Developing a Mathematical Mobile App: A Case Study of an Environmental Model

Emetere M. E., Oluwafemi T. J., Akinlusi T., Allen M. C., Ama E. C.

Abstract—**Developing application that work on environmental models are quite complex because it parameters have high uncertainties. In this project, the ultimate goal is to design a comprehensive module which would simplify the formation of environmental mobile applications. It is expected that the applications should run through the source codes i.e. to fulfill salient conditions so that it could be incorporated into portable mobile device. The success of this project is both lucrative and informative. Above all, it provides a framework for writing environmental mobile applications.**

Index Terms—**Mobile application, Environmental model, Mathematics, developing source codes**

I. Introduction

THE development of framework that simplify the initial processes of writing a mobile environmental application is the objective of this paper. There are over 390, 000 mobile applications in the market. Mobile applications contain several thousand lines of source code which are written in various programming languages. However, the major advantage of a particular language is its potency to make the various pieces of the Web app puzzle come together. This challenge maybe very frustrating in practical sense. Hence, there is the need to there is the need to seek-out ways of simplifying the rigorous thousand lines of source code.

In this paper, we considered a typical concept i.e. the environmental model for determining aerosols loading either indoor outdoor (in the atmosphere). The natural disasters around the globe are tied to negligence of environmental models which is expressed using mathematical principles. The world is gradually becoming too large and busy for meteorological centers and media houses to be solely saddled with the responsibility of creating awareness of impending environmental dangers. Since weather is dynamic in itself, it is expected that multiple assumptions which is tied to complex meteorological outcomes could only be adequately represented using mathematical expressions. In other words, in a typical environmental model proposed by Emetere et al. (1-6), we expect to have above a million lines of source codes. This is problematic because there will be lots of pieces of Web app puzzles to corroborate. In this thesis, the use of mathematical Web puzzles shall be developed to make life easier for developers in the area of conceiving, designing, and implementing the different pieces of the larger Web app puzzles created.

Mathematical web puzzle emanates from mathematical methods. Mathematical method is a systematic, logical and procedural way of displaying mathematical processes. In computing, it is represented as a function or procedure that is used as an interface with an object-oriented programming language. It is frequently used by hardware vendors, such as IBM, Intel and AMD to programme. Mathematical techniques that is used in design, implementation and testing of computer systems are known as Formal Methods (7). For example, hardware vendors adopts the formal method (FM) to undertake parameterized verification of cache coherent protocol, processor execution engine validation, optimization of system functionality, verification of high performance dual-port gigabit Ethernet controller, verification of power gates probing the Cadence technology and troubleshooting hardware protocols.

Recently, app developers have argued that mathematics is not so needed in software engineering (8) since it adopts techniques like; analyzing the process, designing the process, writing codes during the process, testing the process and maintaining the process. In this paper we adopted the Python language to resolve basic problem in environmental mobile application.

II. Choosing the Programming Language for this Research

In this section, the different programming environments that is available for mobile platforms shall be discussed to justify the choice of the current language used for this research. Most android developers are used to the traditional Java. However, the need for expansion of the mobile platforms have brought on board notable programming environment. For example, Microsoft brought C# and .NET invention. Google Labs came-up with the App Inventor for programming apps for Android. Most of this language has wide application to over fifty mobile applications (9). The emergence of the Python for Android 1.2 documentation displayed the flexibility of Python for mobile platforms. Due to the unique requirements and constraints of mobile platforms, the repackaging of Python moved onto the Mobile-SIG. Mobile-SIG exists to improve the usability of

Manuscript received June 29, 2016; revised July 08, 2016. This work was supported in part by Covenant University, Ota, Nigeria.

M. E. Emetere is with the Department of Physics, Covenant University, Canaan Land, Ota Nigeria (+2348035267598: moses.emetere@covenantuniversity.edu.ng).

T. J. Oluwafemi, is with the Department of Physics, Covenant University, Canaan Land, Ota Nigeria.

T. Akinlusi is with the Department of Physics, Covenant University, Canaan Land, Ota Nigeria.

M.C. Allen, is with the Department of Physics, Covenant University, Canaan Land, Ota Nigeria.

E.C. Ama, is with the Department of Physics, Covenant University, Canaan Land, Ota Nigeria

Python on mobile devices (10). However, SSL libraries of Python are out of date and may not be update easily. Kivy - Open source Python library was also developed to further ensure a rapid development of applications that make use of innovative user interfaces, such as multi-touch apps. Therefore, what is the prospect of Python in recent times? Figure 1 shows the comparative and competitive programming environment and their performance.

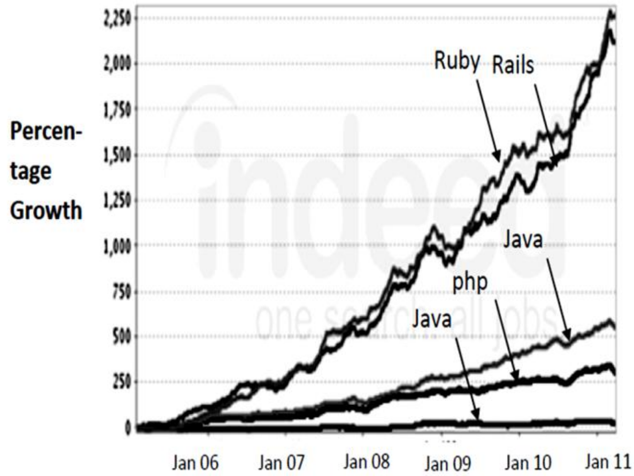

Fig1. Relevance of Python to advance recent research

III. Mobile Device and Its Embedded Mathematical Reality

Mobile devices work on the applications used by technological devices and social apps (such as twitter and Facebook), gaming apps (such as subway surfers) which are direct influences of compiled codes that are mathematical oriented. Some codes used by game apps use complex codes that require the use of numbers, analytical and logical thinking or reasoning to arrive at. There is not one software ever created that was not produced based on analytic thinking and logical reasoning. It takes a lot of time and calculations to come up with this source codes (11, 12). Since, environmental model are quite complex and comprehensive, it is essential for software developer to have a good background in the mathematical field since this aids logical and analytical reasoning (5). Recent mobile devices are designed using the QR codes which are made-up of matrix or two-dimensional barcode. QR code uses four standardized encoding modes i.e. numeric, alphanumeric, byte/binary, and kanji to efficiently store data. The efficiency of the QR codes in mobile device qualifies it to host regional meteorological data to ease the running of environmental models like any other apps on the mobile device. The mobile device would be the best medium to create environmental awareness to the teeming population. Some of the causalities from sudden natural disaster could have been averted if this concept exists.

The cost of development of this proposed product is dependent on the amount of errors made in the object code. In order for the errors made to be reduced to the minimum level, the codes used are checked severally. We recognize that huge thinking should be incorporated into this proposed product because errors made during software systems development could have grave consequences. Huge loss in terms of money could occur; also client may be endangered by false alarm due to such errors. However, its success could be very lucrative and informative.

Formal methods have been proven to help achieve the required level of safety (7). Therefore mathematical environmental model could be incorporated for design, implementation of protocols, testing and control of software systems. The application of this mathematical methods or techniques to the creation and verification of software systems, as well as its importance to environmental information or prediction are labour intensive. Therefore it is not advisable to check all the desired properties and functions of a software program in a detailed manner. It is more cost and time effective to first of all determine the important properties of the software before choosing the type of environmental model to embark upon.

It is important to note that the size of the codes is very important to curb error. Environmental models may be intensive with lots of iterative procedure (2-5). The size of these developed codes should be reduced to its minimum, for greater compatibility and faster compilation. This risk can be reduced using the JSZap. It transforms source codes into three streams: AST production rules, identifiers, and literals, each of which is compressed independently. Another way of writing sizeable codes is familiarizing oneself with different mathematical methods to help in understanding, thinking and working method. Also, originality, authenticity, and efficiency in problem solving are important when it comes to the use of codes. These days it is observed that a lot of physics methods and analysis is put into the development of efficient codes. This physical methods no doubt have their roots deep down in mathematics. Therefore one of the sensitive functions of mathematical methods includes error management, estimation and rounding off, softcopies of mathematical representation and easy learning (13).

Therefore, it is salient to ask the role of the mathematical representation and the environmental model that will be adopted. Here are various roles mathematical methods play in the formation of environmental models.

a. Basis for Computing: they are like the basis for computing. When it comes to environmental modelling, mathematical methods play a vital role as it were. Also, it is a strong foundation and anchor for the design, structuring, development, accuracy and display of various finding in environmental exploration.

b. Accuracy and Precision: Environmental models sometimes require the auto piloting system which needs a lot of programming and mathematical accuracy in controlling the re-ordering of reiterative protocols. Here, principles in vector analysis, probability and some part of calculus come into full play.

c. Error Analysis and Reduction: Numerical analyses in mathematical methods take credit for this role. Here, the round-off and round-down processes in calculations help to check errors and then reduce it to an insignificant deficit with 100% accuracy which occurs in rare cases.

d. Calibration of Industrial Equipments: Several findings have shown that recent environmental hazard was caused by inadequate calibration of measuring instruments [5-7]. This may be due to external factors such as factory errors or environmental agents. Ultimately, the measuring equipment loses their accuracy and precision, and ultimately these errors lead to in accurate results which on a large scale would cause a lot of damage to scientists. In this case, a well developed mathematical methods such as numerical analysis and complex function analysis can be employed to rectify such errors.

IV. Developing the Microarchitecture for Mathematical Web Puzzles

In this section, a simple illustration shall be made using the extract from an environmental model i.e. an extension of the dispersion model in Ref [1]

$$\psi(\lambda) = {a_1}^2 \cos\left(\frac{\mathrm{n}_1 \pi \tau(\lambda)}{\mathrm{k_y}} + \alpha\right) \cos\left(\frac{\mathrm{n}_1 \pi \tau(\lambda)}{\mathrm{k_z}} + \alpha\right) + {a_2}^2 \cos\left(\frac{\mathrm{n}_2 \pi \tau(\lambda)}{\mathrm{k_y}} + \beta\right) \cos\left(\frac{\mathrm{n}_2 \pi \tau(\lambda)}{\mathrm{k_z}} + \beta\right) + \cdots \dots \dots \dots \dots {a_n}^2 \cos\left(\frac{\mathrm{n_n} \pi \tau(\lambda)}{\mathrm{k_y}} + \beta\right) \cos\left(\frac{\mathrm{n_n} \pi \tau(\lambda)}{\mathrm{k_z}} + \beta\right) \quad (1)$$

Here α and β are the phase differences, k is the diffusivity, τ is the AOD, ψ is the concentration of contaminant, λ is the wavelength, a and n are atmospheric and tuning constants respectively. Each term in the equation (1) has its perculiar condition for its existence. Aside each term with its atmospheric conditions, the adaptation between geographical locations may increase the nth term of the equation. The AOD is resolve in each of the terms.

$$\tau = \frac{8.5 \times (1 + R_1)C}{1.5 \times RH} + \frac{8.5 \times (1 + R_2)C}{3 \times RH} + \frac{0.3 \times (1 + R_3)C}{6 \times RH} + \frac{1 \times (1 + R_4)C}{12 \times RH} + \frac{1.5 \times (1 + R_5)C}{24 \times RH} + \frac{2.5 \times (1 + R_6)C}{48 \times RH} + \frac{6 \times (1 + R_7)C}{96 \times RH} + \frac{9 \times (1 + R_8)C}{192 \times RH} + \frac{15 \times (1 + R_9)C}{384 \times RH} + \frac{60 \times (1 + R_{10})C}{768 \times RH} + \frac{2000 \times (1 + R_{11})C}{1536 \times RH} + \frac{15000 \times (1 + R_{12})C}{3072 \times RH} \quad (2)$$

Here, R is the magnitude of rain fall, RH is the relative humidity and C is the cloud cover. The numeral in equation (2) varies from one region to another.

The computer architecture in this command entails instruction set architecture (ISA), microarchitecture and system design. We are more interested in the microarchitecture. The traditional microarchitecture to combine equation (1) i.e. Figure 2a and equation (2) i.e. Figure 2b.

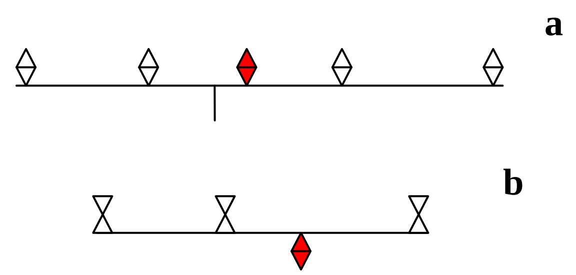

Fig 2. Traditional architecture

To obtain the proposed mathematical web puzzle, we adopted Batten (14) microarchitecture frame i.e. general-purpose graphics processing unit (GPGPU) in Figure 3.

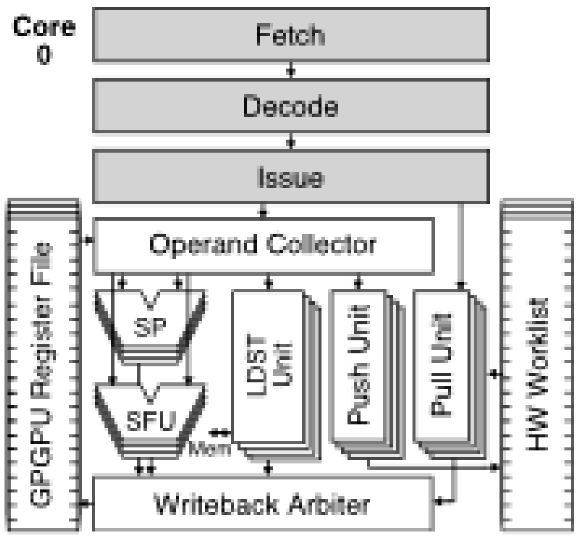

Fig 3. General-purpose graphics processing unit

GPGPU was used to accelerate execute irregular algorithms (due to meteorological dynamism) that read and modify graph-based data structures and dynamically generate additional parallelism. The HW work list in Figure 3 was modified to accommodate equation (2). The chaos generator module was incorporated to enable us input sets of Python commands to control the uncertainties as it relates to random signals with 256 possible states.

V. Conclusion

In conclusion, though the project discussed in this paper is still ongoing. It is imperative to state that the development of environmental mobile platform based on the Python is unequaled. The success of this project would be majorly to adequately disseminate environmental alerts without

ISBN: 978-988-14047-1-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

necessarily relying on public alert system. Hence, the project is both lucrative and informative if all mentioned procedures are carefully executed.

ACKNOWLEDGMENT

The authors appreciate the partial sponsorship of Covenant University.

REFERENCES

[1] M. E. Emetere, M. L. Akinyemi, and O. Akinojo. (2015). Parametric retrieval model for estimating aerosol size distribution via the AERONET, LAGOS station, *Environmental Pollution*, 207. pp. 381-390.
[2] M. E. Emetere. (2013). Modeling Of Particulate Radionuclide Dispersion And Deposition From A Cement Factory. Annals of Environmental Science,7(6). pp. 71-77
[3] M. E. Emetere. (2014). Forecasting Hydrological Disaster Using Environmental Thermographic Modeling. Advances in Meteorology 2014, pp. 783718.
[4] M. E. Emetere. (2016). Statistical Examination Of The Aerosols Loading Over Mubi-Nigeria: The Satellite Observation Analysis, Geographica Panonica, 20(1), pp. 42-50.
[5] M. E. Emetere., M. L. Akinyemi, and T. E. Oladimeji. (2016). A Conceptual Mathematical Examination Of The Aerosols Loading Over Abuja-Nigeria. Communications in Mathematics and Applications, 7(1). pp. 47–54.
[6] M. E. Emetere. (2014). Volcanic Eruption Trends in the Five-Years Pre-Eruption Era. Journal of Volcanology and Seismology, 8 (6), pp. 411–417.
[7] P. Cousot, R. Cousot, J. Feret, L. Mauborgne, A. Mine, and X. Rival. (2009). Why does Astee scale up? Formal Methods in System Design, 35, pp. 229-264.
[8] K. Devlin, (2001). The Real Reason Why Software Engineers Need Math, Communications of the ACM, 44, pp. 21-22..
[9] Unknown, *Mobile application development*: https://en.wikipedia.org/wiki/Mobile_application_development (Retrieved August 2016).
[10] Unknown, *Python Mobile SIG*: https://www.python.org/community/sigs/current/mobile-sig/ (Retrieved August 2016).
[11] R . L. Glass. (2000), A New Answer to 'How Important is Mathematics to the Software Practitioner?, IEEE Software, pp. 135-136.
[12] P. Henderson, et. al., (2001). Striving for Mathematical Thinking, SIGCSE Bulletin - Inroads, 33, pp. 114-124.
[13] T. Lethbridge. (2000). What Knowledge is Important to a Software Professional?, IEEE Computer, 33, pp. 44–50.
[14] Christopher Batten, *Current Research Projects*: http://www.csl.cornell.edu/~cbatten/research.html (Retrieved August 2016).
[15] W. D. Doyle, "Magnetization reversal in films with biaxial anisotropy," in *1987 Proc. INTERMAG Conf.*, pp. 2.2-1–2.2-6.
[16] G. W. Juette and L. E. Zeffanella, "Radio noise currents n short sections on bundle conductors (Presented Conference Paper style)," presented at the IEEE Summer power Meeting, Dallas, TX, Jun. 22–27, 1990, Paper 90 SM 690-0 PWRS.