

Reducing the Unit Costs of Supplying Drilling Rigs with Spare Parts Based on Information Provided by the Neural Network Classifier

Fares Abu-Abed^{1,*}

¹Tver State Technical University, A. Nikitin Street, 22, 170026, Tver, Russia

Abstract. Minimizing costs when organizing the supply of drilling rigs with spare parts is an important task. To do this, it is necessary to develop evaluation criteria or rely on information obtained from the monitoring system in real time. The paper proposes a model for processing the output of a drilling rig, depending on the possible strategy for operating the equipment. The work is an integral part of previously published developments presented in the materials of articles in 2-nd and 3-rd International innovative mining symposiums (2017-2018).

1 Introduction

When developing any simulation model, it is necessary to make a choice between universal programming languages and specialized languages and software packages (shells) developed for the purposes of simulation modeling. This choice is not always obvious and depends on the objectives of the development of the model and the environment in which this model is to be used [12]. In this case, when building a simulation model, it is necessary to use previously developed means of recognizing Pre-crash and emergency situations on drilling rigs [11] written in C ++. Therefore, the use of ready-made shells is impractical, it is only possible to use high-level programming languages that are compatible with C ++. Of the modern cross-platform programming languages, C # is such a language.

The C # language is essentially a development of C ++ and borrows almost all its syntax. At the same time, it has a developed built-in library of standard classes, and its source code editor simplifies and significantly speeds up the writing of source texts of programs through an advanced substitution mechanism.

Therefore, as a result of the analysis of existing modeling languages and requirements for a simulation model of the drilling material supply system, the C # programming language has been adopted to develop this model. This language supports object-oriented programming (OOP) technology, according to which all the data and functions necessary to implement the modeling process are encapsulated in a number of classes.

* Corresponding author: aafares@mail.ru

2 Materials and Methods

The composition of the classes being developed usually depends on the subject area, in this case, on the subject matter of the simulation [1]. The logistics system consists of a set of interacting objects, the main of which are:

- Drilling rigs, the operation of which with minimal downtime is necessary to ensure. Each installation consists of many components required for its work, and also has certain geographic coordinates. Since the purpose of this work is not to create a specialized geographic information system, where all data are necessarily tied to geographic coordinates, the exact coordinates can be neglected. To build a model of the supply system, there is enough data on the removal of the drilling rig from the corresponding point of the spare parts KIT, from where it can receive components if necessary [7];
- Warehouses spare parts, both single and group, storing a given range of components needed to replace the failed, as a result of resource exhaustion, and as a result of accidents caused by emergency situations;
- The components themselves, which have a certain cost, a residual resource, as well as the cost of storage in a warehouse;
- Vehicles required for the delivery of components from a group spare parts to drilling rigs in the event of their failure and to supply the warehouses of the group spare parts themselves.

Developing a hierarchy of classes of the simulation model

The many classes that implement a program developed using object-oriented approach usually form a certain hierarchy, which corresponds to one of the fundamental principles of OOP, the principle of inheritance.

In C #, in addition to the inheritance implemented by the developer of the class hierarchy of the application, there is also built-in inheritance — all classes created by the developer are originally descendants of the abstract base class Object, from which some useful functions inherit [8].

Figure 1 shows the hierarchy of classes of the simulation model of the system of material supply of drilling rigs, developed in accordance with the above-stated results of an object-oriented analysis:

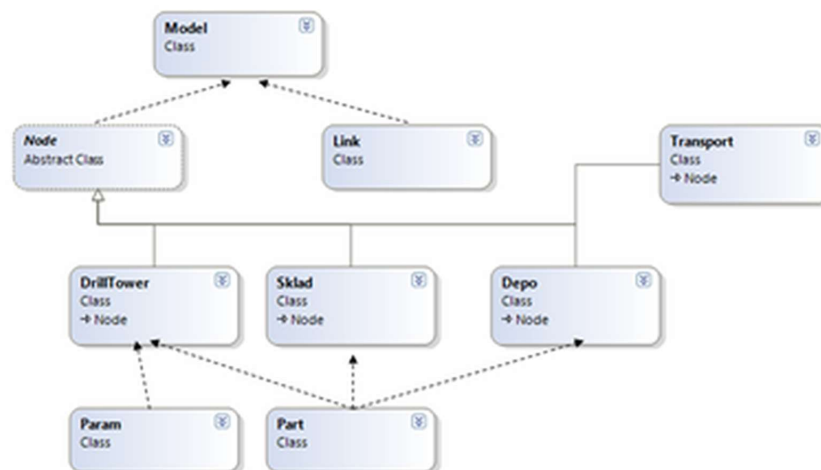


Fig. 1. Diagram of the hierarchy of classes of the simulation model.

The main class for the simulation model is the Model class, which implements container functions for the other classes. The whole modeling process is carried out using the functions of the Model class.

The Node class is the base class of the hierarchy of objects that make up the network model of the QS [3]. Since the model is developed by the user in

a special graphical editor and can function in the animation mode, each of its objects must have a graphic display.

The DrillTower class inherits the Node class and is designed to simulate the main production objects of the subject area - drilling rigs.

The Sklad class inherits the Node class and is designed to simulate a component storage that implements the functions of a group spare.

The Depo class inherits the Node class and is designed to simulate a top-level warehouse, from where warehouses for group spares are supplied with components.

The Transport class inherits the Node class and is designed to simulate transportation services for drilling rigs upon delivery of components to them from warehouses of a group spares, as well as to supply these warehouses from a top-level warehouse.

The Part class is designed to simulate the components required for the operation of drilling rigs and those included in the spare parts KIT.

The function ProcessDrillFault(DrillTower drill) performs the processing of messages about the failure of one or another component of a given drilling rig. The function algorithm is presented in Figure 2.

The TimerTick() function is central to the simulation process and performs the processing of the timer signals, which causes the movement of model time, the generation and processing of all applications in the model. The algorithm of this function is presented in Figure 3.

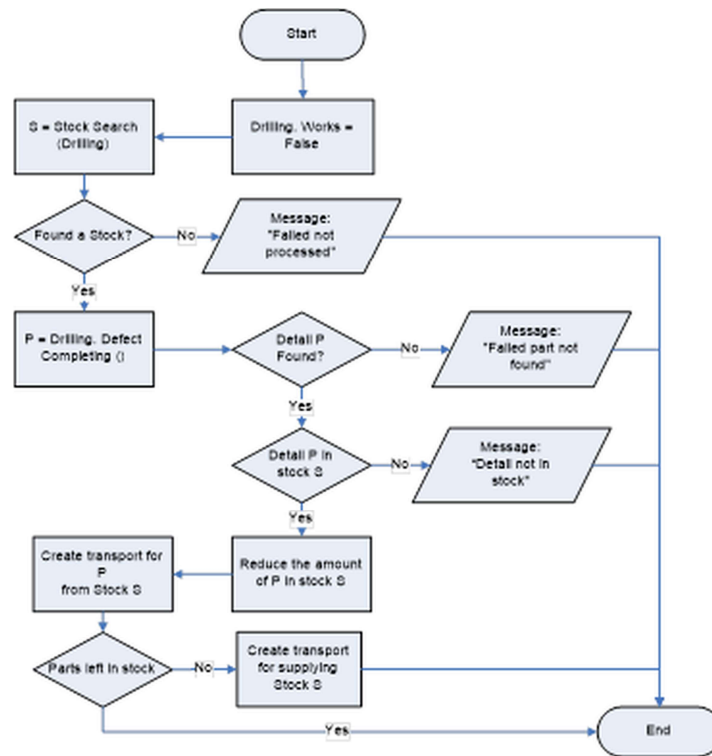


Fig. 2. Algorithm for processing the failure of a drilling rig.

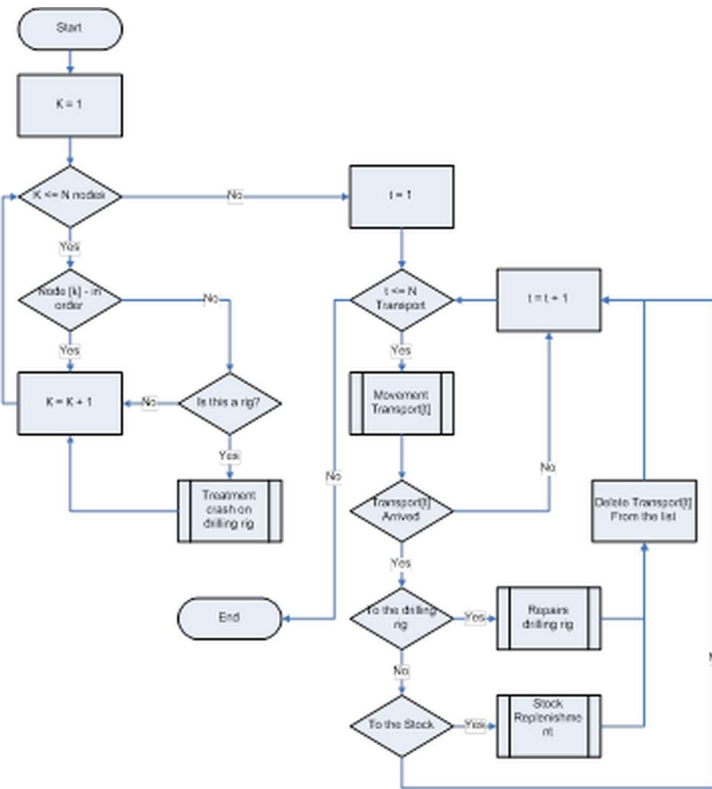


Fig. 3. The processing algorithm of the system timer.

3 Results and Discussion

Discussion of the simulation model of the supply system of spare parts

The study of the model of the supply of spare parts for a group of drilling rigs within one field was carried out for four strategies for operating the equipment [6]:

- Operation on a resource (development of a specified number of hours or meters of penetration);
- Operational status (detection of a critical state during the next routine inspection);
- Operation before failure due to failure or breakdown;
- Operational status (using neural network classifier) [4].

The results of simulation for each specified strategy was:

- Average waiting time for the start of repair work by the drilling rig;
- Total reduced economic effect from drilling operation.

To estimate the economic effect, the following expression was used [2]:

$$S = S_{Exp} + S_{down} + S_{Store} + S_{Trans} \quad (1)$$

The first two components correspond to the cost of operating the rig - the cost of operating in working condition and the cost of downtime when any component fails:

$$S_{\text{Exp}} = \sum_{t=0}^T s_t^{\text{Exp}}, \quad s_t^{\text{Exp}} = \begin{cases} a_{\text{Exp}}(t), & t \in T_{\text{Exp}} \\ 0, & t \in T_{\text{down}} \end{cases} \quad (2)$$

$$S_{\text{down}} = \sum_{t=0}^T s_t^{\text{down}}, \quad s_t^{\text{down}} = \begin{cases} a_{\text{down}}(t), & t \in T_{\text{down}} \\ 0, & t \in T_{\text{Exp}} \end{cases} \quad (3)$$

Here s_t^{Exp} - the cost of operating the drill at time t ;
 $a_{\text{Exp}}(t)$ - the cost of operating the rig per unit of time;
 s_t^{down} - the cost of a simple drilling at time t ;
 $a_{\text{down}}(t)$ - the cost of drilling downtime per unit of time.

The last two components correspond to the cost of storing components in spare parts stores and the cost of transporting them from warehouses to drilling sites, respectively. To calculate them, the following expressions are used:

$$S_{\text{Store}} = \sum_{i=1}^{N_{\text{stock}}} \sum_{j=1}^{N_k^i} s_{ij} t_{ij}, \quad S_{\text{Trans}} = \sum_{i=1}^{N_{\text{drill}}} \sum_{j=1}^{N_{\text{ck}}} \sum_{k=1}^{N_k^i} \bar{s}_{ij} d_{ij} n_{ik}, \quad (4)$$

Here N_{stock} - the number of warehouses spare parts;
 N_k^i - the number of components of the k -th type in the i -th stock;
 N_{drill} - the number of drilling rigs;
 s_{ij} - is the cost of storing the j -th component in the i -th stock;
 t_{ij} - is the storage time of the j -th component in the i -th stock;
 \bar{s}_{ij} - - the cost of delivery of components to the i -th drill from the j -th treasure;
 d_{ij} - - the distance between the i -th drilling and j -th warehouse;
 n_{ik} - - the number of components of the k -th type, delivered to the i -th drilling;

Since the processes in the simulation model of the QS are probabilistic in nature, it is necessary to estimate the number of model runs that are sufficient to obtain a stable result. The values of the criterion S were calculated during the modeling interval T , while the modeling process was carried out N times [5].

As a result of the analysis of the results obtained, it can be concluded that both the change in the value of S_{Medium} and the final result of the simulation at $N = 100$ and 200 practically coincide. Therefore, for further modeling, the value $N = 100$ model runs was taken [9, 10].

To study the effect of using the neural network classifier of the state of the drilling, we compared the simulation results for all four of the above strategies for operating the equipment.

The simulation results are shown in Tables 1 and 2. The economic effect presented was estimated as a quotient of dividing the effect obtained by modeling any strategy to the basic economic effect obtained by modeling the most ineffective strategy — operation before failure due to failure or failure.

Table 1. Average waiting time for the start of repair for various modes of equipment operation (h.).

N	Field	Strategy - 1	Strategy - 2	Strategy - 3	Strategy - 4
1.	Field - 1	18.46	20.43	22.34	15.25
2.	Field - 2	20.37	19.29	24.53	18.42
3.	Field - 3	23.18	25.87	30.21	20.34
4.	Field - 4	22.82	24.54	27.58	19.78

Table 2. The total reduced economic effect for various modes of equipment operation.

N	Field	Strategy - 1	Strategy - 2	Strategy - 3	Strategy - 4
1.	Field - 1	1.05	1.12	1	1.27
2.	Field - 2	1.02	1.08	1	1.21
3.	Field - 3	1.06	1.08	1	1.23
4.	Field - 4	1.08	1.11	1	1.25

4 Conclusion

From the obtained data, we can conclude that the use of a neural network classifier of pre-emergency situations reduces the average waiting time for the start of repair by 12% and increases economic efficiency by an average of 11% compared to the maximum value obtained using other strategies. The last estimate is approximate, because in assessing the economic effect, not all of its components were taken into account. Analysis of the simulation results allows us to conclude that the use of a neural network classifier of the state of the drilling rig in the organization of the supply system for drilling parts is advisable. For all the considered strategies for operating the equipment, the specific costs for the supply of drilling parts can be reduced by using information about the state of the drilling rig, issued by the neural network classifier. In this case, the minimum decrease is observed when using the strategy “before failure”, which is rarely used in real practice.

References

1. F. Abu-Abed., A. Khabarov, JFAS, **9:2S**, 1171-1181 (2017)
2. *Predictive Asset Analytics to Improve Mining Operations* (Schneider Electric Software, Berlin, 2015)
3. T. Maertens, J. Walraevens, H. Bruneel, Queuing systems, **55:1**, 27-39 (2007)
4. S.I. Gallant, *Neural Network Learning and Expert Systems* (MIT Press, Boston, 1993)
5. V. Gel'man, O. Marganiya, *Resource Curse and Post-Soviet Eurasia: Oil, Gas, and Modernization* (Logano Pub., Chicago, 2010)
6. M.D. Hammer, V.N. Syzrantsev, S.L. Golofast, *Simulators based on software and hardware platform for technical education* (Tyumen State Oil and Gas Univ., Tyumen, 2011)
7. F. Abu-Abed, A. Khabarov, E3S Web Conf., **41**, 01028 (2018)
8. F. Abu-Abed, N. Borisov, E3S Web Conf., **21**, 01019 (2017)
9. F.N. Abu-Abed, Oil Industry, **21**, 100-102 (2014)
10. F.N. Abu-Abed, Oil Industry, **11**, 107-111 (2015)

11. F. Abu-Abed, E3S Web Conf., **41**, 01025 (2018)
12. R. Lafore, *Object-oriented programming in C ++* (Peter, Sankt-Petersburg, 2003)