Solving the problem of service requests based on the algorithm for minimizing the maximum time offset

L V Rossikhina\textsuperscript{1,2} and E Yu Egorov\textsuperscript{1}

\textsuperscript{1}Voronezh institute of the Russian Federal Penitentiary Service, Irkutskaya St., 1a, Voronezh, 394072, Russia
\textsuperscript{2}Academy of Management of the Interior Ministry of Russia, Zoi and Aleksandra Kosmodemyanskih St., 8, Moscow, 125171, Russia

E-mail: rossihina_lv@mail.ru, turop_34@mail.ru

Abstract. The article proposes an approximate algorithm for forming the optimal sequences of service requests for repair of ISS elements with a minimum absolute error of the target function. The algorithm includes two steps. In the first step, new policy deadlines are set for the completion of application servicing. In the second step, a well-known algorithm for solving the problem with new Directive deadlines is used to determine the optimal sequence from which the desired solution to the original problem will differ by no more than an absolute error.

1. Introduction
Solving the problem of optimizing the service of requests for repair of ISS elements, it is of scientific and practical interest to consider the problem of schedule theory – the problem of minimizing the maximum time offset.

The problem consists in finding the optimal maintenance schedule requirements on one device with the smallest value of the maximum offset requirement equal to the difference between the date of completion of service requirements and policy completion of its service.

Work on solving this problem began in the 50s of the last century and is still underway [1,2]. In [3], it is proved that the problem of minimizing the maximum time offset is difficult.

To solve this class of problems, several approaches have been developed based on the use of reduced iteration methods (the method of branches and borders) [4,5,6], the method of dynamic programming [7,8,9], and the creation of heuristic, metaheuristic, and hybrid algorithms [10,11].

In the group of heuristic algorithms, approximations are singled out separately, for which error estimates of the resulting solution are found [12].

2. Problem statement
Let's assume that there are requests for repair of ISS elements $N = \{1, 2, \ldots, n\}$. Requests are serviced by a single employee, and it is prohibited to simultaneously service requests and interrupt service. For a repair request $j \in N$ have $t_j$ - the moment when the service request is received (the minimum
possible time when the service starts); \( p_j \geq 0 \) - duration of service; 
\( d_j \) - directive deadline for service completion.

The sequence of service requests is determined by the aggregate \( \pi = \{ S_j \mid j \in N \} \) highlights start of service.

The sequence of service requests \( \pi \) it is called valid if \( S_j(\pi) \geq \tau_j, \forall j \in N \).

Let \( c_j(\pi) \) - the moment when the request service was completed in the sequence \( \pi \); \( L_j(\pi) = c_j(\pi) - d_j, j \in N \) - temporary offset of the request \( j \) in the sequence. 

Maximum time offset of the request service \( j \) in the sequence \( \pi \):

\[
F(\pi) = \max_{j \in N} L_j(\pi^*)
\]  

Find the optimal sequence of service requests for repairs \( \pi^* \) with the lowest value of the maximum time offset \( \min_{\pi \in \Pi(N)} \max_{j \in N} (\pi) = \max_{\pi \in \Pi(N)} \) \( L_j(\pi^*) \), where \( L_j(\pi) = \max \{ c_j(\pi) - d_j \} \), \( \Pi(N) \) - a set of maintenance sequences for a set of repair requests \( N \). This task is NP - a difficult problem, for which an approximate algorithm is proposed, the idea of which is in changing the directive deadlines for servicing repair requests and obtaining such a sequence of service requests that provides the minimum value of the absolute error of the optimal value of the target function (1).

The approximate algorithm includes two steps:

Step 1. For applications for repair of many \( N \) find new decision-making time \( d_j, j \in N \).

Step 2. Solve the original problem under the following conditions:

\[
d_1 \leq \ldots \leq d_n, \quad \eta_1 \geq \ldots \geq \eta_n
\]  

Determine the desired sequence of service requests for repairs.

3. Description of the algorithm for changing policy deadlines

Let \( \pi^*, \pi \in \Pi(N) \) - optimal sequence to service requests for the repair of the many \( N \) when decision-making time \( d_j \) and \( d_j' \), respectively, \( \max_{j \in N} L_j(\pi^*) = \max_{\pi \in \Pi(N)} \max_{j \in N} \{ c_j(\pi) - d_j \} \). It is obvious that \( F(\pi^*) \geq F(\pi^*) \).

For tasks that minimize the maximum time offset with directive deadlines [13]: \( F(\pi^') \leq F(\pi^*) + \rho \), \( \rho = \max_{j \in N} \{ d_j - d_j' \} - \min_{j \in N} \{ d_j - d_j' \} \). Hence, the value of the target function (1) of the optimal sequence of service requests with directive deadlines \( d_j \), \( j \in N \) differs from the value of the target function (1) of the optimal sequence of service requests with directive deadlines \( d_j', j \in N \) no more than on \( \rho \).

Therefore, if you can choose a directive deadline for completing the service of applications \( d_j' \) so that the problem is solved by an effective algorithm, the resulting sequence of request servicing will be an approximate solution of the original problem with an estimate of the absolute error of the optimal value of the target function (1) not exceeding \( \rho \).

The problem of minimizing the absolute error \( \rho \) it can be represented as a mathematical programming problem (5), (6) provided that the repair requests of the set \( N \) numbered by the non-decreasing moments of the start of service requests.
\begin{equation}
r_1 \leq r_2 \leq \ldots \leq r_n
\end{equation}

with
\begin{equation}
r_j = r_{j+1} \Rightarrow d_j \geq d_{j+1}, \forall j = 1, n-1
\end{equation}

\begin{equation}
\max_{j \in N} \{d_j - d_j'\} - \min_{j \in N} \{d_j - d_j'\} \geq \frac{\min_{r=1}^{n!} n!}{r!(n-r)!}
\end{equation}

under the constraint
\begin{equation}
d_1' \geq d_2' \geq \ldots \geq d_n'
\end{equation}

Algorithm for solving the problem (5), (6).

Step 1. Renumber repair requests \( N \) according to (3), (4). Get a set \( N_1 = N \).

Step 2. If \( N_1 \neq \emptyset \), then find \( l_1 = \max\{j \in N_1 : d_j = \max d_j\} \), \( \overline{N_1} = \{j \in N_1 : j \leq l_1\} \), believe \( d_j = d_{l_1} \), \( j \in \overline{N}_1 \), \( N_2 = N_1 \setminus \overline{N}_1 \).

Behind \( n \) iterations will determine the directive terms of service requests \( \overline{d}_j \), \( j \in N \), minimizing absolute error \( p \).

The complexity of the algorithm is \( O(n \log n) \) operations based on from the fact that to renumber repair requests, you will need \( O(n \log n) \) operations [14].

To find new policy deadlines \( d_j \), \( j \in N \) you will need no more than \( O(n) \) operations.

4. Description of the second step of the approximate algorithm for solving the request service problem

Sequence of maintenance requests for repairs \( \pi \in \Pi(N,t) \) not earlier than the moment of time \( t \) we will consider it acceptable if \( F(\pi) \leq \gamma \), where \( \gamma \) - real number.

At the second step of the approximate request service algorithm the amount allowed for repairs is set relative to the specified value \( \gamma \) sequence of requests \( \pi_h \in \Pi(N,t) \), or it is established that such a sequence does not exist.

Step 1. To determine \( \gamma^0 = F(\pi_{h_0}) \), where \( \pi_{h_0} \in \Pi(N,t) \), such a sequence of maintenance requests for repairs, for which \( F(\pi_{h_0}) \) differs from the optimal value \( F(\pi^*) \) by no more than \( p_{\max}(N) = \max_{j \in N} p_j \) [15].

Step 2. To construct is valid with respect to \( \gamma^{k-1} \) sequence of service requests for repairs
\( \pi^{k-1} \in \Pi(N,t) \), \( 1 \leq k \leq n \), as follows: renumber repair requests so that the following conditions are met (2).

If \( \max\{n_i, i+1\} + p_i - d_i \geq \gamma \), \( \forall i = t, \ldots, P_k \), then \( \gamma = \max\{n_i, t\} \), \( N_1 = \{1\} \), \( P_1 = \max\{n_i, t\} + \sum_{j \in N} p_j - p_i \), to \( \pi^i = \pi^{k-1} \), otherwise \( \pi^i = (1) \).

Let them be known \( N_k, \ P_k, \ \pi^i_k \), \( 1 \leq k < n \), \( \forall i = t, \ldots, P_k \), believe \( N_{k+1} = N_k \cup \{k+1\} \), \( P_{k+1} = P_k + P_{k+1} \).

For each \( i = t, \ldots, P_{k+1} \) we get a sequence of maintenance requests for repairs.
\( \pi_i = \pi^0 \), if \( F(k+1, \pi_k, \max \{ \pi_{k+1} \} + p_{k+1}) \geq \gamma \),

otherwise \( \pi_i = (k+1, \pi_k, \max \{ \pi_{k+1} \} + p_{k+1}) \);

\( \pi^* = \pi^0 \), if \( F(\pi^0, k+1) > \gamma \),

otherwise \( \pi^* = (\pi^0, k+1) \).

Believe \( \pi'_{k+1} = \pi^0 \), if \( \pi_i = \emptyset \) otherwise \( \pi'_{k+1} = \arg \min \{ T(\pi) | \pi \in \Pi_i \} \), \( \Pi_i = \{ \pi \in \{ \pi_i, \pi^* \} : \pi^0 \pi^0 \} \).

The complexity of the algorithm implemented in step 2 is \( O(nP) \) operations [16].

Step 3. Build a sequence of service requests for repairs \( \pi_h^k \).

For \( \gamma^k = \gamma^{k-1} - 1 \), using the algorithm of step 2, we construct a valid relative \( \gamma^k \) sequence of requests for repairs \( \pi_h^k \in \Pi(N, t) \).

If \( \pi_h^k = \emptyset \), then the optimal sequence of servicing repair requests with the minimum maximum time offset is \( \pi^* = \pi_h^{k-1} \).

5. Conclusion

The paper considers an approximate algorithm for forming an optimal sequence of maintenance requests for repairs, based on changes in the directive terms of the original task and minimizing the absolute error of the optimal value of the target function.

The complexity of the presented algorithm is \( O(n^2P + np_{\max}P) \) operations.

The solution of a large number of examples (~1000) showed that in 20% of the examples, optimal sequences of maintenance requests for repairs were obtained, in other cases, the average value of the absolute error differed from the theoretically known value by no more than 15%.

References

[1] Lenstra J K, Rinnooy Kan A H G and Brucker P 1977 Complexity of machine scheduling problems An. of Dis. Math. vol 1 pp 343-62

[2] Lazarev A A, Sadykov R R and Sevastyanov S V 2006 Scheme of approximate solution of the problem Dis. ana. and oper. res. vol 13 pp 57-76

[3] Lenstra J K, Rinnooy Kan A H G and Brucker P 1980 Complexity of machine scheduling problems Euro. jour. of oper. res. vol 4 pp 270-5

[4] Sigal I H and Ivanov A P 2002 Introduction to applied discrete programming Fizmat. p 240

[5] Carlier J 1922 The one – machine sequencing problem Euro. j. of oper. res. vol 11 pp 42-7

[6] Lawler E L, Lenstra J K, Rinnooy Kan A H G and Shmoys P B 1993 Sequencing and scheduling: algorithms and complexity Handbo. in oper. res. and manag. sci. vol 4 (Logistics of Production and Inventory New York: North Holland) pp 452-522

[7] Gary M and Johnson D 1982 Computing machines and labor-intensive tasks Tra. with eng. p 416

[8] Papadimitriou H and K Steiglitz 1985 Combinatorial optimization Algorit. and comple. (Moscow: Mir) p 512

[9] Brucker P 2001 Scheduling Algorithms p 365

[10] Korbut A A, Segal I H and Finkelstein Y Yu 1988 Hybrid methods in discrete programming (Izv. USSR ACADEMY OF SCIENCES:Techn. Cybernet) p 65-77

[11] Jain V and Grossman I 2001 Algorithms for hybrid MILP/CLP models for a class of optimization problems Informs j. computing. vol 13 p 258 – 276

[12] Lazarev A A 2019 Theory of schedules. Methods and algorithms A A Lazarev (IPU Russian Academy of Sciences) pp 408

[13] Lazarev A A 1989 Effective algorithms for solving some problems in the theory of schedules.
for a single device with Directive terms of service requirements *Dis cand. physical and mathematical sciences* (Kazan) p 108

[14] Knut D 1973 The art of computer programming *Sorting and searching* vol 3 (Moscow: Mir) p 348

[15] Shulgina O N and Shcherbakova N K 2003 On an approximate algorithm for solving an NP-hard problem in schedule theory *Research. by approx. math. and inform.* (Kazan: Kazan state University UN-t Issue 24) p 146-55

[16] Shulgina O N 2001 Procedure for constructing a valid schedule for the task of minimizing the maximum time offset *Research. by approx. math and inform* (Kazan: Kazan Publishing house. math. Ob-VA Issue 23) p 150-8