Software Design of a Data Acquisition System for the Hydraulic Fracturing Experiment of Rock at Tongkuangyu Copper Mine

Jiang Jingjie¹,²,³, Peng Hua¹,²,³, Ma Xiumin¹,²,³, Sun Yao*¹,²,³

¹ Institute of Geomechanics, China Academy of Geological Science, MinzudaxueNanlu, No. 11, Beijing 100081, China
² Key Laboratory of Neotectonic Movement and Geohazard, Institute of Geomechanics, MinzudaxueNanlu, No. 11, Beijing 100081, China
³ Observation and Research Station of Crustal Stress and Strain in Beijing, Ministry of Natural Resources, MinzudaxueNanlu, No. 11, Beijing 100081, China

Abstract. To cooperate with the research on the fragmentation pre-conditioning technology of large hard rock in natural caving mining, we perform a large flow hydraulic fracturing test in Tongkuangyu Copper Mine to investigate the relationship between the occurrence and expansion of fractures in ore bodies and the pressure and flow of water injection. Five sensors of four different types are used in the test. Each sensor uses independent acquisition software. The operation is cumbersome and complex, and data synchronization is difficult and can be easily lost. Therefore, an integrated data acquisition software is needed. The pressure sensor used in the test adopts a non-standard communication protocol, which cannot be used in secondary development by configuration software. Thus, a primary development under the Windows platform using C++ is adopted. The data acquisition module is developed and deployed in background service mode and automatically runs after the acquisition computer OS started. Each communication link establishes an independent data read thread, which reads and analyses the data stream according to the protocol and pushes it to the data processing thread. Such a thread saves the data and provides data service to the data display unit through a TCP/IP service port. All threads in the data acquisition module are synchronized by event signals, and the data read threads have high priority in them to avoid data loss. Multiple data display units are allowed to be connected to the data acquisition module at the same time, which is conducive to the coordination among the field staff. The data display unit shows the field pressure, flow, temperature, and water level. Key data are also developed using multi-thread technology to keep a friendly operation. Such data can be deployed in any computer connected with the acquisition module to assist the staff in performing any operation in the test. Through the field work in a complex environment, this software for data acquisition system can manage types of sensors, with flexible configuration, reliable operation, and convenient expansion and provide further support for the field investigation work.

1. Introduction
Since the 1980s, the Tongkuangyu Copper Mine has been using the block caving method and has accumulated rich theoretical and practical experiences. Considering that the transfer into deep mining
can face problems, such as poor rock collapsibility, large block increase, and floor rock burst, conducting research on the pre-conditioning technology of hydraulic fracturing rock in advance is necessary. To fully understand the mechanism and field operation technology of hydraulic fracturing rock pre-conditioning and to provide technical support for the safe and efficient production of the mine, our institute has participated in the implementation of large flow hydraulic fracturing test for investigating the relationship between the occurrence and expansion of fractures in ore bodies and the pressure and flow during the water injection process.

During the test, two packers and a connecting rod form a combination according to the form and diameter of the borehole. The combination is settled at the specific position of the fracturing section by using a drill pipe. High pressure water is injected into the packers to make them expand and contact with the borehole wall to form a sealed fracturing section. Then, water is injected into the fracturing section for fracturing operation. In addition, the working parameters, including flow, pressure, and temperature, are recorded during the whole test. After the field test, the recorded parameters and micro-seismic data are analyzed to obtain rock fracture initiation pressure, fracture orientation, and fracture expansion range and determine the relationship between the effect of hydraulic fracturing rock pre-conditioning and working parameters. The analysis results can help experts analyze and formulate the later work plan.

Five sensors of four different types are used in the construction site, such as two flow sensors, one pressure sensor, one temperature sensor, and one water level sensor. These sensors are not only numerous but also from different manufacturers using different communication protocols. If each sensor uses independent acquisition software, then the human–machine interaction will be complicated to operate. Moreover, the format of the results data is complex and easily leads to data loss. The subsequent data processing is also difficult. Therefore, we establish an integrated data acquisition software.

2. Requirements of data acquisition software for hydraulic fracturing rock pre-conditioning
The deployment structure of sensors and data acquisition computers in the test is shown in Figure 1. The pressure parameter is the most sensitive to the rock fracture during the testing process. The data sampling rate of the pressure sensor is as high as 100 Hz. Given that the query mode of the standard MODBUS protocol cannot meet the needs of data transmission, the pressure sensor actively sends data to a computer by using a non-standard data protocol through the independent RS485 data transmission channel. The protocol is extended from MODBUS ASCII mode, which is convenient for debugging. The flow meter is used to record the amount of water injected and discharged from the working section. It is also used to analyze the incubation and closure of the fractures. The temperature sensor is used to test the temperature of water injected during the test for the temperature correction of measurement results. The water level sensor monitors the remaining water in the
reservoir and reminds the staff to add water in time to ensure the normal flow of the test. The flow, temperature, and water level parameter do not change dramatically during the testing process and thus can be queried on the same RS485 bus by following the MODBUS RTU transmission protocol.

In general, no RS485 serial port exists on the computer, considering that the stability of using a USB converter in the complex environment may be poor. Thus, the ISS-4 serial port server is used to map the serial port to a TCP/IP port to establish the connection between the sensor and the acquisition computer.

According to the requirements of the test, the necessary functions of the data acquisition system include:
1. Manage the communication connection between the sensors and the acquisition computer and read and parse the data acquired by the sensors;
2. Display the measurement data of each sensor by text and curve in the field to help operators perform the test;
3. Save the data to a file in an appropriate format for subsequent analysis.

In addition to the above necessary functions, the data acquisition software should also realize the data log file playback, error data cleaning, data curve image generation, and other auxiliary functions to support the basic data analysis.

3. Implementation of the data acquisition system software

Accurate data, timely display, convenient configuration, and strong fault tolerance are the important requirements for this data acquisition system. LabVIEW, MATLAB, configuration software (SCADA), and other development tools can effectively achieve the data acquisition software design. However, the sampling rate of pressure sensor is high, and the non-standard data communication protocol is used in this application. A new driver must thus be developed to use these tools, which are relatively difficult to develop and debug. On the basis of the previous development foundation in some projects of our group, this data acquisition software adopts a complete development scheme, uses C++ language to ensure the execution efficiency of the program, and uses multi-threading technology to improve the real-time performance. The software supports running under the Windows operating system and can be ported to other platforms if necessary.

The structure of the data acquisition software is illustrated in Figure 2. The foreground dialog display program can show the acquired data, save the result data, and play back the saved data well. It is responsible for the interaction between the data acquisition system and the operators or data analysts. Human–computer interaction increases the possibility of data loss or time error of data acquisition caused by an unexpected system block, and the background service program has great advantages in this aspect. The background data acquisition service not only has anti-interference during the data acquisition process but also can be simply configured by the configuration file. It improves the fault tolerance of the data acquisition system. The combination of the two modules perfectly realizes the requirements of the data acquisition system software.

![Figure 2. Structure of the data acquisition software](image)
3.1. Design of the background data acquisition service

Microsoft Windows services are long-running executable applications created by users or operating systems that run in their own Windows sessions. These services can be automatically started when computers boot, can be paused and restarted, and do not show any user interface. These features make services ideal for use on servers or whenever users need a long-running functionality, which does not interfere with other users who are working on the same computer and does not affect other applications or be interfered by them. Therefore, Microsoft Windows services are especially suitable for data acquisition operation in this test.

The functions of the data acquisition service include: managing the communication connection of each sensor, sending commands to the sensor as needed, analyzing the data stream returned from the sensor, checking the validity of the data, saving the complete data stream as a log file, and sending it to the foreground dialog display program. In general, the sensors used in the test are non-intelligent, and no time information is contained in the data flow. The data acquisition service needs to add time tick to the data stream and regularize it before saving and sending.

A certain amount of system API is called for data integrity checking, saving, sending, and other operations, which may be blocked by the operating system. A sensor’s data stream may not be handled in time if a single thread is used, and the data time error may occur. Multi-threading technology can solve this problem well by establishing an independent data stream parsing thread and a data transmitting thread. The data stream parsing thread receives the data stream from sensors and code in the data receiving time and then queues it into the Data Saving Queue. The data transmitting thread reads data from the queue, saves data into the log file, and sends them to the foreground dialog display program. Blocking operations, such as write files, in the data transmitting thread do not affect the data stream that receives the thread directly, making the real-time parsing of data in possible. The Data Saving Queue may be accessed by two threads at the same time. A Semaphore object is needed to block the editing request of an opposite thread for avoiding access exception. The block may interrupt the data stream parsing thread, and an algorithm must be designed to avoid it.

The Data Saving Queue can be saved as a linked list data structure, and the “add to tail” operation has no effect on the existing memory. It does not have to block the opposite thread when the data transmitting thread reads out data if the data stream parsing thread only appends to the Data Saving Queue. Obviously, stale data must be removed; if the data stream parsing thread performs this work, then it

![Figure 3. Flow chart of the data transmitting thread](image-url)
needs to append to the tail and remove the head. The data transmitting thread must also block all Data Saving Queue operations when reading the content of the linked list. The data stream parsing thread is likely blocked, although most blocks are unnecessary. If the stale data are removed only in the data transmitting thread, then there will be no block when reading, which occupies most of the operation time; the data stream parsing thread will not be blocked frequently; and the time tick will be accurate. The flow chart of the data transmitting thread is illustrated in Figure 3, whereas the flow chart of the data stream parsing thread is shown in Figure 4.

**Figure 4.** Flow chart of the data stream parsing thread

The ASCII Message Frame (Figure 5) contains a colon as the start symbol and two characters “CR, LF” as the end symbols. The “Address” can be used to identify data sources, and “LRC” is used for data verification to ensure data integrity. The ASCII Message Frame is used not only for data transmission but also suited for log data file with multiple heterogeneous data with high efficiency. Thus, the system uses it as the standard data frame. The data stream parsing thread obtains the accurate data time, adds time tick into the data field, and then encodes the data stream to make an

| Item     | Start | Address | Function | Data                        | LRC | End       |
|----------|-------|---------|----------|-----------------------------|-----|-----------|
| Length   | 1 char| 2 chars | 2 chars  | 0 up to 2×252 char(s)       | 2 chars | 2 chars   |
| Value    |       |         |          |                             |     | CR, LF    |

**Figure 5.** ASCII Message Frame
ASCII Message Frame to save log files and send them to the foreground dialog display program.

The “Function” and “Data” in the ASCII Message Frame make up a protocol data unit (PDU), and a typical response PDU from the sensor used in our application consists of a “Function code,” a “Byte count,” and a “Register value.” “Time tick” is added after the “Byte count” whose value is adjusted (Figure 6). We should note that “Time tick” has a four-byte length and records the millisecond from the beginning of today. One byte in the PDU is two bytes in the ASCII Message Frame, and the maximum count of the Register value is two less than the standard MODBUS protocol after adding the time tick. At most, 124 registers are allowed in one ASCII Message Frame. This number is enough for most applications.

| Item   | Function code | Byte count | Register value |
|--------|---------------|------------|----------------|
| Length | 1 byte        | 1 byte     | 2×N            |
| Value  | Same as request | 2×N        |                |

**Figure 6. Add time tick in Response PDU**

where \( N \) = quantity of input registers

3.2. Design of the foreground dialog display program

The foreground dialog program is used to show the acquiring data to operators or play back the log data to analysts. The program uses the curve and digital two ways to present them. The field pressure, flow, temperature, and water level are shown in the same form, and the display area is illustrated in Figure 7. Most areas of the dialog display the data change over the time in the form of curve to help operators master the overall fracturing process. Meanwhile, the data display bar on the left presents the...
name and real-time value of the corresponding channel in the form of text for quantitative control. The combination of the two display modes meets the needs of field work, and the operators in the test are Chinese. Hence, the software interface is designed in Chinese.

The foreground dialog display program connects to the background data acquisition service through a TCP/IP Port to obtain all data streams. Compared with the case of establishing connection for each sensor, this operation is greatly simplified. The sample data flow is shown in Figure 8. The two characters after the colon are transmission addresses for identifying data sources, and several sensor data streams are transmitted in turn. The foreground dialog display program queries the display location of the data from the display profile according to the transmission address to initialize display interface. The main contents of the profile include: display name, display location, display unit, and display adjustment factor. These parameter settings allow field operators to configure the display interface according to their habits and effectively improve the operation efficiency.

![Figure 8. Sample data flow from the background data acquisition service](image)

The data value and curve will not be timely updated if the display operation is blocked, which may lead to field operation errors. The following programming techniques are used in the system to avoid blocking the display data update, making the software robust.

1. Create independent task threads to complete the data operations that may be blocked, such as stream reading, file saving, and curve refreshing;
2. Clean up the threads that finish tasks in time and use WaitForSingleObject to suspend the waiting task threads and thus avoid invalid scheduling;
3. The work of each thread is decomposed reasonably and does not monopolize the system for a long time. For example: save files at regular intervals, manage the memory allocation...
reasonably, avoid frequent and inefficient access to system resources, and avoid affecting the efficiency of other task threads;
4. The necessary human–computer interaction is performed in the main thread and distributed to the task thread in the form of message. Avoid using modal dialogs in task thread running time, such as setting display parameters and saving parameters;
5. Use double buffering technology and temporary CDC to improve the fluency of the display curve. The BitBlt function is used to copy the main display area saved in memory, and cursors and prompt messages are drawn on temporary CDC to avoid frequently GDI operation.

Through the use of multi-threading, dual view buffer, and other technologies, the acquisition data of all sensors can be displayed on the computer screen in time to help field operators complete the fracturing operation. The foreground dialog display program can be installed in any computer connected to the server, making the field deployment easy.

4. Discussion and Conclusion
From December 6 to 10, 2020, the hydraulic fracturing rock pre-conditioning test at Tongkuangyu Copper Mine was completed successfully, and the data acquisition software worked stably. Approximately 62 MB continuous, accurate, effective data were obtained during the test, which provided strong support for subsequent data analysis.

The software system in this study can manage a certain number and types of sensors, with high stability, long-term continuity, flexible configuration, reliable operation, convenient function expansion, and good anti-interference performance at the same time. It can greatly reduce the proportion of data loss and allow multiple display units to simultaneously connect, which is suitable for the field investigation with a complex environment. Modifying it appropriately for rock deformation, pumping test, in-situ stress by anelastic strain recovery, and in-situ stress by hydraulic fracturing can simplify the software system deployment of related tests, improve the automation level of field operation, and has good practical value.

References
[1] Liu Yuming, Li Wen, Chen Xiaowei, Xia Changnian, Fan Wenlu, Gu Xiuha. Research on pre-conditioning techniques in hard rock block caving metal mines. China Mine Engineering, 2018, 47(3):59-63,74.
[2] Liu Yuming, Chen Xiaoyun, Peng Hua Li Guang, Chen Xiaowei, Li Wen, Xia Changnian, Sun Yao, Fan Lipeng, Zhang Hongchang. Field test of pre-conditioning by hydraulic in hard rock metal mine. China Mine Engineering, 2020, 49(2):1-4,14.
[3] Jiang Jing-Jie, Peng Hua, Li Zhen, Ma Xiu-min. Development and Application of The Log Auxiliary Software for The Marine Geological Survey. Journal of Geomechanics, 2005, 3:66-67,65.
[4] Standardization Administration of the P.R.C., 2008.GB/T 19582.1-2008.Modbus industrial automation network specification-Part 1: Modbus application protocol. Standards Press of China (China: Bei Jing).
[5] Standardization Administration of the P.R.C., 2008.GB/T 19582.2-2008.Modbus industrial automation network specification-Part 2: Modbus protocol implementation guide over serial link. Standards Press of China (China: Bei Jing).

Acknowledgments
This study was supported by the Geological Survey Project of China Geological Survey (DD20190546) and (DD20190290), and the software is used in these projects.