Journal of Engineering Research and Reports

21(3): 13-22, 2021; Article no.JERR.76195
ISSN: 2582-2926

Review of Python Applications in Solving Oil and Gas Problems

Edet Ita Okon[1*], Dulu Appah[1] and Joseph A. Ajienka[1]

[1]*Department of Petroleum Engineering, University of Port Harcourt, Nigeria.*

Authors' contributions

This work was carried out in collaboration of three authors. Author EIO designed and prepared the manuscript. Author DA read and checked the whole manuscript. Author JAA analyzed and interpreted the manuscript. All authors read and approved the final manuscript.

Article Information

DOI: 10.9734/JERR/2021/v21i317448
Editor(s):
(1) Dr. Neha Munjal, Lovely Professional University, India.
Reviewers:
(1) Mihai Gianina, Universitatea "Dunărea de Jos" Galați, Romania.
(2) Abdelaziz Nasr El-hoshoudy, Egyptian Petroleum Research Institute, Egypt.
Complete Peer review History: https://www.sdiarticle4.com/review-history/76195

Received 11 August 2021
Accepted 22 October 2021
Published 29 October 2021

Review Article

ABSTRACT

Python has grown in popularity throughout various industries, corporations, universities, government, and research groups. Its true potential to automate various processes while increased predictability capabilities have been noticed in various industries. The petroleum industry is at the beginning phase of applying it to solve oil and gas problems. The rise in its popularity in the oil and gas industry is due to the digital transformation such as sensors and high-performance computing services that enable artificial intelligence (AI), machine learning (ML), big data acquisition, and storage in digital oilfields. A quick search on the number of publications in the oil and gas industry with the Society of Petroleum Engineers (OnePetro) in the past few years attests to this fact. Hence, it has proven to be a promising application that can bring about a revolutionary change in the oil and gas industry and transform the existing features for solving oil and gas problems. This will help the production and reservoir engineers to better manage the production operation without any need for expensive software. It will also reduce the overall operating cost and increase revenue.

Keywords: Python; oil and gas problems; petroleum.

ABBREVIATIONS

Artificial intelligence (AI)
Internet of things (IoT)
Machine learning (ML)
Sucker rod anti-buckling pump (SRABS)
Finite difference (FD)
Finite element (FE)

**Corresponding author: Email: eddymyprince@gmail.com;*

Climbingswarm (CS)
Material-balance (MatBal)
Net-present-value (NPV)
Pressure Volume Temperature (PVT)
Graphical user interface (GUI)
American Petroleum Institute (API)
Neural Network (NN)
Long Short-Term Memory (LSTM)
Principal component Analysis (PCA)
Support Vector Machine (SVM)
Artificial Neural Networks (ANN)
Finite Element Method (FEM)
Deep neural network (DNN)
Leak-off pressure (LOP)
Norwegian Petroleum Directorate (NPD)
Gigabytes (GB)
Central processing unit (GPU)
Invisible lost time (ILT)
Non-productive time (NPT)
Intelligent well completions (IWC)

1. INTRODUCTION

Python is an interpreted high-level general-purpose programming language and has become an attractive application in building intelligent models that can predict, diagnose or analyze reservoir and well performance efficiently and accurately in the oil and gas industry. In another way, it is a programming language that stands out for its readable and clean code. The open-source license allows its use in different contexts. Its objective is the automation of processes to save both compilations and time, fundamental for the optimization in the labor field. It is the engine for algorithm development, prototyping, and early-stage deployment. It is a powerful engineering computing/coding program with strong capability in data management, visualization, and debugging. It provides various toolboxes for signal processing, machine learning, and statistical analysis. It has evolved to have a great impact in the technical computing community and enabled technical programming by an extensive set of libraries. Moncorgé *et al.* [1] proposed an approach of using low-level interfaces to communicate between the core and the modules (such as the fluid, the petrophysical, or the surface facility modules). The performance of this system was then simulated using a combination of MATLAB and Python codes, visualized in the interface of Abaqus. These simulations confirm that this wire rope can indeed replace the rod string in a sucker rod pumping unit. Detailed stress, load, and movement profiles were also compiled to allow for a comprehensive analysis. Kushkumbayeva *et al.* [2] proposed an evaluation of the technical assessment of multistage stimulation based on lithology, porosity, natural fractures, stimulation type, infectivity, and the production logging data using Python script. The zonal injectivity of each well generated in Excel describes the pre-and post-stimulation injectivity relative to the stimulation type (acid wash, high-rate matrix, and acid fracturing). They used Python's script to read Excel injectivity analyses data for all the wells and generated plots for multivariable analyses of the impact of various parameters on productivity.

Maiorano *et al.* [3] used Python in an IAM to vary injected salinity considering the real mixing process of seawater and produced water. Shoaib *et al.* [4] used Python scripts to pass the gas lift rate between the reservoir simulator and the network simulator. Hesar *et al.* [5] used a python script to illustrate that large-scale simulation of subsea systems is not only possible but also necessary to be carried out in an integrated manner. This indicates that an integration of the subsea clusters into a single model simulates reality better than conventional approaches and removes the need for major simplifications or sub-modeling. The integration also captures the “system effects” which would otherwise be masked if different units were to be modeled separately. To investigate the time-lapse, three-dimensional (so-called four-dimensional/4D) stress during production/injection, a Python code is provided to communicate data between the finite difference (FD) and finite element (FE) grids [6]. Diakonova *et al*. [7] presented an optimization riser configuration that allows the best compromise between all parameters while remaining fit for purpose and cost-efficient. The methodology was developed using a Python script as an interface between Orcaflex software and the user-defined parameters to best respect the project-specific weighted criteria. Podskarbi and Knezevic [8] presented a framework, based on a cloud-based simulation platform and Python-based scripting, to automate a workflow. The proposed digital twin provides a single source of truth about asset conditions and allows data-driven communication and decision-making between multiple stakeholders – operators, contractors, regulators, etc. Noshi [9] developed Python code that can identify proper inflection points at the mid-point of the curve turns and use inclination and azimuth indices. The technique is inspired by the discipline of ophthalmology, specifically a method to determine tortuosity from retinal blood vessels. The approach successfully produced a tortuosity metric with three different risk categories characterizing three ranges of the index. The indices generated were matched against operator reports of drilling incidents and NPT. The methodology matched highly tortuous wells with greater downhole tool failures rates ranking it in the high-risk zone. Khait and Voskov [10] demonstrated how to extract complex physics-related computations from the main simulation loop, leaving only an algebraic multilinear interpolation kernel instead. They described the integrated simulation framework built on top of this technique and showed the applicability of the approach to various challenging physical and chemical problems. All simulation engines along with linear solvers, well controls, interpolation engines, and state operator evaluators are implemented in C++11 and exposed into Python coupling the flexibility of the script language with the performance of C++.

Zhou *et al.* [11] demonstrated the flexibility power of the python programming language in reservoir management. It empowers engineers to utilize simulation in new ways and extends simulator capabilities to enable them to implement flexible-control logic to solve field management challenges. Olusola *et al.* [12] presented an original methodology to learn how to perform these tasks faster and at a lower cost to improve oil recovery. The procedure is explained with the use of an actual H&P gas-injection pilot horizontal well in the Eagle Ford Shale, the performance of which is matched using the methodology. The methodology includes the use of an original climbing swarm (CS) derivative-free algorithm that drives, without human intervention, desktop computer or laptop material-balance (MatBal) and net-present-value (NPV) calculations. The code was written in an open-source Python programming language. Oladipo and Nwankwo [13] used Python programming language to compute a table of friction data using both the Colebrook-White Equation and the Weymouth Friction Factor equation. A correction factor was introduced into the Weymouth friction factor that takes into consideration the variation of pipe roughness. Further, the new friction factor relationship was used to modify the existing Weymouth equation. Ojah *et al.* [14] deployed Python and Excel to compute all the dimensionless pressures for different well designs. The dimensionless pressure derivatives of a vertical oil well are studied to search for an optimum well location that can guarantee satisfactory oil production without the premature influence of the external boundaries. Eriavbe and Uzoamaka [15] analyzed the typical workflow governing probabilistic evaluation methodologies and proposed a Python script-based approach that enables the user to run a fast and simple mineral components evaluation based on porosity and basic input logs. Data from a typical Niger Delta well is used to evaluate the workflow and the results are compared with a deterministic evaluation to see the added benefits. Ujjwal *et al.* [16] presented a case study of advanced analytics applied to a mature waterflood. They carried out a study for a mature waterflood field with over 15 years of water injection history and over 100 active producers. Python programming was used to clean up and integrate various data sources into an integrated visualization dashboard. Yang *et al.* [17] developed a workflow for generating reservoir fluid from data logs and PVT databases. The workflow consists of two main processes; first a quality assessment of logs data and second the computation of reservoir fluid properties. The entire workflow is written in python programming language and embedded into existing commercial petrophysics software. Kiisi *et al.* [18] developed a user-friendly graphical user interface executable application to compare the results of the Dykstra and Parson method, and the Reznik et al extension using python scripts. The results of the program for both methods gave a close match with that obtained from the simulation performed with Flow (Open Porous Media).

2. DIGITAL OILFIELD WITH PYTHON

Saadallah *et al.* [19] developed a simulator capable of simulating transient hydraulics, temperature, torque and drag, and cuttings transport. The simulated drilling data can be accessed by several means. First, through a user-friendly web application used as a tool for teaching the physics involved in drilling operations. Secondly, drilling data can be accessed programmatically through a web API or via programming language APIs written in MATLAB, Python, and. NET. Bhowmik and Naik [20] proposed a design automation framework for all the standard pipeline calculations including code checks and are performed through a web-based graphical user interface (GUI) designed in a cloud-based digital field twin. In the design phase of the subsea pipeline, some more advanced level pipeline finite element analyses are performed for buckling and walking assessment. All the standard pipeline calculations are developed using Python API and connected to cloud-based digital twin Subsea-

XD. Patel *et al.* [21] demonstrated how cloud-based digital oilfield twin can be leveraged to automate subsea flow assurance engineering workflows and consequently, achieve efficient collaborations, faster and reliable designs, and reduced costs. They used a web application built on top of a cloud-based digital twin platform to perform flow assurance calculations and design analysis. The web-based platform integrates multiphase flow simulators and other relevant engineering tools using python scripts. Khan *et al*. [22] described the application of a comprehensive field management framework that can create an integrated virtual asset by coupling reservoir, wells, network, facilities, and economic models and provides an advisory system for efficient asset management. This was accomplished through the Python-enabled extensibility and generic capability of the field management system.

Fuad *et al.* [23] collected huge data from more than 30,000 tags/sensors in real-time. The real-time data were collected up to seconds and quality check need to be done up to each data collected. Firstly, each equipment tags/sensors were checked and arranged. Then, API was developed with the real-time platform. The data quality check and validation model using python script which interact with structured access for the system to read and perform quality checking and analysis. For the first phase, NumPy (a library for the Python programming language, multi-dimensional arrays, and matrices, mathematical functions to operate on arrays) was used and percentile method had been applied in the python environment. Adeyemi *et al*. [24] proposed a new approach to quantifying the wellbore instability. They focused on data analytics and the development of the Bayesian Algorithm (with code in Python) to predict the wellbore failure probability using real-time pore pressure and fracture gradients data obtained from the wellbore. Bhowmik and Naik [20] demonstrated a cost-effective, user-friendly, and highly reliable subsea pipeline and subsea structure design automation method developed on a cloud-based digital field twin platform with python scripting. In the cloud-based design automation method, a significant number of calculation hours are saved due to a systematic and sequential approach with minimum remediation work by reducing human error.

3. ARTIFICIAL INTELLIGENCE

Asala et al. [25] used a shale gas network and supply chain optimization for a mixed-integer non-linear programming formulation. The model relies on at least 4 major efforts including reservoir simulation, which in turn relies on output from a feed-forward Neural Network (NN) algorithm. The trained NN algorithm was deemed suitable for recommending re-frac candidates, necessary decision variables for multiphase reservoir simulation. Finally, NPV optimization relied on a four-layer Long Short-Term Memory (LSTM) recurrent neural network, developed for forecasting local shale gas demand. Both neural network algorithms were scripted using the python programming language. Noshi *et al.* [26] proposed a more interdisciplinary approach to integrate seventy-eight land-based wells using a data-driven approach to predict the reasons behind casing failure. They used statistical software in collaboration with Python Scikit-learn implementation to apply different Data Mining and Machine Learning algorithms on twenty-four different features on the twenty failed casing data sets. Descriptive analytics manifested in visual 8 representations included Normal Distribution Charts and Heat Map. Principal component analysis (PCA) was used for dimensionality reduction. Supervised and unsupervised approaches were selected respectively based on the response. The algorithms used in their model included Support Vector Machine (SVM), Random Forest, Naïve Bayes, XG Boost, and K-Means Clustering. Mohammadmoradi, *et al.* [27] proposed a model to embed artificial intelligence algorithms in reservoir uncertainty modeling and present a mechanistically-supported data-driven model applicable for production forecasting of gas condensate wells with higher confidence. The outcome entails a new set of mathematical models, implemented using Apache Spark cluster computing engine with APIs in Python, that enables rigorous and robust optimization of the recovery process, designing and discovering hidden patterns in production data, and extracting reservoir information indirectly in seconds. Noshi *et al.* [9] applied both descriptive visual representations such as Mosaic and Box Plots and predictive algorithms including Artificial Neural Networks (ANN) and Boosted Ensemble trees on eighty land-based wells, of which twenty possessed casing and tubing failures. They used predictive analytics software and python coding to evaluate twenty-six different features compiled from drilling, fracturing, and geologic data. This approach enabled them to find out the possible factors contributing to casing failures using both descriptive and supervised ML algorithms. An automatic 3D Finite Element Method (FEM) computation procedure scripted by

MATLAB/ABAQUS/ Python was proposed for a stochastic solid model of corroded plates by Zeng *et al.* [28]. Comparing the simulation and experiment results show that, the results using the proposed package are quite reasonable and believable in high probability, particularly at the early stage of corrosion. Alamu *et al.* [29] developed an autoencoder using the Python programming language along with the Keras deep learning framework. It had 7 layers with the exponential linear unit as the activation function for training. During reconstruction, the autoencoder never produces a perfect reconstruction of input data, it, however, performs a good reconstruction on data similar to what it was trained on. Choi *et al*. [30] presented a novel prediction model for the leak-off pressure (LOP) offshore Norway. The model uses a deep neural network (DNN) applied on a public wellbore database provided by the Norwegian Petroleum Directorate (NPD). They used a Python-based web scraping tool to collect data from more than 6400 wells (1800 exploration wells and 4600 development wells) from the NPD fact pages. Then, analyzed the collected data to investigate the impacts of spatial and regional factors on the collected LOPs. Okoro, *et al.* [31] proposed a novel idea, a multilayer perceptron approach which is a deep learning neural network model built on Tensor flow using the python programming language. The model was implemented to further increase the accuracy of the output set variables which are matched with the simulation results.

4. MACHINE LEARNING WITH PYTHON

Abbas and Mustapha [32] implemented the ensemble model predictive control algorithm using Python with the simulator. Avila [33] developed a python simulator to model well PI degradation and optimize intelligent well completions (IWC) according to the specific operational philosophy of a deepwater Gulf of Mexico asset. Agito and Bjorkevoll [34] proposed a hybrid approach between machine learning (ML) and physics-based modeling to provide decision support for drilling problems using python scripting. Konoshonkin, *et al*. [35] proposed a metric-based machine-learning approach to identify and describe spatial trends in reservoir heterogeneity/facies property distribution using wireline and production data. They demonstrated how the proposed method can help to partition reservoir heterogeneity and discover and verify spatial trends for a real mature producing field in Western Siberia. The obtained clustering of reservoir facies based on the wireline logs (alpha-SP) demonstrated a good agreement with the reservoir zonation based on manual log interpretation and the geological concept. Lyall *et al*. [36] developed a temporary proxy for downhole pressure measurements after gauge failure on an offshore gas production well. A solution was found in the machine learning space by applying multivariate linear regression to represent relationships within the production system. The workflow presented was based on Python code using the open-source learn library. Prince [37] developed a Machine Learning algorithm called Multiple Linear Regression using Python programming language to predict the production volume of oil in an oilfield. The model was developed and fitted to train and test the factors that affect and influence the oil production volume. The relationship between oil production volume and the affecting factors was observed and drawn to a perfect conclusion that the model can be of immense value in the oil and gas industry if implemented because of its ability to predict oilfield output more accurately. Viggen *et al.* [38] developed a model in python using a dataset of around 60 km of well log data and compared it with specialist interpreters logs according to the bond quality (6 ordinal classes) and hydraulic isolation (2 classes) of solids outside the casing. They trained the ML systems to reproduce these reference interpretations in segments of 1m length.

5. BIG DATA WITH PYTHON

Saini *et al*. [39] described how the storyboarding process was applied to a dataset of more than 100 gigabytes (GB) from 16 shale wells drilled in North America. Scripts were written in Matlab and Python to automatically process the raw data and generate more than 20 different types of one-page visualization. The illustrated information includes insights into its performance, wellbore tortuosity, quality, vibrations, weight on bit transfer, and other drilling dynamics. Rodger and Garnett [40] presented data recorded by Pason electronic drilling recorders at 970 wells along with end-of-day reports for 370 of these wells. Scripts written in the Python programming language were implemented to break the 8½ in. drilling stage down into depth sections and automatically generate the best composite time model for each field in the study. Individual good data was compared to this benchmark allowing the drilling performance to be compared to other wells in the same field and identified removable time was classified as either invisible lost time (ILT) or non-productive time (NPT). In total over 4500 hours, or approximately 49.5% of the total 8½ in. drilling time, was identified as removable time across 828 wells. Ejimuda and Ejimuda [41] explained that the key parameter required to design and implement an effective risk management strategy is a visual inspection. They demonstrated how using state-of-the-art computer vision and deep learning techniques could address such challenges. They applied python programming language, Tensorflow Application Programming Interface, Resnet deep learning architecture, GPU machines, and cloud computing technologies to achieve this. Cornel and Vazquez [42] developed an approach to capture the rig sensors data. These steps were as follows: Capture big data using sensors integrated on the rig to produce drilling parameters data sets. An algorithm to clean large data sets with the use of Python Pandas libraries was created.

6. SIMULATION TIME WITH PYTHON

Chong and Shen [43] demonstrated how a scripted Python code approach cut down the run preparation time by at least two weeks compared to the manual solution. It was designed to provide an "automated moving window" to find the best intervals along a good trajectory. This automated script was executed in the pre-processor of the dynamic simulator which has a workflow window with Python-embedded capability. Naufal and Metra [44] proposed a series of workflows to simplify model deployment and set up an automatic advisory system to provide insight as a means to justify an engineer’s day-to-day engineering decision. The workflows leverage technologies from a flow assurance simulator, python script, open-source machine learning packages in python, and a commercial visualization dashboard. A total of three steps were prepared to achieve a field-level automated optimization system. From the first and second steps, time consumed was reduced from 30 minutes/well to 10 minutes/well in bulk well-modeling workflow and from 2 hours to 10 minutes for the network model merge with the assumption of 100 wells in one network. Srinivasan *et al*. [45] used a python script to pre-process, restructure and create unified data frames. This significantly reduces the time required to pre-process a diverse number of subsurface data sources consisting of static, dynamic reservoir models, log data, historical production & pressure data, and wells & completion data to name a few.

7. CONCLUSION

Python is another object-oriented programming language that enables developers to give fewer efforts to program functions in lesser code lines compared to other programming languages. It has become an attractive application in building intelligent models that can predict, diagnose or analyze reservoir and well performance efficiently and accurately in the oil and gas industry. It is considered as a programming language with multiple paradigms with easier coding syntax and methods. It comes with a massive set of inbuilt standard libraries and features that make it a language of practical usability. On the other hand, is more in the talent field and is mostly popular in machine learning, IoT, and AI fields. Finally, as the demand for its application has grown exponentially in recent years especially with the advent of AI and machine learning, we can safely say that it holds a promising future for the oil and gas industry.

8. RECOMMENDATIONS

A comparison of the Python-based simulator and commercial software, however, can provide quality assurance for well/reservoir performance analysis. If the two values agree, then confidence in the well/reservoir performance analysis will increase but if the two values are away different, then the assumptions need to be reexamined.

CONTRIBUTIONS TO KNOWLEDGE

This will empower engineers to utilize simulation in new ways by extending simulator capabilities and enabling the oil and gas industry to implement flexible control logic to solve field management challenges. The following areas demonstrate the benefits of python scripting in reservoir simulation:

1. Support the development of
2. programming skills for petroleum engineers.
3. Promote the development of Open-Source initiatives for the oil and gas industry.
4. Help petroleum engineers to better manage the production operation without the need for expensive software.
5. Quality control models by extracting and examining data from models.
6. Model dynamic well behaviors.

7. Link reservoir simulator to other applications such as Excel.

COMPETING INTERESTS

Authors have declared that no competing interests exist

REFERENCES

1. Moncorgé A, Patacchini L, Romain de Loubens. Multi-Phase, Multi-Component Simulation Framework for Advanced Recovery Mechanisms. Presented at the SPE Reservoir Simulation Conference. 2012;10–11. SPE 161615. Available:https://doi.org/10.2118/193930-MS.
2. 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. Available: https://doi.org/10.2118/192569-MS.
3. 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.
4. 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.
5. 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.
6. Tang X, Haiyan Z, Qingyou L, Yujia S. A Reservoir and Geomechanical Coupling Simulation Method: Case Studies in Shale Gas and CBM Reservoir. Presented at the International Petroleum Technology Conference. 2019;IPTC-19288-MS. Available:https://doi.org/10.2523/IPTC-19288-MS.
7. 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.
8. Podskarbi M, Knezevic David J. Digital Twin for Operations - Present Applications and Future Digital Thread. Presented at the Offshore Technology Conference. 2020;OTC-30553-MS. Available:https://doi.org/10.4043/30553-MS.
9. 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.
10. Khait, Mark, Denis Voskov. Integrated Framework for Modelling of Thermal-Compositional Multiphase Flow in Porous Media. Presented at the SPE Reservoir Simulation Conference. 2019;SPE-193932-MS. Available: https://doi.org/10.2118/193932-MS.
11. 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.
12. 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.
13. Oladipo A, Nwankwo P. Development of a Modified Weymouth Equation for Large Diameter Pipelines. Presented at the SPE Nigeria Annual International Conference and Exhibition. 2020;SPE-203764-MS. Available:https://doi.org/10.2118/203764-MS.

14. 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.
15. Eriavbe, Francis E, Uzoamaka, Ogochukwu O. Understanding the Science of Statistical Log Analysis: A Review of a Practical Workflow Application to Determine Mineral Volume Fraction. Presented at the SPE Nigeria Annual International Conference and Exhibition; 2020;SPE-203681-MS.
Available: https://doi.org/10.2118/203681-MS.
16. Ujjwal, Manu, Pradeep T, Viren K, Manu S. Andrew James. Rapid Insights into Waterflood Management Using Advanced Analytics and Machine Learning. Presented at the SPE/IATMI Asia Pacific Oil & Gas Conference and Exhibition. 2019; SPE-196460-MS.
Available: https://doi.org/10.2118/196460-MS.
17. Yang T, Gulnar Y, Knut U, Ibnu, Hafidz A. Predicting Fluid Properties from Advanced Mud Gas Data. SPE Reservoir Evaluation & Engineering. 2021; SPE-201635-PA.
Available: https://doi.org/10.2118/201635-PA.
18. Kiisi, P. L. Computer Implementation of the Dykstra-Parsons Method of Waterflood Calculation. Presented at the SPE Nigeria Annual International Conference and Exhibition. 2021; SPE-207151-MS. Available: https://doi.org/10.2118/207151-MS.
19. 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.
20. Bhowmik, Subrata and Naik, Harit. Subsea Structure and Pipeline Design Automation Using Digital Field Twin. Presented at the Offshore Technology Conference. 2020;OTC-30909-MS.
Available:https://doi.org/10.4043/30909-MS.
21. Patel, H., Jianlin C., Gautier, N. and Subrata B. (2021). Digital Transformation and Automation of Flow Assurance Engineering Workflows Using Digital Field Twin. Presented at the Offshore Technology Conference, Virtual and Houston, Texas, August. OTC-31017-MS. https://doi.org/10.4043/31017-MS.
22. Khan OH, Samad A, Elfeel MA, Biniwale S, Dandekar R. Integrated Field Management System for LNG Assets: Maximizing Asset Value Through Representative End-To-End Modeling. Presented at the SPE Annual Technical Conference and Exhibition, Dubai, UAE, September. 2021; SPE-205969-MS. Available: https://doi.org/10.2118/205969-MS.
23. Fuad, Iqmal Irsyad Mohammad, Mohd Fuad Ngah Demon, Husiyandi Husni. Automated Real Time Anomaly Detection Model for Operation and Production Data at Scale. Presented at the Abu Dhabi International Petroleum Exhibition & Conference. 2020; SPE-203194-MS.
Available:https://doi.org/10.2118/203194-MS.
24. Adeyemi, Emmanuel A, David O, Ogbe, Xingru Wu. Quantifying the Risks of Wellbore Failure During Drilling Operations Using Bayesian Algorithm. Presented at the SPE Nigeria Annual International Conference and Exhibition. 2020; SPE-203710-MS.
Available: https://doi.org/10.2118/203710-MS.
25. 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.
26. 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.
27. 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.

28. Zeng Y, Liu J, Li G, Rong Yu. Analysis of Ultimate Strength of Steel Plates with Random Pitting Corrosion Using Stochastic FEM. presented at the 30th International Ocean and Polar Engineering Conference. 2020;ISOPE-I-20-4131.

29. Alamu, Olabode A., Pandya Deval A, Warner O. Igor Debacker. ESP Data Analytics: Use of Deep Autoencoders for Intelligent Surveillance of Electric Submersible Pumps. presented at the Offshore Technology Conference. 2020;OTC-30468-MS.
Available: https://doi.org/10.4043/30468-MS

30. 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.

31. Okoro Emmanuel E, Samuel E, Sanni David I, Olatunji, Paul, Igbinedion, Babalola, O. and Oyinkepreye, D. Orodu. 2020. Numerical Based Optimization for Natural Gas Dehydration and Glycol Regeneration. Presented at the SPE Nigeria Annual International Conference and Exhibition. 2020;SPE-203751-MS. Available:https://doi.org/10.2118/203751-MS.

32. 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.

33. Avila, Juan Carlos. Forecasting Intelligent-Well Completion Production through a Coupled Reservoir Model with Custom Functionality. Presented at the Offshore Technology Conference, Houston, Texas, USA. 2020;OTC-30771-MS.
Available:https://doi.org/10.4043/30771-MS.

34. Aghito, Manuel, Bjørkevoll, Knut Steinar. Hybrid Approach for Drilling Automation. Presented at the SPE Norway Subsurface Conference. 2020;SPE-200734-MS.
Available:https://doi.org/10.2118/200734-MS.

35. Konoshonkin D, Gleb S, Ivan M, Aleksandra V, Valeriy R, Vasily D, Boris I. Machine Learning Clustering of Reservoir Heterogeneity with Petrophysical and Production Data. Presented at the SPE Europec. 2020;SPE-200614-MS.
Available: https://doi.org/10.2118/200614-MS

36. Lyall, J., Watt, A., Murphy, C. Automated Pattern Recognition as a Temporary Replacement for Downhole Gauges.Presented at the SPE Asia Pacific Oil & Gas Conference and Exhibition. 2020;SPE-202387-MS.
https://doi.org/10.2118/202387-MS.

37. Prince, Okechukwu Innocent. Application of Machine Learning in Predicting Crude Oil Production Volume. Presented at the SPE Nigeria Annual International Conference and Exhibition. 2021;SPE-207079-MS.https://doi.org/10.2118/207079-MS.

38. Viggen EM, Lasse L, Ioan AM, Svein EM. Automatic Interpretation of Cement Evaluation Logs through Feature Engineering. Presented at the SPE/IADC International Drilling Conference and Exhibition. 2021; SPE/IADC-204057-MS. Available: https://doi.org/10.2118/204057-MS.

39. 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.

40. 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.

41. 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 Exhibition. 2018;SPE-190036-MS.
Available: https://doi.org/10.2118/198863-MS.

42. Cornel, Simon, Gonzalo Vazquez. Use of Big Data and Machine Learning to Optimize Operational Performance and Drill Bit Design. Presented at the SPE Asia Pacific Oil & Gas Conference and Exhibition. 2020;SPE-202243-MS.
Available:https://doi.org/10.2118/202243-MS.

43. Chong E, Shen DO. Automated Selection of Completion Interval Workflow Using Python Script for Probabilistic Range of EUR in a Greenfield Development. Presented at the International Petroleum Technology Conference, Virtual. 2021; IPTC-21412-MS.
Available:https://doi.org/10.2523/IPTC-21412-MS.

44. Naufal AA, Metra S. A Digital Oilfield Comprehensive Study: Automated Intelligent Production Network Optimization. Presented at the SPE/IATMI Asia Pacific Oil & Gas Conference and Exhibition. 2021;SPE-205735-
Available:MS.https://doi.org/10.2118/205735-MS.

45. Srinivasan, A., Modi, G., Agrawal, R., Kumar, V. Application of Advanced Data Analytics for Gas Reservoirs and Wells Management. presented at the SPE Trinidad and Tobago Section Energy Resources Conference Virtual. 2021; SPE-200927-MS. Available: https://doi.org/10.2118/200927-MS.

Peer-review history:
The peer review history for this paper can be accessed here:
https://www.sdiarticle4.com/review-history/76195