Real-Time Modeling of Drill String Torque

Abtin Abbaspour
University of Waterloo
aabbaspour@uwaterloo.ca

MohammadAmin Sedaghat
University of Waterloo
m4sedagh@uwaterloo.ca

Hy Lac Nguyen
University of Waterloo
hl6nguye@uwaterloo.ca

Eihab Abdel-Rahman
University of Waterloo
eihab@uwaterloo.ca

Sarbast Rasheed
University of Waterloo
sarbast.rasheed@uwaterloo.ca

***Abstract*—Onshore and offshore rotary drilling suffers from stick-slip vibrations, a severe mechanical issue that damages equipment and slows operations. Current downhole sensors transmit data too slowly to predict and proactively avoid those events. This paper presents a deep learning approach to predict drill string torque before damage occurs. Using data from the OpenLab drilling simulator, a Long Short-Term Memory (LSTM) neural network was trained to forecast future torque at the drill bit. Relying on only three available sensor inputs, the model predicts torque for 1, 5, and 10 seconds into the future. Model accuracy was evaluated using the coefficient of determination (R^2). Results demonstrate high accuracy for 1-second ($R^2 = 0.974$) and 5-second ($R^2 = 0.868$) forecasts, offering warning window for automated surface controllers to adjust and avoid those instances. While 10-second predictions show some error ($R^2 = 0.696$), this method proves that proactive real-time vibration control is highly effective even with limited sensor data.**

I. Introduction

The process of rotary drilling, onshore or offshore, is an inherently complex industrial process that requires the continuous optimization of mechanical, hydraulic, and thermodynamic systems operating under extreme physical condtions. Modern wellbore trajectories frequently extend several kilometers vertically, creating an environment where the tubular steel “drill string” is subjected to immense stresses. This paper introduces a data-driven methodology designed to predict the torsional dynamics of the drill string in real-time. By utilizing simulated, high-fidelity datasets and advanced recurrent neural network architectures, this research establishes a predictive framework capable of forecasting torque anomalies before they manifest physically at the drill bit, addressing one of the most common and destructive mechanical challenges in modern drilling engineering.

A. Motivation

With emerging applications in the areas of industrial automation, digital twin infrastructure, and predictive maintenance, machine learning (ML) plays an increasingly important role in modern drilling operations [1]. The primary motivation for this research stems from the devastating physical and economic consequences of unmitigated torsional vibrations within the drill string, specifically the phenomenon known as stick-slip [2]. In rotary drilling, the drill string acts as a giant torsional spring, transmitting rotational energy from the surface top drive mechanism down to the drill bit, which is actively engaged with the rock formation [1]. The interaction between the bit and the formation is highly nonlinear, dictated by complex velocity-weakening (Stribeck effect) friction dynamics. When the drill bit engages the rock, static friction at the bit-rock interface can temporarily exceed the dynamic torque supplied from the surface. This causes the bottom hole assembly (BHA) to stall or “stick” while the surface motor continues to rotate at a constant angular velocity [2].

During this sticking phase, immense elastic potential energy accumulates rapidly within the highly flexible drill pipes. Once this stored torsional energy surpasses the static resistive friction threshold of the rock, the bit violently ”slips,” accelerating to rotational speeds that can exceed the surface rotary speed by multiple folds [3]. This erratic cycling between zero velocity and high-frequency, violent spin generates catastrophic torque spikes that propagate like shockwaves upward through the string. The 120 seconds of initial bit-to-rock engagement are notoriously the most unpredictable period of the drilling cycle, setting the stage for long-term mechanical degradation. The physical consequences of sustained stick-slip oscillations are universally recognized as the most destructive among all modes of drilling vibration [4]. Drilling vibrations dramatically accelerates the wear and degradation of drill bit whose components are highly susceptible to brittle fracture and frequently chip or completely shear off due to rapid backward rotation during the violent slip phase. Furthermore, the resulting high-frequency vibrations induce severe cyclic fatigue in the BHA, leading to accelerated tool joint damage, casing wear, washouts, and ultimately, catastrophic twist-offs where the drill string severs completely downhole [4].

To effectively mitigate these destructive torsional dynamics, automated surface control systems require high-fidelity, real-time downhole torque data. A reliable, early warning of incoming torque spikes can be fed directly into automated drilling control systems, allowing top-drive controllers to proactively adjust surface revolutions per minute (RPM) or weight on bit (WOB) to alleviate a stick event or absorb an incoming shock. However, the realization of this proactive control paradigm is obstructed by the severe limitations of current downhole telemetry systems. The industry standard for transmitting data from the BHA to the surface is measurement while drilling (MWD) facilitated by mud pulse telemetry (MPT) with slow data transmission preventing real-time mitigation of downhole damage.

B. Related Works

Physics-Based Models utilize fundamental physical equations to simulate the drill string's behavior, often representing it as a mass-spring-damper system [5]. High-resolution approaches like the Finite Element Method (FEM) models can accurately replicate complex mechanical interactions, such as the Stribeck effect, dry friction, and non-Newtonian fluid damping [6]. However, their reliance on solving complex equations makes them computationally prohibitive for real-time control applications at the rig site. Furthermore, they require precise continuous inputs regarding downhole conditions (like bit wear states and exact formation lithology) that are rarely available during actual operations [7].

To overcome the computational bottlenecks of purely physics-based approaches, traditional machine learning algorithms like Random Forests (RF), Support Vector Machines (SVM), and standard Artificial Neural Networks (ANN) have been widely deployed [8]. These models excel at classification and regression tasks; for example, RF models have demonstrated up to 98% accuracy in identifying the presence of stick-slip vibrations from historical data [8]. The primary limitation of traditional ML in this domain is that it treats data as independent static snapshots. Because these standard algorithms lack the internal architecture to process sequential time-series data, they are generally incapable of forecasting future transient vibration states.

Deep learning architectures, particularly Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks, are specifically designed to handle sequential data and capture long-term temporal dependencies. Research shows LSTMs can successfully forecast downhole torque and vibration events 10 to 30 seconds into the future [9]. Despite their high predictive power, these deep learning models face challenges regarding data sparsity; they typically require extensive downhole sensor arrays to function optimally and can suffer severe performance drops when exposed to high levels of sensor noise. Training deep neural networks requires massive volumes of labeled data encompassing diverse operational failure modes. In the field, catastrophic stick-slip events that lead to twist-offs are relatively rare as operators actively try to avoid them, meaning historical field datasets are fundamentally biased toward normal, benign operating conditions. To overcome the inherent scarcity of labeled, high-frequency fault data, researchers have increasingly turned to digital twins and advanced drilling simulators.

The OpenLab drilling simulator [10], developed by the Norwegian Research Centre (NORCE), stands out for this purpose. Validated against real-world well data, OpenLab provides a cloud-based high-fidelity transient dynamics engine capable of simulating complex fluid hydraulics, thermodynamics, cuttings transport, and the precise mechanical interactions of torque and drag force along the wellbore. By operating via a programmable web application programming interface (API), OpenLab facilitates the rapid generation of clean, time-synced datasets across multiple simulated sensors. This simulated environment allows artificial injection of stochastic noise, initiation of sudden parameter shifts, and trigger of transient stick-slip limit cycles [10]. By training on this heavily augmented, simulated data, machine learning algorithms can be exposed to a much wider distribution of failure states than would ever be available in a single real-world dataset, ensuring the resulting model is robust enough to generalize to the unpredictable physical realities of an offshore rig.

C. Problem Definition

The core problem addressed in this research is the real-time prediction of transient bottom hole assembly (BHA) torque under sparse sensor data conditions to preemptively mitigate destructive torsional stick-slip vibrations. Because acoustic mud pulse telemetry is severely bottlenecked by low bandwidth and high signal latency, estimating the instantaneous downhole conditions in real-time is operationally prohibitive.

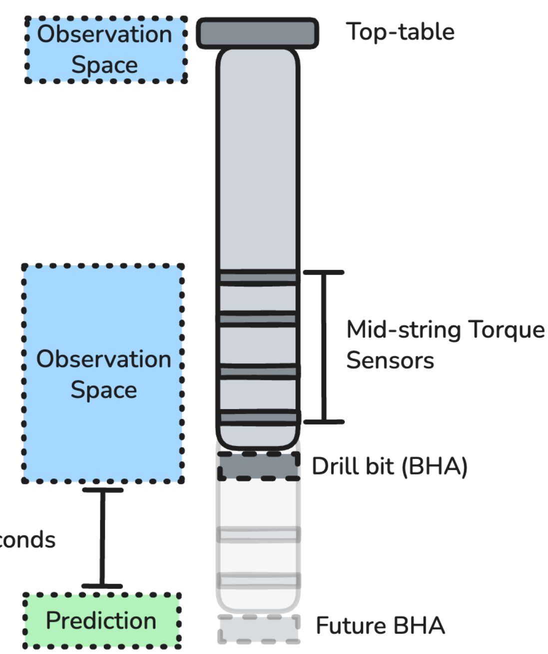

Fig. 1. Visualization of Drill String Inputs and Model Outputs

Therefore, as visualized in Fig. 1, this study explores a Long Short-Term Memory (LSTM) model that processes a sequential observation matrix $X \in \mathbb{R}^{T\times 3}$ containing easily accessible surface torque, mid-string torque, and currently reported BHA torque. This framework aims to directly map these sparse inputs to predict the future scalar BHA torque $y = \hat{T}_{BHA}(t+h)$ at discrete prediction horizons (h) of 1, 5, and 10 seconds, bridging the telemetry latency gap and enabling proactive automated surface control.

II. Methodology

The objective of this study is to forecast future drill-string torque states from a compact temporal representation of the system. Instead of modeling the full spatial torque profile at each timestep, a reduced representation is constructed that captures the torque at the surface and a set of measurements near the drill bit. This representation preserves key boundary information while reducing the dimensionality of the problem.

A. Data

To train and evaluate the machine learning model, data was collected using the OpenLab drilling simulator. The simulator was set up to model a vertical offshore well. It uses a transient torque model to accurately recreate the physical behavior of a rotating drill pipe. The system records data at a rate of 10 Hz, meaning it captures 10 timesteps every second.

To make sure the model can handle the natural variability of real drilling, five different datasets were created. A baseline setup was established using these standard starting values:

- Surface RPM: 50 rpm (0.83 Hz).
- Desired Weight on Bit (WOB): 20 t.
- Top of String Velocity: 0.02 m/s.
- Flow Rate In: 1800 L/min (0.03 m^3/s).

Because the OpenLab simulator is inherently deterministic, it will always produce the same outputs for a given set of inputs. Introducing stochastic noise and sudden parameter shifts breaks away from idealized physics to allow the simulator explore its neighborhood within parameter space, artificially replicating erratic friction dynamics, mechanical fluctuations, and sensor inaccuracies experienced on a live rig. Thus, random triggers were applied during the runs to suddenly change the input settings. Randomized noise was also added to the flow rate and RPM. These sudden triggers happened at random time ranges:

- Trigger 1 between 150 and 250 timesteps
- Trigger 2 between 450 and 600 timesteps
- Trigger 3 between 750 and 900 timesteps

For each trigger, new values for the Flow Rate In and RPM were sampled using a continuous uniform distribution bounded by the operational limits of the rig. Specifically, the Surface RPM was varied between 0.78 and 2.17 Hz and the Flow Rate In between 0.025 and 0.053 m^3/s.

This injected variability ensures the generated datasets accurately captures the uncertainty embedded in offshore drilling, forcing the machine learning model to adapt and generalize to new conditions rather than simply memorize a fixed (deterministic) response.

B. Per-Timestep Feature Representation

For each physical timestep t, the torque measurements along the drill string are ordered from the surface to the bottom of the well. We construct a feature vector, x_t given by

$$\mathbf{x}_t = \left[\tau_t^{top}, \tau_t^{bot,1}, \tau_t^{bot,2}, \ldots, \tau_t^{bot,m}\right] \in \mathbb{R}^{m+1}, \quad (1)$$

where τ_t^{top} denotes the torque at the top of the drill string and $\tau_t^{bot,i}$ denotes the i-th torque value among the final m measurements near the bottom of the string. In our implementation, we set $m = 5$, resulting in a six-dimensional feature vector for each timestep.

Given a sequence of T timesteps, the preprocessing stage produces a temporal feature matrix

$$\mathbf{X}_{seq} \in \mathbb{R}^{T\times(m+1)}. \quad (2)$$

This representation captures both the applied surface torque and the torque response near the drill bit, which together provide a compact summary of the drill string state.

C. Temporal Windowing and Forecast Targets

To model temporal dynamics, we construct training samples using a sliding window approach. Each input sample consists of the previous k timesteps:

$$\mathbf{X}^{(i)} = [\mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_{i+k-1}] \in \mathbb{R}^{k\times(m+1)}. \quad (3)$$

In our experiments we use $k = 15$ past timesteps.

Rather than predicting only the next timestep, the model performs *direct multi-horizon forecasting*. Let the set of prediction horizons be

$$\mathcal{H} = \{10, 50, 100\}. \quad (4)$$

For each input window, the model predicts the torque representation at each future horizon:

$$\mathbf{Y}^{(i)} = \begin{bmatrix} \mathbf{x}_{i+k-1+10} \\ \mathbf{x}_{i+k-1+50} \\ \mathbf{x}_{i+k-1+100} \end{bmatrix} \in \mathbb{R}^{3\times(m+1)}. \quad (5)$$

Direct multi-horizon prediction avoids the error accumulation associated with recursive forecasting strategies and enables the network to learn horizon-specific temporal dynamics.

D. Feature Normalization

Prior to training, both input features and prediction targets are standardized using statistics computed from the training set. Each feature is transformed as

$$\tilde{x} = \frac{x-\mu}{\sigma}, \quad (6)$$

where μ and σ denote the mean and standard deviation estimated from the training data. Feature standardization improves optimization stability and convergence when training neural networks [11].

E. LSTM Forecasting Architecture

The forecasting model is implemented as a stacked Long Short-Term Memory (LSTM) network shown in Fig. 2, which is well suited for modeling temporal dependencies in sequential data [12].

The model receives an input tensor of shape $(k, m+1)$ and outputs predictions for all horizons simultaneously. The network architecture consists of the following components:

- An LSTM layer with 128 hidden units that processes the input sequence and returns hidden states for all timesteps.
- A dropout layer with dropout probability 0.2 to mitigate overfitting [13].

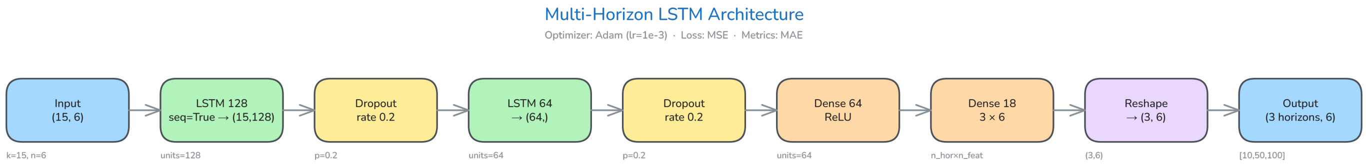

Fig. 2. Multi-Horizon LSTM Architecture.

- A second LSTM layer with 64 hidden units that summarizes the temporal history into a single latent representation.
- A second dropout layer with dropout probability 0.2.
- A fully connected layer with 64 neurons and ReLU activation.
- A final dense output layer producing $H(m+1)$ outputs, which are reshaped into an $(H, m+1)$ prediction matrix.

Let $F = m + 1$ denote the number of features per timestep and H the number of prediction horizons. The model therefore learns a mapping

$$f_\theta : \mathbb{R}^{k \times F} \rightarrow \mathbb{R}^{H \times F}, \tag{7}$$

where θ represents the trainable parameters of the network.

F. Training Procedure

The model is trained using the Adam optimizer [14] with a learning rate of 10^{-3}. The training objective is the mean squared error (MSE) computed across all predicted horizons and output features:

$$\mathcal{L}_{MSE} = \frac{1}{NHF} \sum_{n=1}^{N} \sum_{h=1}^{H} \sum_{f=1}^{F} (\hat{Y}_{n,h,f} - Y_{n,h,f})^2. \tag{8}$$

During training, two regularization strategies are employed:

- **Early stopping**, which halts training if validation loss does not improve for several epochs and restores the best performing model.
- **Learning rate reduction on plateau**, which reduces the optimizer learning rate when validation loss stagnates.

III. Results

To assess generalization across wells, we adopt a cross-well evaluation strategy. The forecasting model was trained on three of the simulated OpenLab datasets and evaluated on the remaining two. This 3-to-2 train-test split ensures the model is evaluated on completely unseen drilling conditions and well dynamics, proving its ability to generalize to new drilling conditions rather than simply memorizing the training data. Prediction accuracy was evaluated using three standard metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2) calculated using the implementation provided in scikit-learn [15].

Fig. 3 displays the model's training history, demonstrating the convergence of both the training and validation loss (MSE) and MAE over 50 epochs. The stable and close tracking between the training and validation curves indicates that the model learned the underlying physical dynamics effectively without severely overfitting to the training samples.

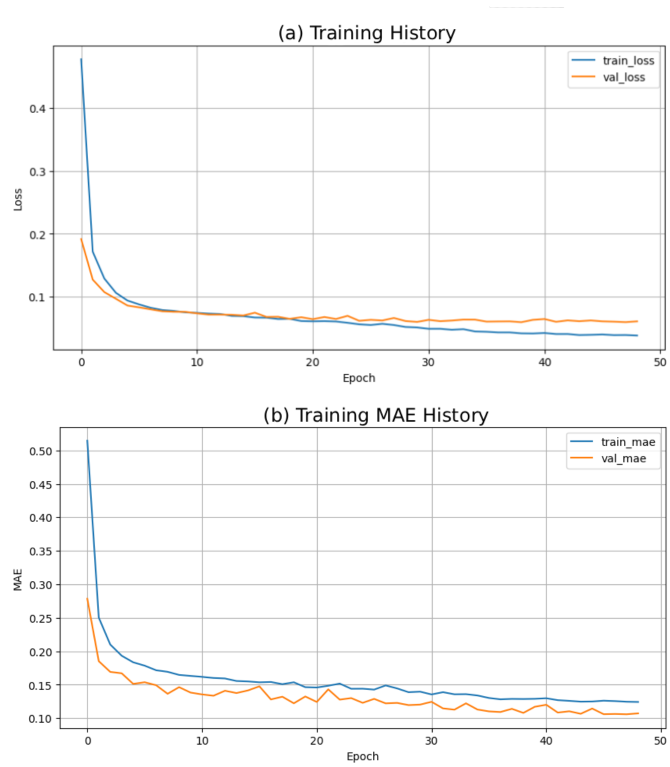

Fig. 3. Training Loss (a) and MAE (b).

The multi-horizon predictions were evaluated at 1-second (10 timesteps), 5-second (50 timesteps), and 10-second (100 timesteps) intervals. The quantitative performance metrics across the test datasets are summarized in Table I.

TABLE I
Model Performance Metrics across Prediction Horizons

Horizon	Timesteps	MSE	MAE	R^2
1 second	10	226.76	10.76	0.974
5 seconds	50	1165.74	22.41	0.868
10 seconds	100	2721.89	36.21	0.696

As the prediction horizon expands, a natural decay in predictive accuracy is observed. At the 1-second horizon, the model performs well and captures the immediate transfer of energy and torque with predicted values following BHA torque closely. This high degree of accuracy ($R^2 = 0.974$) is visually confirmed in Fig. 4, where the prediction tightly tracks the highly transient ground truth data.

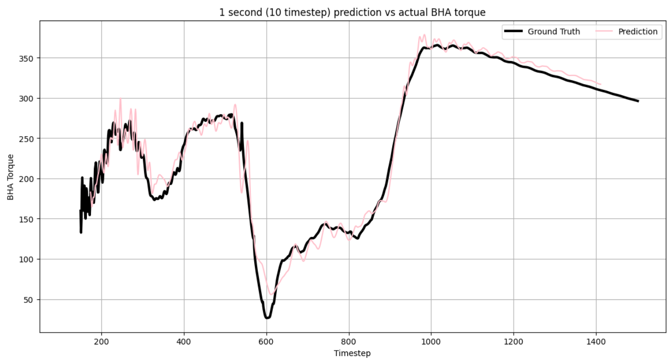

Fig. 4. 1 second (10 timestep) prediction vs actual BHA torque.

For the 5-second horizon shown in Fig. 5, the model still successfully tracks the broader trends but struggles with details. Specifically, the predicted signal exhibits a high frequency content not present in the ground truth.

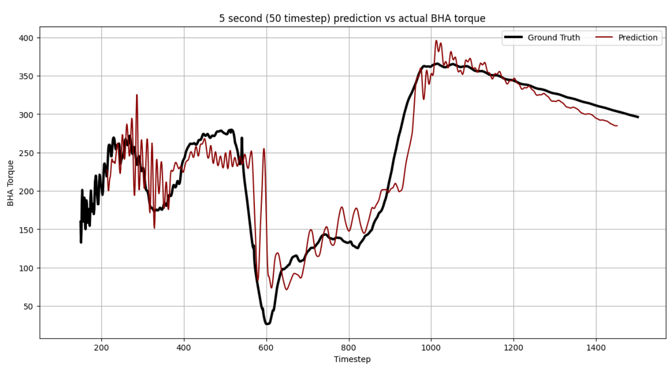

Fig. 5. 5 second (50 timestep) prediction vs actual BHA torque.

At the 10-second horizon, unpredictable physical changes compound over time, leading to a larger deviation from actual torque values. As depicted in Fig. 6, the 10-second horizon model occasionally predicted sudden ”spikes” in torque that did not actually occur in the observed simulator data. These false spikes are a common challenge in time-series forecasting and likely stem from compounding errors and a lack of training data.

IV. Conclusion

This research demonstrates that deep learning architectures, specifically Long Short-Term Memory (LSTM) networks, are highly effective at forecasting the complex torsional dynamics of a drill string in real-time. The high accuracy achieved at the 1-second prediction horizon ($R^2 = 0.974$) serves as a strong proof of concept for predictive drilling analytics. Notably, the results indicate that future BHA torque is highly predictable even when relying on a sparse set of only three core inputs: top table torque, mid-string torque, and current BHA torque. This suggests that an overwhelming array of downhole sensors is not a prerequisite for achieving precise short-term forecasts.

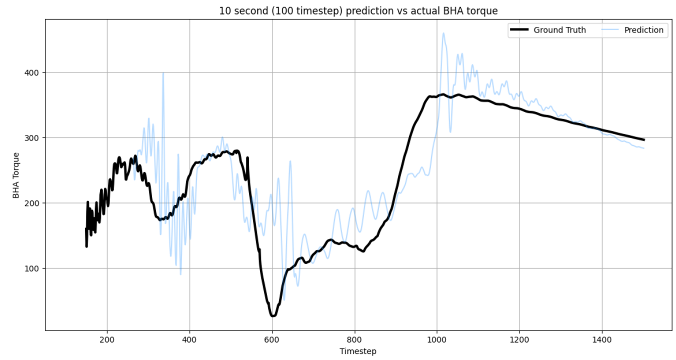

Fig. 6. 10 second (100 timestep) prediction vs actual BHA torque.

The 1-second and 5-second prediction windows offer significant practical utility for offshore drilling operations. Providing a reliable split-second warning of incoming torque spikes allows for the integration of these models into automated drilling control systems. With a lead time of 1 to 5 seconds, automated top-drive controllers can proactively adjust surface parameters to absorb mechanical shocks. This rapid, proactive response enables mitigation of destructive stick-slip vibrations at the exact moment the bit engages the rock formation.

While the short-term models demonstrated robust performance, the 10-second horizon revealed inherent limitations in long-term forecasting. The model often predicted torque surges and dips that did not manifest in the actual data. These errors are a characteristic challenge in time-series forecasting, typically resulting from the compounding of small mathematical errors over longer sequences and a lack of sufficient training data to cover extreme variability.

For the transition of this framework toward field deployment, future research should focus on expanding the training data to include diverse well conditions to ensure the model generalizes across varied drilling environments. Additionally, incorporating additional input features will provide deeper physical context to the LSTM and help mitigate error compounding. Finally, refining the network architecture will be essential to maintaining smooth, reliable torque predictions over horizons exceeding 10 seconds.

V. Acknowledgements

The authors would like to thank Bing Hu and the team at WAT.ai for the valuable support throughout this project. We are also deeply grateful to Professor Maurice Dussault from the University of Waterloo’s Department of Earth and Environmental Sciences for sharing his drilling expertise and for introducing us to his colleagues. Through this connection, we had the privilege of learning from Professor Dan Sui and Professor Bernt Sigve Aadnoy at the University of Stavanger, who provided further drilling insights and connected us with the OpenLab team. Finally, we extend our sincere thanks to Jan Einar Gravdal for his technical guidance and support in using the OpenLab drilling simulator.

REFERENCES

[1] D. Sui and J. Chen, “Robust Control of Drillstring Vibrations: Modeling, Estimation, and Real-Time Considerations,” *Applied Sciences*, vol. 15, no. 24, Dec. 2025.

[2] C. Ejike, I. F. Obuobi, S. Avinu, K. Abid, and C. Teodoriu, “Investigation and Analysis of Influential Parameters in Bottomhole Stick–Slip Calculation during Vertical Drilling Operations,” *Energies*, vol. 17, no. 3, 2024. [Online]. Available: https://www.mdpi.com/1996-1073/17/3/622

[3] R. Riane, M. Z. Doghmane, M. Kidouche, K. F. Tee, and S. Djezzar, “Stick-Slip Vibration Suppression in Drill String Using Observer-Based LQG Controller,” *Sensors*, vol. 22, no. 16, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/16/5979

[4] Z. Hongshan, W. Wenchang, and A. Kaixian, “The impact of multidirectional excitation on the stick–slip vibration characteristics of a drill string,” *Scientific Reports*, vol. 15, no. 1, p. 28483, 2025. [Online]. Available: https://www.nature.com/articles/s41598-025-13880-w

[5] F. S. Boukredera, A. Hadjadj, and M. R. Youcefi, “Drill String Torsional Vibrations Modeling With Dynamic Drill Pipe Properties Measurement and Field Validation,” *Journal of Energy Resources Technology*, vol. 144, no. 024502, 2021. [Online]. Available: https://doi.org/10.1115/1.4051299

[6] M. Z. Doghmane, “Nonlinear Dynamic Modeling and Analysis of Drill Strings Under Stick–Slip Vibrations in Rotary Drilling Systems,” *Energies*, vol. 18, no. 14, 2025. [Online]. Available: https://www.mdpi.com/1996-1073/18/14/3860

[7] R. Wada and T. Kaneko, “Integration of Physics-based Models and Data-driven Models in Drilling Technology,” *Journal of the Japan Petroleum Institute*, vol. 69, no. 1, pp. 1–10, 2026.

[8] R. Saadeldin, H. Gamal, and S. Elkatatny, “Machine Learning Solution for Predicting Vibrations while Drilling the Curve Section,” *ACS Omega*, vol. 8, no. 39, pp. 35 822–35 836, 2023. [Online]. Available: https://doi.org/10.1021/acsomega.3c03413

[9] N. Vishnumolakala, D. M. Murphy, T. Nguyen, E. Z. Losoya, V. R. Kesireddy, and E. Gildin, “Predicting Dysfunction Vibration Events while Drilling Using LSTM Recurrent Neural Networks,” in *SPE/IATMI Asia Pacific Oil & Gas Conference and Exhibition*. OnePetro, Oct. 2021.

[10] N. Saadallah, J. E. Gravdal, R. Ewald, S. Moi, A. Ambrus, B. Daireaux, S. Sivertsen, K. Hellang, R. Shor, D. Sui, S. I. Sandor, M. Chojnacki, and J. Odgaard, “OpenLab: Design and Applications of a Modern Drilling Digitalization Infrastructure,” in *SPE Norway One Day Seminar*. Bergen, Norway: OnePetro, May 2019.

[11] Scikit-learn Developers, “Standardscaler documentation,” 2023. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

[12] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

[14] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference for Learning Representations*, San Diego, 2015.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, “Scikit-learn: Machine learning in python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.