Research Article

i-Proclaim .my

Asia Pac. j. energy environ.

A Commentary on the Applications of Python in Resolving Issues Concerning Energy and the Ecosystems

Takudzwa Fadziso

Institute of Lifelong Learning and Development Studies, Chinhoyi University of Technology, ZIMBABWE

*Email for Correspondence: fadziso.tak@gmail.com

Manuscript Received: 09 July 2021 - Revised: 17 August 2021 - Accepted: 27 August 2021

Abstract

The energy industry is just getting started with applying it to problems with energy and eco-systems so they can find solutions. Python's popularity has increased across a variety of sectors, including businesses, academic institutions, government agencies, and research organizations. The true potential it possesses to automate a variety of processes while simultaneously increasing the capabilities of various industries to predict outcomes has been observed. Because of the digital transformation, such as sensors and high-performance computing services, which enable artificial intelligence (AI), machine learning (ML), big data acquisition, and storage in digital oilfields, its popularity has been on the rise in the industry that deals with energy and eco systems. This is one of the primary reasons why. This can be easily verified by conducting a quick search for the number of publications that have been produced in the field of energy and eco systems by the Society of Petroleum Engineers over the past few years. Without having to invest in pricey software, the production and reservoir engineers will be able to better manage the production operation thanks to this development. In addition to this, it will lead to a decrease in the overall operating costs and an increase in revenue. As a result, it has been demonstrated to be a promising application that has the potential to bring about a revolutionary change in the industry of energy and eco systems and to transform the features that are already in place for the purpose of resolving issues related to energy and eco systems.

Keywords

Python, Gas Problems; Energy, Artificial intelligence (AI), Internet of things (IoT), Machine learning (ML)

Introduction

Python, an interpreted high-level, general-purpose programming language, is used for constructing intelligent models that anticipate, diagnose, and assess reservoir and well performance in the energy and ecosystems industry. Its clean, readable code makes it a programming language. Its open-source license allows multiple uses (Achar, 2015). Automation saves compilation and time, optimizing labor. It powers algorithm research, prototyping, and early deployment. Data management, visualization, and debugging are significant points of this engineering computing/coding program (Amin, 2019). It offers statistical, machine learning, and signal processing toolboxes and has significantly impacted technical computing and enabled technical programming with many libraries.

This system's performance was simulated using MATLAB and Python routines in Abaqus' interface. Simulations show that this wire rope can replace the rod string in a sucker rod pumping device (Achar, 2016). To analyze comprehensively, stress, load, and movement profiles were created. Kushkumbayeva et al. (2018) evaluated multistage stimulation technical assessment based on lithology, porosity, natural fractures, stimulation kind, infectivity, and production logging data using Python script (Hossain et al., 2015). Each well's Excel zonal injectivity shows pre- and post-stimulation injectivity related to the stimulation kind (acid wash, high-rate matrix, and acid fracturing). They plotted multivariable productivity analyses using Python's script to read Excel injectivity analysis data for all wells.

Maiorano et al. (2019) employed Python in an IAM to change injected salinity based on seawater-produced water mixing. Using Python scripts, Shoaib et al. (2019) passed the gas lift rate between the reservoir and network simulators. Hesar et al. (2018) showed that a Python script that integrated large-scale subsea system simulation is doable and necessary. This suggests that integrating the subsea clusters into a single model reflects reality better than conventional approaches and eliminates the need for significant simplifications or sub-modeling. Integration captures "system impacts" that would be hidden if units were modeled separately. A Python code communicates data between finite difference (FD) and finite element (FE) grids to study time-lapse, three-dimensional (4D) stress during production/injection [6]. Diakonova et al. (2019) proposed an optimization riser configuration that optimizes all parameters and is cost-effective. To respect project-specific weighted criteria, a Python script interfaced with Orcaflex software and user-defined parameters. Sarkar et al. (2018) developed a Python-based workflow automation framework using a cloud-based simulation platform.

The digital twin provides a single source of truth about asset conditions and enables data-driven communication and decision-making among operators, contractors, regulators, etc. Noshi et al. (2018) wrote Python code to find inflection points at the midpoint of curve turns using inclination and azimuth indices. Ophthalmology's retinal blood vessel tortuosity method inspired it. The tortuosity metric provided three danger categories for three index ranges. Operator drilling incidents and NPT were compared to the indexes. They explained the integrated simulation framework created on this technique and demonstrated its application to complex physical and chemical challenges. All simulation engines, linear solvers, healthy controls, interpolation engines, and state operator evaluators are implemented in C++11 and exposed to Python, combining Python's versatility with C++'s performance.

Hosseinimotlagh (2014) proved that the Python programming language possesses the flexibility and power necessary for effective reservoir management. It enhances the capabilities of simulators, which enables engineers to design flexible-control logic to address field management difficulties, and it empowers engineers to apply simulation in new ways. Achar (2018b) presented an innovative strategy for determining how to carry out these activities more efficiently and at a lower cost to maximize oil recovery. An actual H&P gas-injection pilot horizontal well in the Eagle Ford Shale is used to demonstrate the procedure. The method is used to match the performance of the well, which is then used to explain the procedure. Python, a free and open-source programming language, was used to write the code. Aipa et al. (2015) computed a friction data table using the Colebrook-White Equation and the Weymouth Friction Factor equation. This was accomplished through the use of the programming language Python.

The Weymouth friction factor was updated to include a correction factor considering varying degrees of pipe roughness. In addition, the new friction factor connection was applied to the classic Weymouth equation to make certain adjustments Python and Excel were used by Velmurugan and Radhakrishnan (2018) to perform the computations necessary to determine all of the dimensionless pressures for the various well designs. Research is conducted on the dimensionless pressure derivatives of a vertical oil well to find the best possible site for the well that will allow for good oil production while minimizing the impact of any external boundaries that may come into play too soon. Achar (2018a) examined the typical workflow that governs probabilistic evaluation methodologies and proposed a Python script-based approach. This approach enables the user to run a quick and easy mineral components evaluation based on porosity and raw input logs. Achar (2018b) also analyzed the typical workflow that governs probabilistic evaluation methodologies. The workflow is evaluated using data from an average well in the Niger Delta. The results are compared to those of a deterministic evaluation to see whether or not there are any additional benefits. They researched a mature waterflood field with over 100 active producers, and a history of water injection spanned over 15 years. Python programming was utilized to organize and incorporate several data sources into a unified graphical dashboard. A workflow for creating reservoir fluid from data logs and PVT databases was created by Al Mahmud (2012). The workflow consists of two primary processes: first, an evaluation of the data quality provided by the logs, and then, second, the computation of the fluid properties of the reservoir. Python, a general-purpose programming language, was used to write the complete workflow, and it was integrated into an existing piece of commercial petrophysics software. Atikol et al. (2013) built a user-friendly graphical user interface executable application to compare the Dykstra and Parson Method findings with the Reznik et al. extension using python scripts. This application was used to analyze the data. The program outcomes for both approaches produced findings that were highly comparable to those acquired from the simulation carried out with Flow (Open Porous Media).

SMART ENERGY FIELD WITH PYTHON

Saadallah et al. (2019) designed a simulator that can simulate transient hydraulics, temperature, torque, drag, and the transport of cuttings. The data obtained from the simulated drilling can be accessed in a few different ways; first, through a web application that is both simple to use and effective as a teaching tool for the physics involved in drilling operations. Second, drilling data can be accessed programmatically by either a web API or programming language APIs written in MATLAB, Python, and. NET. These APIs can be found on the respective websites. Finally, a design automation framework is proposed for the usual pipeline calculations, including code checks. These calculations are carried out utilizing a web-based graphical user interface (GUI) and designed in a cloud-based digital field twin.

Regarding the design phase of the subsea pipeline, some of the most sophisticated level pipeline finite element analyses are carried out to assess buckling and walking (Graziatti, 2017). The Python Application Programming Interface (API) was used to build all of the conventional pipeline computations, and it was coupled to the cloud-based digital twin Subsea-XD. It is shown how a cloud-based digital oilfield twin may be utilized to automate subsea flow assurance engineering operations. As a result, effective collaborations, faster and more reliable designs, and lower costs can be achieved. In order to carry out flow assurance calculations and design analysis, they used a web application that had been developed on top of a digital twin platform that was hosted in the cloud. Python scripts enable the web-based platform's integration with multiphase flow simulators and other pertinent engineering tools. It describes applying a comprehensive field management framework that can create an integrated virtual asset by coupling reservoirs, wells, networks, facilities, and economic models. It also provides an advisory system for efficient asset management. This framework can also provide an integrated virtual asset by coupling reservoirs, wells, networks, facilities, and economic models (Ojha & Bairagi, 2013). This was made possible because of the field management system's extensibility, which Python allows, and its generic capacity.

It collects massive real-time data from more than 30,000 tags and sensors. Real-time data were gathered for a maximum of seconds, and quality checks needed to be performed on each data point obtained (Awan, 2015). To begin, all of the equipment tags and sensors were inspected and reorganized. After that, the API was built in conjunction with the real-time platform. Python scripts are used for the data quality check and validation model. These scripts interact with structured access so the system can read the data and execute quality checks and analyses. For example, NumPy, a library for the Python programming language, multi-dimensional arrays, and matrices, mathematical methods to act on arrays, was utilized during the first part of the project.

Additionally, the percentile approach was utilized within the Python ecosystems. A fresh method for assessing the wellbore instability has been put up for consideration. They focused on data analytics and the creation of the Bayesian Algorithm (with code in Python) to estimate the wellbore failure probability using real-time pore pressure and fracture gradients data acquired from the wellbore. This was accomplished by utilizing the Python programming language. A cost-effective, user-friendly, and highly reliable technique for automating the design of subsea pipelines and structures has been presented on a cloud-based digital field twin platform with python scripting. In addition, the cloud-based design automation method saves a substantial number of computation hours due to a systematic and sequential approach with little remediation work by lowering the number of human mistakes, which in turn reduces the amount of work that needs to be done for correction (Nasreen & Hassan, 2019).

ARTIFICIAL INTELLIGENCE

Asala et al. (2017) employed a shale gas network and supply chain optimization for mixed-integer non-linear programming. The model uses at least four key efforts, including reservoir simulation, which uses feed-forward Neural Network (NN) algorithm output. Multiphase reservoir simulation requires re-frac candidates, which the trained NN algorithm can recommend. Finally, NPV optimization used a four-layer LSTM recurrent neural network to estimate local shale gas demand. Python wrote both neural network algorithms. Noshi et al. (2019) developed a data-driven, interdisciplinary approach to integrate seventy-eight land-based wells to anticipate casing failure. They applied several Data Mining and Machine Learning methods to twenty-four features on twenty failed casing data sets using statistical software and Python Scikit-learn implementation. Regular Distribution Charts and Heat Maps showed descriptive analytics. PCA reduced dimensionality. Response determined supervised and unsupervised techniques. Their model used SVM, Random Forest, Naïve Bayes, XG Boost, and K-Means Clustering. Mohammadmoradi et al. (2018) presented a mechanistically-supported data-driven model for gas condensate well production forecasting using artificial intelligence algorithms. A novel set of mathematical models implemented using Apache Spark cluster computing engine with Python APIs allows rigorous and resilient optimization of the recovery process, creating and identifying hidden patterns in production data and indirectly collecting reservoir information in seconds. Noshi et al. (2018) used Mosaic and Box Plots and prediction algorithms such as ANN and Boosted Ensemble trees on eighty land-based wells, twenty of which had casing and tubing failures. Using predictive analytics and python coding, they evaluated twenty-six features from drilling, fracturing, and geology data. Descriptive and supervised ML techniques helped them identify casing failure causes. Automatic 3D FEM (FEM). Comparing modeling and experimental data shows that the proposed package's results are plausible and likely, especially during the early stage of corrosion. Python and Keras were used to create an autoencoder. The exponential linear unit activated training in 7 layers. The autoencoder never reconstructs input data perfectly, although it does well on data similar to its training set. Choi et al. (2019) developed a leak-off pressure (LOP) prediction model offshore Norway. The model utilizes a DNN on the Norwegian Petroleum Directorate's public wellbore database (NPD). Using Python, they scraped NPD fact pages for data from over 6400 wells (1800 exploration and 4600 development). Analyzed the data to see how spatial and regional factors affected LOPs. A unique multilayer perceptron technique is a deep learning neural network model developed on Tensor flow using Python. The model improved output set variables that match simulation findings.

It was designed as a temporary replacement for downhole pressure readings following the breakdown of the gauge on an offshore gas production well. Inside the realm of machine learning, a solution was discovered by using multivariate linear regression to express the relationships within the production system (Yahaya & Mato, 2017). Python code written with the open-source learn package was used as the foundation for the described workflow. In order to calculate an accurate forecast of the amount of oil that would be extracted from an oilfield, a technique for machine learning called Multiple Linear Regression was devised and implemented in Python. The model was designed, constructed, and outfitted to facilitate the training and testing of the factors that determine and influence oil production volume. The observed relationship between oil production volume and the affecting factors led to the perfect conclusion that the model, if implemented, has the potential to be of immense value in the energy and ecosystems industry due to its capacity to predict oilfield output more accurately. This conclusion was reached after observing the relationship between oil production volume and the affecting factors. It is a developed model in Python using a dataset of approximately 60 kilometers of well-log data. Then it is compared with the logs that specialists interpret according to the bond quality (six ordinal classes) and hydraulic isolation (two classes) of solids outside the casing. They taught the ML systems to recreate these reference interpretations in exactly one meter-long segments.

Machine Learning with Python

Abbas and Mustapha (2019) implemented the ensemble model predictive control approach using Python with the simulator. In order to estimate the deterioration of the well's production index (PI) and optimize intelligent well completions (IWC) following the particular operational philosophy of a deepwater Gulf of Mexico asset, a python simulator has been constructed. In order to give decision support for drilling issues through the use of python scripting, a technique that is a hybrid of machine learning (ML) and physics-based modeling has been developed. Using wireline and production data, a metric-based machine learning approach is offered to identify and describe regional patterns in reservoir heterogeneity and the property distribution of its many 'facies.' They used an actual mature producing field in Western Siberia to demonstrate how the suggested method can assist in the partitioning of reservoir heterogeneity as well as the discovery and verification of regional patterns. The clustering of reservoir facies generated from the wireline logs (alpha-SP) displayed a good agreement with the reservoir zonation based on the interpretation of manual logs and the geological concept. It was designed as a temporary replacement for downhole pressure readings following the breakdown of the gauge on an offshore gas production well. In the realm of machine learning, a solution was discovered by employing multivariate linear regression to express relationships inside the production system. This enabled the system to function optimally. Python code written with the open-source learn package was used as the foundation for the described workflow. In order to calculate an accurate forecast of the amount of oil that would be extracted from an oilfield, a technique for machine learning called Multiple Linear Regression was devised and implemented in Python. The model was designed, constructed, and outfitted to facilitate the training and testing of the factors that determine and influence oil production volume. The observed relationship between oil production volume and the affecting factors led to the perfect conclusion that the model, if implemented, has the potential to be of immense value in the energy and ecosystems industry due to its capacity to predict oilfield output more accurately. This conclusion was reached after observing the relationship between oil production volume and the affecting factors. It is a developed model in Python using a dataset of approximately 60 kilometers of healthy log data. Then it is compared with the logs that specialists interpret according to the bond quality (six ordinal classes) and hydraulic isolation (two classes) of solids outside the casing.

Big Data with Python

The technique of applying storyboarding to a dataset of more than 100 terabytes (GB) in size and originated from 16 shale wells drilled in North America was detailed in Saini et al. (2018) .'s article. Matlab and Python scripts were created to automatically process the raw data and generate over 20 distinct types of one-page visualizations. The information that has been illustrated provides some insights into its performance, as well as the wellbore tortuosity, quality, vibrations, weight on bit transfer, and other drilling dynamics. Rodger and Garnett (2018) reported data recorded by Pason electronic drilling recorders at 970 wells and end-of-day reports for 370 of these wells. These reports were presented in conjunction with the data. To automatically construct the optimal composite time model for each investigated area, scripts written in the computer language Python were implemented. These scripts were responsible for breaking the 812 in. drilling stage down into depth portions. Individual good data was compared to this benchmark, which allowed the drilling performance to be compared to that of other wells in the same field. Identifiable removable time was then classified as either invisible lost time (ILT) or non-productive time, depending on the nature of the loss it represented (NPT). Across 828 wells, a total of over 4,500 hours, equivalent to approximately 49.5% of the entire time spent digging an 812-inch hole, was determined to be detachable. According to Ejimuda & Ejimuda (2018), a visual inspection is essential to build and implement an effective risk management strategy. They highlighted how to address such difficulties by applying cutting-edge computer vision and deep learning techniques. In order to accomplish this goal, they utilized the Python programming language, the Tensorflow Application Programming Interface, the Resnet deep learning architecture, GPU processors, and cloud computing technology. A method for capturing the data collected by the rig sensors was devised by Achar (2019). The following steps were taken: Gather massive amounts of data by integrating sensors into the rig to provide data sets for drilling parameters. Python's Pandas library was used to develop a method for creating an algorithm that can clean up enormous data sets.

Simulation Time with Python

It has been demonstrated how a technique based on scripted Python code can reduce the time required for run preparation by at least two weeks compared to the manual alternative. It was conceived to offer an "automatic moving window" to locate optimal intervals along a favorable route (Akter et al., 2013). This Python-based automated script was run in the pre-processor of the dynamic simulator, which features a workflow window capable of incorporating Python. It is proposed to set up a series of processes that will ease the model deployment process and establish an automatic advisory system that will provide insight to validate an engineer's day-to-day engineering decisions. The workflows use technology from a commercial visualization dashboard, python scripts, open-source machine learning tools written in Python, and a flow assurance simulator. To develop a field-level automated optimization system, a total of three steps was produced (Hossain et al., 2019). In the bulk well-modeling workflow, the time spent per well was cut in half, from 30 minutes to 10 minutes, thanks to the first and second steps.

Additionally, the time spent on the network model merge was cut in half, from two hours to ten minutes, based on the assumption that there are 100 wells in one network. In order to pre-process the data, rearrange it, and build unified data frames, a script written in Python is used. This results in a significant reduction in the amount of time necessary for the pre-processing of a wide variety of subsurface data sources, including static and dynamic reservoir models, log data, historical production, and pressure data, and wells and completion data, to name a few of these data types (Achar, 2019).

Conclusion

Python is an additional object-oriented programming language that, compared to other programming languages, helps developers reduce the amount of time and effort required to create functions in a smaller number of lines of code. In the fields of energy and ecological systems, this application has emerged as a desirable option for constructing intelligent models that can predict, diagnose, or analyze the performance of reservoirs and wells effectively and precisely. It is a programming language that is regarded to have many paradigms and simpler coding syntax and methodologies. It is widely used. It has an extensive collection of built-in standard libraries and features, which combine to make it a language with a high level of usability in the real world. On the other hand, focuses more on the talent sector and is most often used in machine learning, the internet of things, and artificial intelligence. It promises a promising future for the energy and ecosystems business due to the exponential growth in demand for its use over the past several years, particularly in light of the emergence of artificial intelligence and machine learning.

Policy Implications

However, quality assurance for well/reservoir performance analysis can be obtained by comparison of the Python-based simulator and commercial software. If the two numbers are the same, then there is reason to have more faith in the performance analysis of the well or reservoir. On the other hand, if the two values are significantly different, then it is necessary to reconsider the assumptions. Because of this, engineers will be able to apply simulation in new ways because the capabilities of simulators will be extended. Additionally, this will enable the energy and ecosystems business to create flexible control logic to handle field management difficulties. The advantages of using Python scripting in reservoir simulation can be shown in the following areas:

- Provide financial assistance to petroleum engineers who wish to improve their programming skills.
- Encourage the growth of open-source projects within the industry of energy and ecosystems.
- Assist petroleum engineers in more effectively managing production operations, preferably without the need for pricey software.
- Ensure the models' quality by extracting and analyzing the data.
- Model dynamic well behaviors and link the reservoir simulator to other apps such as Microsoft Excel.

REFERENCES

Abbas H, Mustapha H. An Efficient Method to Control Oilfield Wells Through Advanced Field Management Ensemble-Based Optimization Capabilities. Presented at the Abu Dhabi International Petroleum Exhibition and Conference. 2019; SPE-197730-MS. Available: https://doi.org/10.2118/197730-MS

Achar, S. (2015). Requirement of Cloud Analytics and Distributed Cloud Computing: An Initial Overview. *International Journal of Reciprocal Symmetry and Physical Sciences*, pp. 2, 12–18. Retrieved from https://upright.pub/index.php/ijrsps/article/view/70

Achar, S. (2016). Software as a Service (SaaS) as Cloud Computing: Security and Risk vs. Technological Complexity. *Engineering International*, *4*(2), 79–88. https://doi.org/10.18034/ei.v4i2.633

Achar, S. (2018a). Data Privacy-Preservation: A Method of Machine Learning. *ABC Journal of Advanced Research*, *7*(2), 123–129. https://doi.org/10.18034/abcjar.v7i2.654

Achar, S. (2018b). Security of Accounting Data in Cloud Computing: A Conceptual Review. *Asian Accounting and Auditing Advancement*, *9*(1), 60–72. https://4ajournal.com/article/view/70

Achar, S. (2019). Early Consequences Regarding the Impact of Artificial Intelligence on International Trade. *American Journal of Trade and Policy*, *6*(3), 119–126. https://doi.org/10.18034/ajtp.v6i3.634

Akter, N., Rahman, M. M., & Akter, S. (2013). Introducing a Phenomenon of Single Junction Multiple Band-gap Solar Cell Compared to Single Junction or Multi Junction Solar Cell to Achieve High Efficiency. *ABC Journal of Advanced Research*, *2*(1), 30-35. https://doi.org/10.18034/abcjar.v2i1.16

Al Mahmud, T. (2012). A Survey on How Dynamically Changes Topology in Wireless Sensor Network. *ABC Journal of Advanced Research*, *1*(1), 28-34. https://doi.org/10.18034/abcjar.v1i1.3

Amin, R. (2019). Innovations in Information Systems for Business Functionality and Operations Management. *ABC Research Alert*, *7*(3), 148–158. https://doi.org/10.18034/abcra.v7i3.546

Asala HI, Chebeir J, Zhu WI, Gupta A, Taleghani D, Romagnoli J. A Machine Learning Approach to Optimize Shale Gas Supply Chain Networks. Presented at the SPE Annual Technical Conference and Exhibition. 2017; SPE-187361-MS. Available: https://doi.org/10.2118/187361-MS

Atikol, U., Enayatollahi, R., & Madani, S. S. (2013). Performance of a Solar Humidification Dehumidification Desalination System on December 27th and 28th in North Cyprus. *ABC Journal of Advanced Research*, *2*(1), 20-29. https://doi.org/10.18034/abcjar.v2i1.15

Awan, A. G. (2015). Environmental Challenges to South Asian Countries. *Asian Accounting and Auditing Advancement*, *6*(1), 83–102. https://doi.org/10.18034/4ajournal.v6i1.38

Choi Jung C, Skurtveit E, Lars Grande. Deep Neural Network Based Prediction of Leak-Off Pressure in Offshore Norway. Presented at the Offshore Technology Conference. 2019; OTC-29454-MS. Available: https://doi.org/10.4043/29454-MS.

Diakonova E, Bonnet G. Yann Brouard. Innovative Design Methodology Supporting Concept EngineeringThrough to Project Cost Optimization. Presented at the Offshore Technology Conference. 2019;OTC-29529-MS. Available: https://doi.org/10.4043/29529-MS

Dipa, D. D., Abedin, K., Khan, M. M., & Hasan, M. M. (2015). Impacts of Energy Subsidy in Bangladesh: An Analysis. *ABC Journal of Advanced Research*, *4*(1), 39-56. https://doi.org/10.18034/abcjar.v4i1.44

Ejimuda CC, Ejimuda CM. Using Deep Learning and Computer Vision Techniques to Improve Facility Corrosion Risk Management Systems. Presented at the SPE Nigeria Annual International Conference and xhibition. 2018; SPE-190036-MS. Available: https://doi.org/10.2118/198863-MS

Graziatti, L. V. (2017). The Treaty of Rome EEC and EURATOM 1957. *ABC Research Alert*, *5*(3), Peru. https://doi.org/10.18034/abcra.v5i3.316

Hesar M, TziPiau C, Qingjing M, Leonardo G, Carlos C. Seismic Design of Large Scale Integrated Subsea Facilities. presented at the 28th International Ocean and Polar Engineering Conference. 2018;10–15:ISOPE-I-18-681.

Hossain, S., Mrida, M. I. A., Das, P. C., Sayed, A., Rahman, N., & Rashid, M. M. (2015). Mimo Channel and Performance Analysis using OFDM System for Reduced Bit Error Rate. *ABC Research Alert*, *3*(3), Bangladesh. https://doi.org/10.18034/abcra.v3i3.299

Hossain, S., Rahman, M. S., & Yeasmin, S. (2019). Environmental Radioactivity Monitoring and Assessment of Excess Lifetime Cancer Risk to People in Demra Thana, Dhaka, Bangladesh. *ABC Research Alert*, *7*(3), Bangladesh. https://doi.org/10.18034/abcra.v7i3.269

Hosseinimotlagh, S. N. (2014). Computation of time energy gain in D-3He mixture: Energy deposited through deuterium ignition beam. *Asia Pacific Journal of Energy and Environment*, *1*(2), 153-168. https://doi.org/10.18034/apjee.v1i2.217

Kushkumbayeva G, Zhumabayev B, Daniyar G, Ruslan B, Justus A, Tamer S, Giovanni B, Alberto B. Evaluating Effectiveness of Stimulation Treatment in Karachaganak Multistage Wells. Presented at the SPE Annual Caspian Technical Conference and Exhibition. 2018; 2:SPE-192569-S. https://doi.org/10.2118/192569-MS

Maiorano S, Selvaggio P, Distaso RE, Rossi R, Stano E. Innovative Water Salinity Management Through Integrated Asset Model Applied to Mexico Area-1. Presented at the SPE Reservoir Characterization and Simulation Conference and Exhibition. 2019; 17– 19.SPE-196622-MS. Available: https://doi.org/10.2118/196622-MS

Mohammadmoradi P, Moradi, Hessam M, Kantzas A. Data-Driven Production Forecasting of Unconventional Wells with Apache Spark. Presented at the SPE Western Regional Meeting. 2018; SPE-190098-MS. Available: https://doi.org/10.2118/190098-MS

Nasreen, H., & Hassan, M. (2019). Nitazoxanide – A New Option in Biliary Ascariasis. *ABC Research Alert*, *7*(1), Bangladesh. https://doi.org/10.18034/abcra.v7i1.236

Noshi C, Noynaert S. Jerome Schubert. Data Mining Approaches for Casing Failure Prediction and Prevention. presented at the International Petroleum Technology Conference. 2019; IPTC-19311-MS. Available: https://doi.org/10.2523/IPTC-19311-MS

Noshi CI, Noynaert SF, Schubert JJ. Failure Predictive Analytics Using Data Mining: How to Predict Unforeseen Casing Failures? Presented at the Abu Dhabi International Petroleum Exhibition & Conference. 2018;SPE-193194-MS. Available: https://doi.org/10.2118/193194-MS

Ojah M, Emumena E, Collins O, Oduh P, Steve A. Determination of Optimal Well Location in Bounded Reservoirs Using the Dimensionless Pressure Derivative. SPE-203627-MS. presented at the SPE Nigeria Annual International Conference and Exhibition; 2020. Available: https://doi.org/10.2118/203627-MS

Ojha, A., & Bairagi, S. (2013). Corporate Restructuring. *ABC Research Alert*, *1*(1), India. https://doi.org/10.18034/abcra.v1i1.243

Olusola Bukola K, Orozco D, Aguilera R. Optimization of Recovery by Huff and Puff Gas Injection in Shale Oil Reservoirs Using the Climbing Swarm Derivative Free Algorithm. Presented at the SPE Latin American and Caribbean Petroleum Engineering Conference. 2020; SPE-199028-MS. Available: https://doi.org/10.2118/199028-MS

Rodger I, Andrew, Garnett. Lost Time Analysis of Queensland Coal Seam Gas Drilling Data and Where Next for Improvement? Presented at the SPE Asia Pacific Oil and Gas Conference and Exhibition. 2018; SPE-192034-MS. Available: https://doi.org/10.2118/192034-MS

Saadallah N, Jan Einar Gravdal, Robert Ewald, Sonja Moi, Adrian Ambrus, and Benoit Daireaux, Norce; Stian Sivertsen and Kristian Hellang, Miles; Roman Shor, Dan Sui; Stefan Ioan Sandor, Aker BP; Marek Chojnacki, Jacob Odgaard. OpenLab: Design and Applications of a Modern Drilling Digitalization Infrastructure. Presented at the SPE Norway One Day Seminar. 2019; SPE-195629-MS. Available: https://doi.org/10.2118/195629-MS

Saini G, Chan H, Ashok PE, Van, Oort MR. Isbell. Automated Large Data Processing: A Storyboarding Process to Quickly Extract Knowledge from Large Drilling Datasets. Presented at the IADC/SPE Drilling Conference and Exhibition. 2018; IADC/SPE-189605-MS. Available: https://doi.org/10.2118/189605-MS

Sarkar, M. S. K., Sadeka, S., Sikdar, M. M. H., & Badiuzzaman. (2018). Energy Consumption and CO2 Emission in Bangladesh: Trends and Policy Implications. *Asia Pacific Journal of Energy and Environment*, *5*(1), 41-48. https://doi.org/10.18034/apjee.v5i1.249

Shoaib M, Selvaggio P, Cominelli A, Rossi R, Thompson A, Amoudruz P. Flexible Integrated Asset Models for Deep Water Developments Presented at the Offshore Mediterranean Conference and Exhibition. 2019; OMC-2019-0996.

Velmurugan, C., & Radhakrishnan, N. (2018). Visualizing Energy and Environment Research Productivity in Australia: A Scientometric Profile. *Asia Pacific Journal of Energy and Environment*, *5*(1), 11–26. https://doi.org/10.18034/apjee.v5i1.246

Yahaya, I., & Mato, U. (2017). Workforce Diversity and Organizational Effectiveness in 21st Century Business Arena. *Asian Accounting and Auditing Advancement, 8*(1), 24–29. https://doi.org/10.18034/4ajournal.v8i1.46

Zhou W, Samad Ali, Shripad B. Openness in Reservoir Simulation: Empowering Flexible Field Management. Presented at the SPE Asia Pacific Oil & Gas Conference and Exhibition. 2020; SPE-202303-MS. Available: https://doi.org/10.2118/202303-MS

--0--