.3%	0
5	5.2	7	5.191667	7.0000	0.8%	0
6	6.62	7.25	6.616667	7.2500	0.3%	0
7	8.1	7.5	8.091667	7.5000	0.8%	0
					Avg =1.62%	**Avg = 0.72%**

Table.3 – Shows the Error of the Velocity and Acceleration values as obtained by the Classical and Python Method

X. SCOPE

The scope of this study and research is that it can be applied to any member that exhibits uniform or no-uniform circular motion around a fixed point where the velocities and accelerations need to be found out owing to the non-availability of large datasets. In a sense that if there is no large amount of data available on certain motion behavior and only a small set of observations are available, then that data can be used to interpolate and obtain the approximate values of further unknown observations. And more over the research also shows that the Newton-Gregory Forward Interpolation method gives accurate results when applied on the software application PYTHON 3.10 IDLE also.

REFERENCES

[1] Advanced Engineering Mathematics – Erwin Kreyszig**,** 10thEdition, Wiley, India.

[2] Advanced Engineering Mathematics – K. A. Stroud, 5thEdition, Industrial Press, Inc., US.

[3] Design of Flange Coupling on the basis of Strength and Rigidity using Classical Approach and Python Method, in ICMAPE, London, 4th and 5th Feb, 2022, Universal Paper ID – SP-ICMAPE-LNDN-040222-1903

[4] Design of Flange Coupling using Classical Approach and Python 3, ICMMRE, Dallas, USA, 5th & 6th Feb, 2022, Paper ID – WRS-ICMMRE-DALS-050222-962

[5] Design of Flange Coupling using Python IDLE 3.10, ICMMRE, 2nd &3rd Feb, 2022, Reykjavik, Iceland, Paper ID – WRS-MMREREYK – 020222-956

[6] Design of Flange Coupling using Classical Approach and Python 3.10 IDLE, ICRIET, San Francisco, 14th Feb, 2022.