IPTC-24515-MS

Mitigating Nonproductive Time: A Novel Algorithm for Dsl Fault Detection

C. M. Couto-Silva, S. Shetty, A. Olid-Gonzalez, G. Wallez, V. Chatelet, and A. Kohar, SLB

Abstract

Digital slickline (DSL) has been introduced to improve the efficiency of intervention operations in both onshore and offshore wells. DSL cables provide a real-time two-way-telemetry path between the acquisition system at the surface and the downhole sensors. Most nonproductive time (NPT) in DSL operations stems from telemetry issues due to cable faults despite the system's robustness. To address this, we developed a data-driven framework for identifying potential cable damage and its approximate location using the DSL logging telemetry data, including communication signals, pressure, and depth. We tested our method on 992 real-case downhole jobs across almost 30 countries. To validate our method, we compared the method predictions for 60 jobs with labeled potential faults (i.e., cable damage), reaching an accuracy of 98% when considering whether the job has a fault. Thus, our framework enhances cable management, reducing NPT and associated costs.

Introduction

The oil and gas industry consistently pursues optimization and operational efficiency to reduce nonproductive time (NPT) and acquire reliable data during well-intervention tasks. Traditional slicklines and electric lines (e-lines) have long been employed for these tasks, offering basic mechanical services and logging capabilities. The digital slickline (DSL) is a more recent technology, first introduced in the mid-2010s, combining the capabilities of the conventional slickline and e-line (Mahmoud Radwan 2020), meaning that DSL can efficiently perform a wide range of well-intervention operations, including production logging, plug setting, perforation, and more (Loov & Billingham 2014; Razak et al. 2014).

DSL cables provide a path for two-way-telemetry in real time between the acquisition system at the surface and the downhole sensors, including gamma ray, casing collar locator, pressure, temperature, and shock, to name a few. In this article, we call the communication signal sent by the surface equipment to the downhole tool the "downlink". In contrast, the communication upwards (from the tool to the surface sensor) we name the "uplink". Both signals have decibels (dB) as units of measure.

Whilst the telemetry system is robust, we observed (when it occurred) that most of the NPT when using DSL cables is due to coating damage impairing the communication between DSL and the tools. In the presence of coating damage, this communication impact can be noticed when the damaged region comes into contact with a conductive fluid like completion brine. Given these assumptions, it should be possible to detect coating damage by detecting sudden drops in the communication signals according to the depth and estimated location by considering the fluid level depth.

Historically, numerous papers have focused on detecting failures or anomalies in the oil and gas industry. However, to the best of our knowledge, no model for detecting DSL coating damage has been published so far. Usually, anomaly detection using time-series data is challenging, requiring careful data cleaning and a mix of physics and data-driven models (Patri et al. 2014; Martí et al. 2015; Borges Filho et al. 2023).

Therefore, here we proposed a data-driven framework to detect potential faults and their estimated location so they can be remedied as soon as they occur. The framework covers two physics models (detecting the communication loss and the fluid level) and has been tested on 382 well-intervention tasks, successfully identifying faulty cables.

Methods (proposed framework)

Data Acquisition and Preparation

Field engineers upload the DSL telemetry data to a cloud-based system as Log ASCII Standard (LAS) files, which can be accessed using an application programming interface call. In our method, we keep only the files with the latest modification date for each retrieved LAS file. We then parse the LAS files into comma-separated files (CSV) using the lasio Python library, capturing job metadata and measurement data.

As job metadata, we track features like the file name, cable identification (type and serial number), job country, job date, the parsing status (if successful or not, and which error there is for the non-processing status), parsing date, last file modification date, maximum job depth, and the unit of measurement for each curve. These data are then stored in a metadata table, which can be used to perform a check to process only new or modified files to avoid unnecessary computation.

As job measurement data, we process the elapsed time since the job start, depth, pressure, temperature, downlink, and uplink signals. We also ensure that each job has standardized units of measurement. For instance, we standardize all curves: depth in meters, pressure in psi, and temperature in degrees Celsius. Downlink and uplink signals already have standard values in dB ranging from 0 to 80, approximately. We expect a healthy signal of 60 dB for both downlink and uplink signals starting from the surface (depth ~=0).

Data Preprocessing

After we have standard units of measurement for all job data, we can apply standard preprocessing techniques.

As a first step, we filter unexpected data. We remove records with zero or missing pressure values when there is available pressure data. If pressure data is unavailable, we do not apply any filter or correction for the fault location. We also filter observations to keep the ones with a depth greater than 0 and a downlink signal between 5 and 65 dB. We also remove stationary depth points (i.e., consecutive observations with the same depth but distinct elapsed time). After the initial data cleaning, we split the data into run-in-hole (RIH) and pull-out-of-the-hole (POOH) data (curves) by taking the maximum depth by elapsed time. We split into RIH and POOH due to the possible fluid changes in the well, which gives us a linear curve when plotting the telemetry signals by depth (Figure 1).

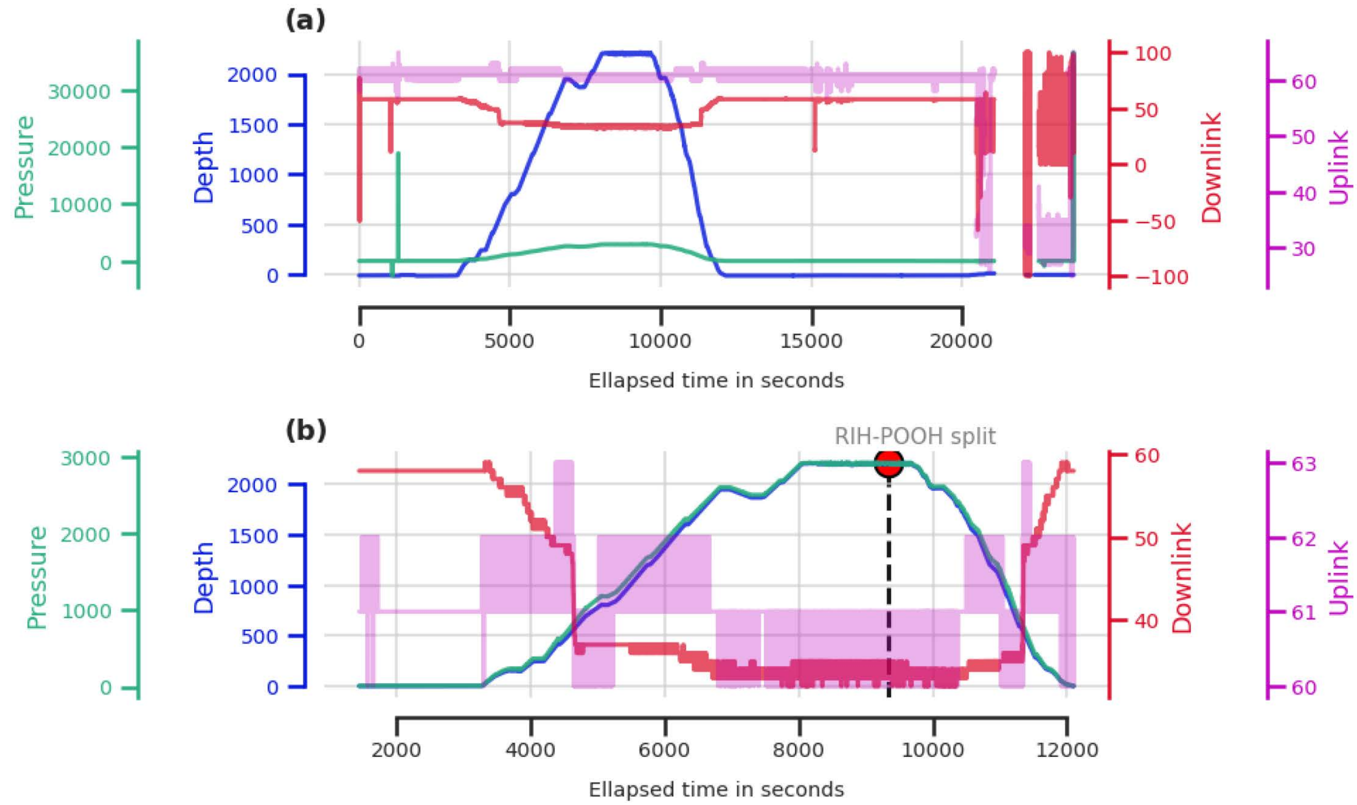

Figure 1—Preprocessing steps. (a) The upper panel shows the telemetry curves by elapsed time. Each curve has its color and scale. (b) The bottom panel shows the cleaned curves, highlighting the maximum depth in meters by elapsed time. Our approach takes this point to split the data into RIH and POOH.

We smooth the telemetry-by-depth curve to remove noise for each RIH and POOH data. To denoise the telemetry data, we first smooth the downlink data using a rolling-window median and then apply a windowed linear regression to replace outliers with the regressed values. We also smooth the pressure data for each RIH and POOH data using the same downlink strategy (smoothing the curve using a rolling window and removing outliers through windowed regression). However, we additionally remove outliers using a distribution histogram-based approach and observations with the pressure-by-depth slopes too far from an expected range.

Modeling

We developed two data-driven physics models to assess if a cable has a fault and where the fault occurred, respectively. The first model aims to identify sudden drops in the job downlink-by-depth curve. We applied a rolling-median approach for each RIH and POOH datum to compute the differences (delta) between the consecutive windows (w_i-$w\_(i-1)$). We add a fault label to the windows (w_i) where the delta exceeds a certain threshold (e.g., 3), storing the observed difference. We then cluster consecutive or nearby fault labels by providing the maximum delta and the average depth to remove noise.

With the above approach, the model predicts if a fault happened, but its location might be inaccurate given the absence of the fluid level depth. We should detect the fluid level to properly find the fault depth in the DSL cable, starting from the end-tip. Figure 2 illustrates why the fluid level normalization works.

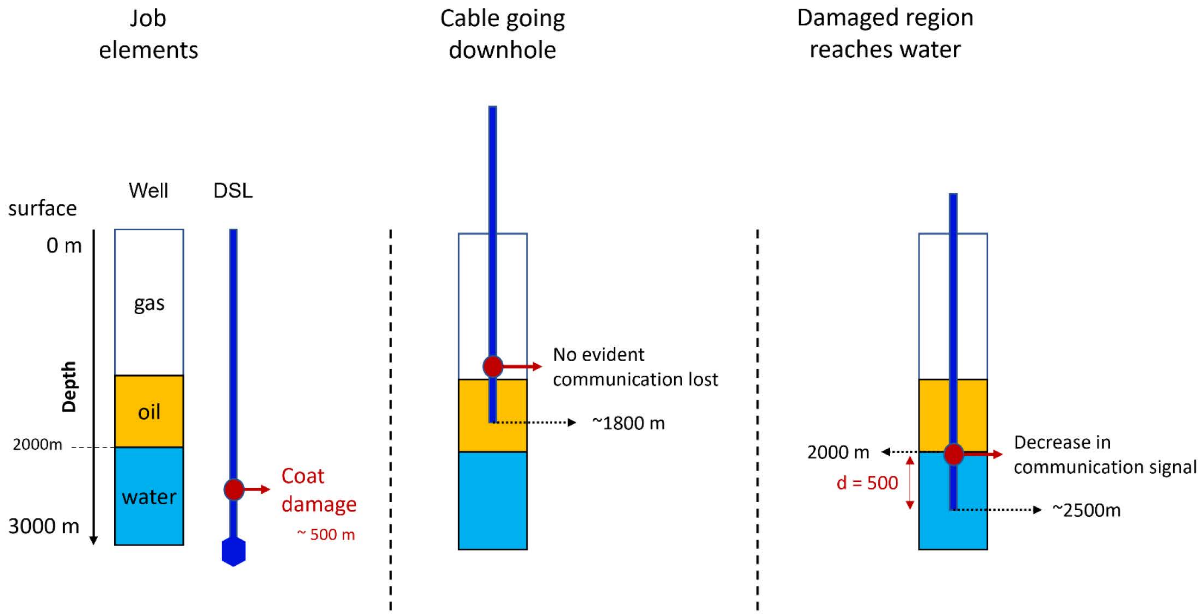

Figure 2—Logic for detecting DSL coating damage. The left panel shows a fictitious well with gas, oil, and water that is 3000-m deep. It also shows a thin blue bar representing a DSL with coating damage around 500 m from its tip. The middle panel shows the DSL downhole up to 1800-m deep, without evident communication loss between the surface system and the downhole tool. The right panel shows the DSL when the damaged coating region reaches the water level, where we should be able to see a sudden decrease in the downlink and uplink signals. The difference between the DSL tip and the water level provides the precise damage (fault) location.

To detect the fluid level, we apply further cleaning steps in the pressure data. The goal is to detect a slope bigger than 1.35 (pressure in psi/depth in meters) since a theoretical slope of 1.4 is expected in the presence of water. We first aggregate the cleaned pressure by taking the average of each 10 m, interpolate the missing values linearly, and then compute the pressure-by-depth slopes. We consider only slopes ranging from 0 to 1.8 and apply the Hampel filter to the slopes to ignore potential noise. We finally report the first depth where the slope is equal to or greater than the selected slope threshold (1.35). If no pressure data are available near the surface, we do not compute the fluid level since it might be inaccurate (the fluid might exist long before the capture start of pressure data). The logic for the fault and water-level detection models is shown in Figure 3.

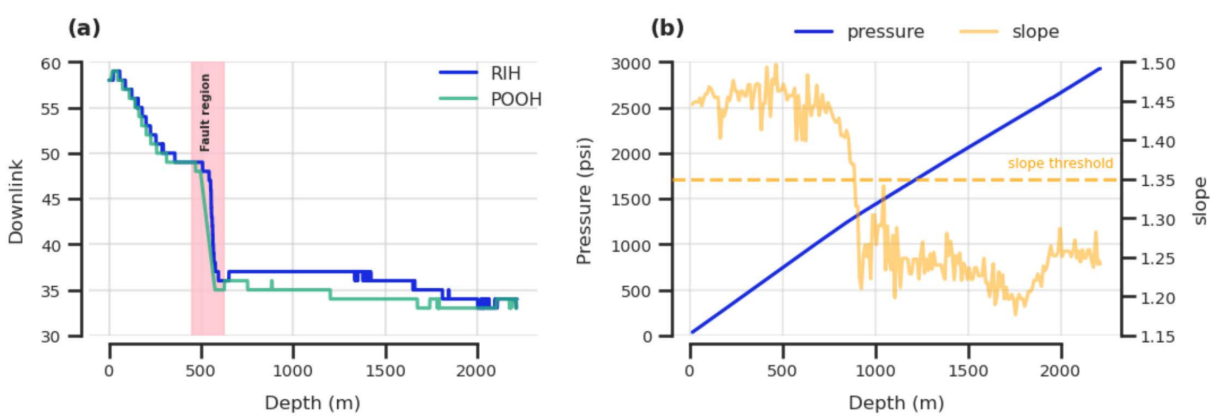

Figure 3—Fault and water-level detection models. (a) Shows the downlink-by-depth curve for both RIH and POOH data. The Pink shadow shows the expected fault region. All nearby detected faults for RIH and POOH are clustered, allowing the model to output only the fault region. (b) Shows an example of the pressure-by-depth curve for RIH data. The model reports the first depth with a slope greater than 1.35 (dashed yellow line; slope values in the right axis). The model first computes RIH and POOH's fault region and water level separately. Then, the model applies the water-level correction and clusters nearby fault depths between RIH and POOH.

Data Post-Processing

After computing the fault and water level predictions, we normalize the fault depth by subtracting the water level depth, and then we cluster the nearby faults considering both RIH and POOH data using the same strategy we applied for clustering individual RIH and POOH faults. We categorize the faults (delta for the rolling median approach) into values ranging from 1 to 5, where 5 is the most severe fault. We consider faults above 3 as high severity (i.e., potential damage with a concern for future operations). Finally, we store the fault depth, level (delta), normalized severity, and prediction date, as well as metadata like water level, a flag about the presence of RIH and POOH data, and a categorical column for both RIH and POOH data telling if the fluid level was analyzed or not (and why not). Processed jobs are not processed again unless the file modification date changes to avoid unnecessary computation.

Model Evaluation

Due to the absence of failure labels for each job, we manually labeled P and fluid levels for 50 random jobs analyzed by the model plus 10 failed jobs according to the model prediction. We additionally labeled these 10 jobs to validate further the positive cases since there are fewer jobs with a high-severity failure. To label the high-severity failure and expected fluid change, the subject matter expert looked at the raw and cleaned telemetry-by-depth and pressure-by-depth curves, providing the expected RIH and POOH high-severity fault and fluid level depths for all job data. Then, we built a Python script to analyze the actual and predicted faults to validate our model. We labeled the instance as a truly positive prediction if the predicted fault depth was close enough (100 m) to the actual fault depth.

Given the labeled data, we first aggregated each job's RIH and POOH faults to tell if there were any true high-severity faults. With this, we could compute a list of true positive, true negative, false positive, and false negative predictions. This approach allows us to assess whether the model accurately detected jobs as failures, regardless of location. We then repeated this logic for the water level detection. Finally, we computed a full match of RIH, POOH, and water level depths for the label and the predictions.

We computed four metrics for validating the results: accuracy, recall, precision, and f1-score. The accuracy tells us how many correct predictions there are. The recall assesses how many faults were correctly detected, while the precision assesses how many predicted faults have been correctly identified. The f1-score is a harmonic mean between recall and precision. Although we aim to increase both recall and precision, there is a trade-off between them in which when one increases, the other decreases. In the oil and gas industry, increasing recall as much as possible without impacting the precision is usually preferable because false negatives (missing to report a fault) can be more expensive than false positives. A damaged cable sent to a job due to a false negative prediction can cause more NPT than an unnecessary check or maintenance (false positive).

Results

We analyzed all job files over a defined period, totaling 992 intervention tasks across almost 30 countries (anonymized data). From the total files, 926 were correctly parsed to CSV, with 66 unable to be parsed due to missing ASCII data (the LAS section with telemetry data). From the 926 parsed files, 382 have been analyzed. Most jobs did not show enough data after applying the required preprocessing filters to predict faults. Of 382 analyzed jobs, 153 files had at least one fault, and 128 were classified as high severity (Figure 4).

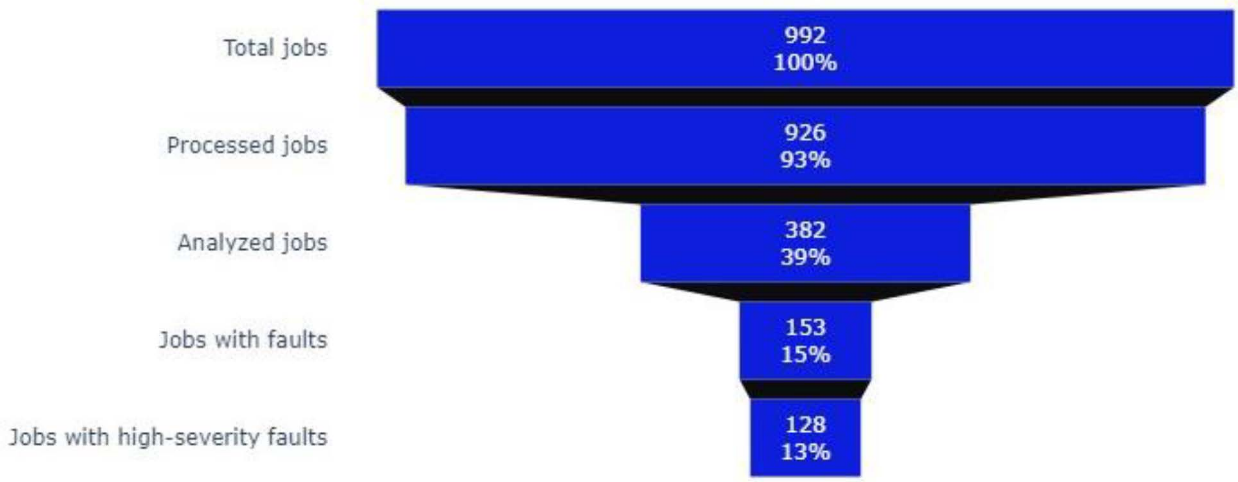

Figure 4—Funnel from received jobs up to jobs with high-severity faults.

More than 80% of the analyzed jobs occurred in four out of 14 countries (Figure 5). Most jobs (>90%) had a maximum depth of less than 4000 m, 9000 psi, and 7 total hours in the job (calculated by taking the difference between the start and end job time) (Figure 6).

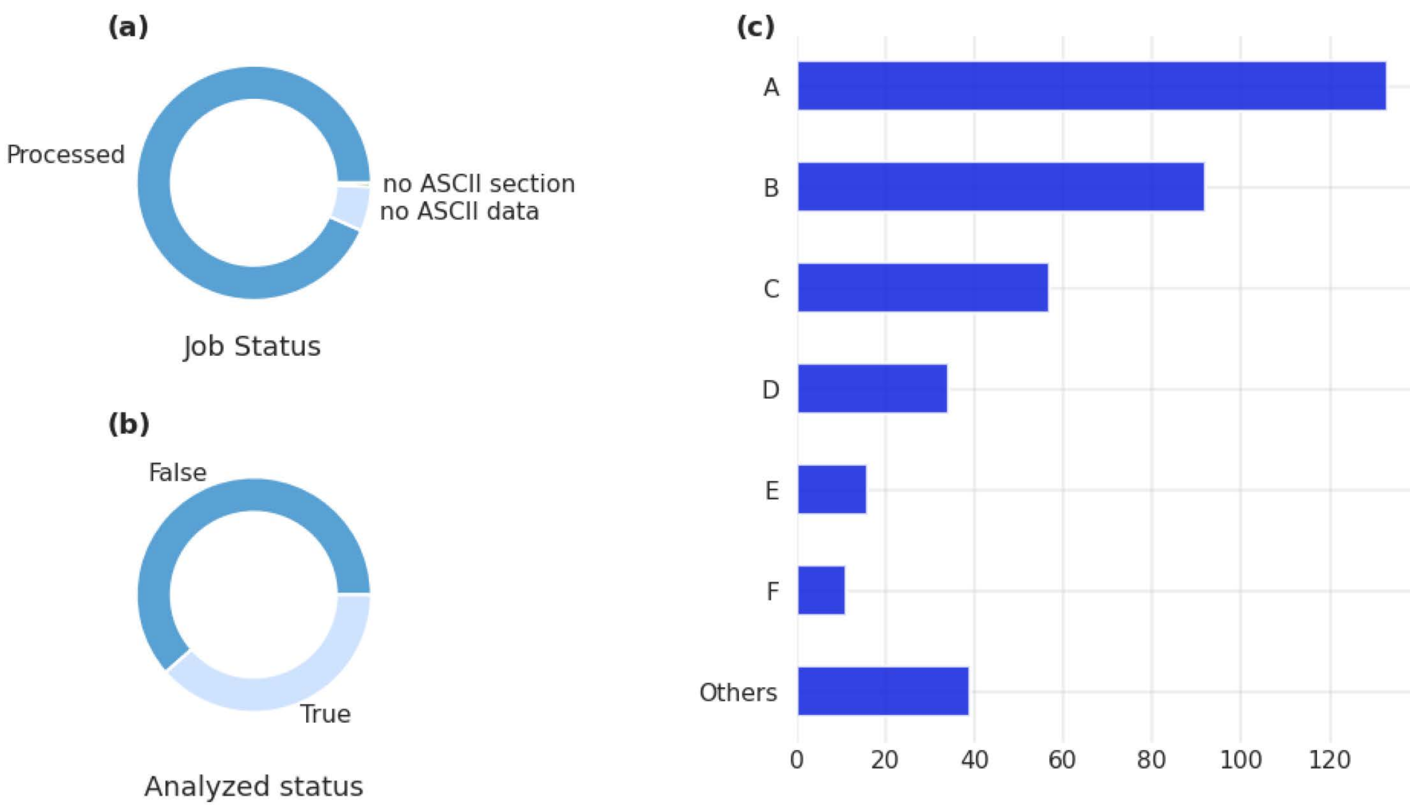

Figure 5—Proportions of processed and analyzed jobs. (a) The proportion of LAS files properly processed and parsed to CSV. (b) The proportions of processed jobs from the previous step analyzed by the model. It highlights that most files were not analyzed due to missing data before or after the data cleansing. (c) The proportion of analyzed jobs per country (anonymized), countries with fewer data as "Others".

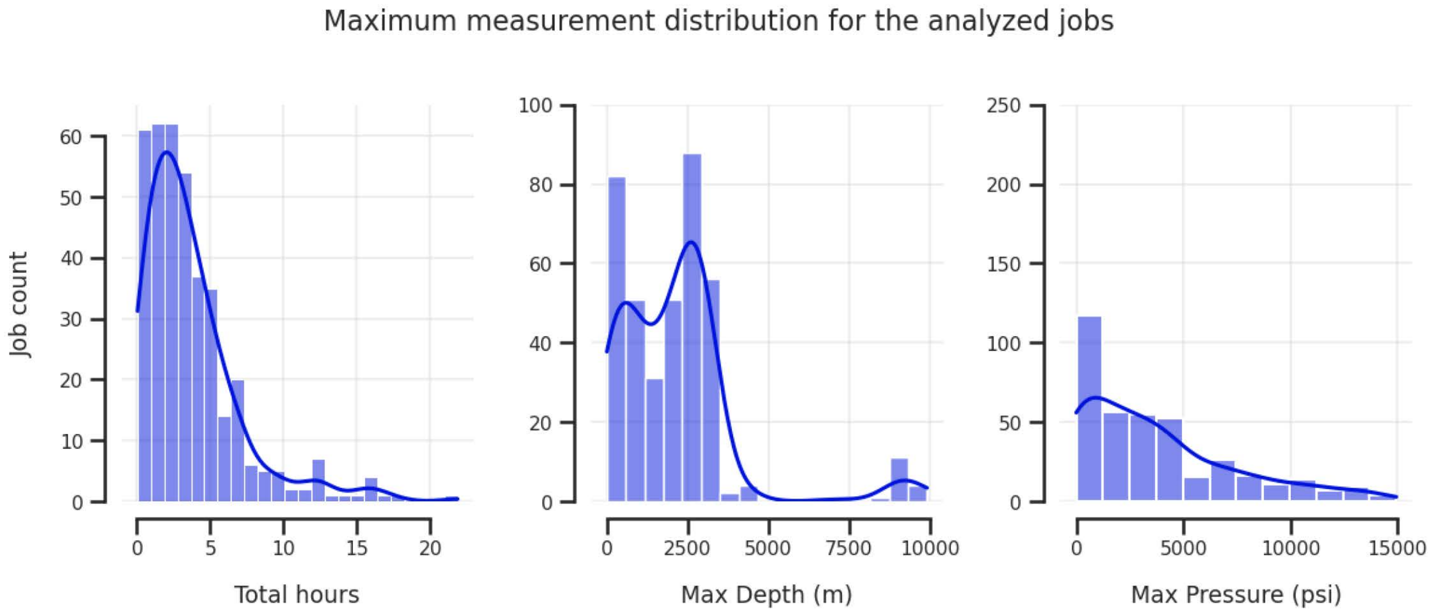

Figure 6—Distribution of the aggregated measures for each job data. Y-axes show the job count for the three plots.

Most of the predicted failed jobs did not have the water level detection available, mainly due to missing pressure data either in its entirety or near the well surface. In such cases, we use the model to alert about a possible fault, but it is not possible to provide an accurate location for the coating damage (Figure 7a). We could also notice that the high-severity (4 and 5 normalized severity) downlink drop ranged from 6 to almost 50 (Figure 7b).

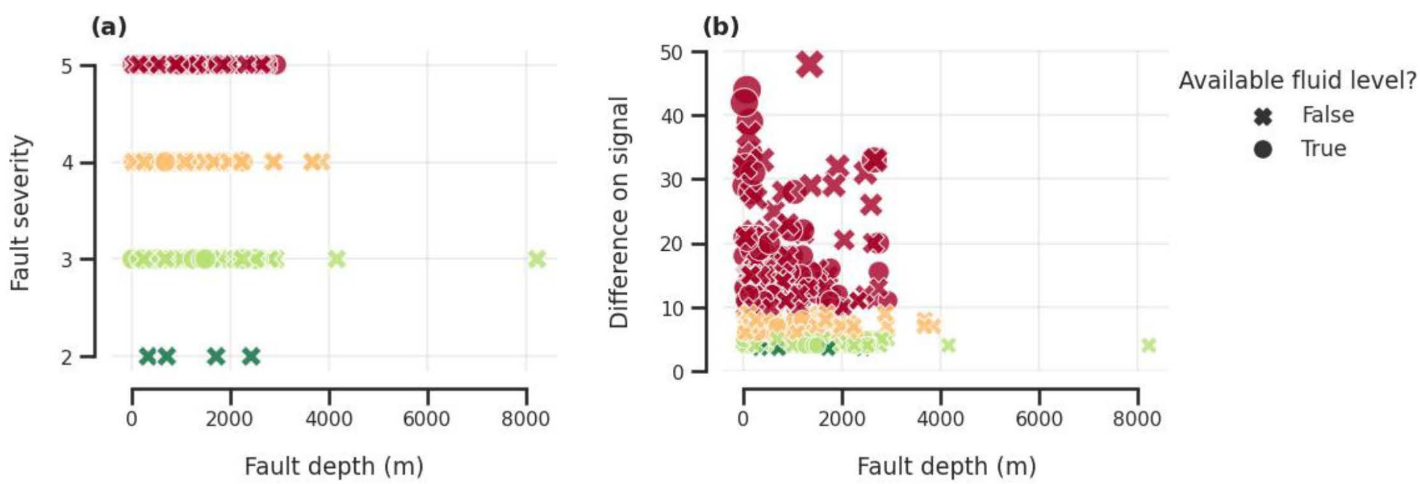

Figure 7—Fault severity and depths in meters. (a) Fault severity by depth. (b) Downlink delta (drop) by depth. The circles represent faults with predicted water levels for both (a) and (b), while the cross means the opposite. The higher number of crosses is directly related to missing pressure data.

To evaluate the model, we first computed the accuracy, recall, precision, and f1-score to assess if the job has a failure, regardless of its location (therefore, not considering the fluid level detection). This was done for a sample of 60 jobs (50 randomly selected jobs plus 10 predicted failed jobs). Our results were 98% accuracy, 100% recall, 96% precision, and 98% f1-score (Table 1). Only one false positive occurred out of the 60 job files. The equivalent approach for the water level detection is 95% accuracy, with one false positive and two false negatives. The overall recall for correctly detecting the presence of fluid is 91%, and 97% for detecting the absence of fluid. We labeled the non-available fluid as "no fluid" given the absence of the required pressure data. Therefore, even if some curves might have a conductive fluid level, we cannot detect it due to the missing data, and we did not penalize it, considering such cases as the "no fluid" label. Also, we did not report water level detection for jobs that did not fail according to the model prediction since our model is currently built to output the water level only when a failure is present on either RIH or POOH data.

Table 1—Metrics for model validation

Approach	Accuracy	Precision	Recall	f1-score
Any faults	0.98	0.96	1.00	0.98
Any detected fluid level	0.95	0.95	0.91	0.93
Full fault match	0.90			
Full fault match + fluid level	0.85			

Note: The fluid level detection might be inaccurate due to missing or non-evaluated pressure data.

Finally, we provide a further evaluation to assess whether the algorithm correctly detected all labeled RIH and POOH faults and the water level. A true positive prediction here means that every RIH, POOH, and water level depth was correctly detected (allowing 100 m of difference between the actual and predicted faults). We achieve 90% when considering a perfect match between the label RIH and POOH fault depths, ignoring the fluid levels. When considering the fluid level, the resulting accuracy is 85%.

Given the success rate observed on the historical data, the algorithm has been deployed to process and predict failures on new job data, automatically creating a work order for the DSL asset in case of a predicted high-severity fault. If pressure data are available, the approximate location is also provided. All aggregated data per job file is shown in the company dashboard so the engineers can visualize the predicted faults and locations.

Discussion

The proposed framework was applied and successfully validated using actual data. Most files were not analyzed due to missing data before or after the data cleaning step. An additional reason for non-analyzed files is the existence of job files with testing or simulated data. It is possible to note that even for the analyzed jobs, many do not have pressure data near the surface.

The missing pressure data makes providing the proper fault location more challenging. Most of the wrong predictions were due to the fluid level detection or gaps in the telemetry data due to the applied filters. Unexpected patterns can also lead to wrong predictions. Therefore, inspecting the data to spot unexpected patterns is essential to correctly handle these patterns in the model pipeline. Depending on the observation, the data scientist might update the algorithm to clean the data accordingly or ignore rare situations. However, providing a label and link to these jobs is a good practice so the jobs can be tracked and inspected later when necessary.

The framework has some assumptions that should be followed to provide good predictions. For instance, the model expects data (telemetry curves) from a single job per file, meaning one downhole run to be split into RIH and POOH data. It also expects the same unit of measure for the target channels (downlink, uplink, and pressure). Downlink and uplink signals should have a fixed range of values, starting from approximately 65 dB. Noisy curves should be handled as proposed here or using an alternative best suited to the target data. Given the following assumptions, this framework can be used or adapted to failure detection. Using the proposed framework, we support field engineers in providing better cable management by using healthy cables, benefiting from better communication and telemetry logging for the downhole intervention tasks. DSL with predicted damage and approximate location can be inspected and fixed accordingly.

Conclusions

Our framework has successfully identified faults in real-case scenarios. Using a data-driven physics model instead of a classical supervised model, we could predict faults even in the absence of historically labeled data, and the interpretation is also evident since the high-severity fault necessarily means a large drop occurred in the tool communication signal.

The efficiency of the proposed framework also highlights the importance of good practices for data integrity. Given the reliable data and statistical techniques, we have shown that it is possible to build efficient algorithms for detecting failures on DSL. Although the specificities of each data kind might change, our proposed framework can help build a similar approach for detecting failures, reducing NPT and their associated cost.

Abbreviations

CSV Comma-separated values
DSL Digital slickline
LAS Log ASCII Standard
NPT Nonproductive time

References

Borges Filho, M. N., Mello, T. P., Scheid, C. M., et al 2023. Real-Time Anomaly Detection Methodology for Drilling Fluids Properties. Paper presented at the SPE/IADC International Drilling Conference and Exhibition, Stavanger, Norway, 7–9 March. SPE-212443-MS. https://doi.org/10.2118/212443-MS.

Loov, R. Billingham, M. 2014. Beyond Logging: Slickline Operations Can Now Provide a More Efficient and Cost Effective Alternative to Traditional Intervention Operations. Paper presented at the SPE/ICoTA Coiled Tubing and Well Intervention Conference and Exhibition, The Woodlands, Texas, USA, March. SPE-168252-MS. https://doi.org/10.2118/168252-MS

Mahmoud Radwan, M. F. (2020). Safe and Economic Attractive Rigless Operations Using a Digital Slickline in Unmanned Platform with Low Structure Loads and Spacing. Paper presented at the Abu Dhabi International Petroleum Exhibition & Conference, Abu Dhabi, UAE, 9–12 November. SPE-202857-MS. https://doi.org/10.2118/202857-MS.

Martí, L., Sanchez-Pi, N., Molina, J., et al 2015. Anomaly Detection Based on Sensor Data in Petroleum Industry Applications. *Sensors* **15** (2): 2774–2797. https://doi.org/10.3390/s150202774.

Patri, O. P., Panangadan, A. V., Chelmis, C., et al 2014. Predicting Failures from Oilfield Sensor Data using Time Series Shapelets. Paper presented at the SPE Annual Technical Conference and Exhibition, Amsterdam, The Netherlands, 27–29 October. SPE. https://doi.org/10.2118/170680-MS.

Razak, M. A., Dasan, L. M., Effendi, S. M., et al 2014. Real Time Slickline: Unlocking Additional Production with Reduced Uncertainties in Limited Space Platforms. Paper presented at the SPE/ICoTA Coiled Tubing and Well Intervention Conference and Exhibition, The Woodlands, Texas, USA, 25–26 March. SPE-168252-MS. https://doi.org/10.2118/168252-MS.