OPTIMIZATION OF SYSTEM PARAMETERS OF 2: 3 GOOD SERIAL SYSTEM USING DEEP LEARNING

Shakuntla Singla[1], Shilpa Rani[2*], Umar Muhammad Modibbo[3], Irfan Ali[4]

•

[1,2] Department of Mathematics and Humanities, MMEC, Maharishi Markandeshwar
(Deemed to be University), Mullana, Ambala, India
Shakus25@gmail.com[1] gargshilpa46@gmail.com[2]

•

[3]Department of Operations Research,
Modibbo Adama University, P.M.B.2076, Yola Nigeria.
Email: umarmodibbo@mau.edu.ng

•

[4]Department of Statistics and Operations Research,
Aligarh Muslim University, Aligarh 202002, UP, India.
Email: irfii.st@amu.ac.in
*Corresponding author: gargshilpa46@gmail.com[2]

Abstract

In this paper, Optimization of System Parameters of 2:3 Good Serial System controlled with the help of a controlling unit Using Deep Learning Optimization with packing unit in series with priority in repair and single server which never fails is carried out. There are three units of different capacities working in parallel in which if three/two units are working then the system is working at full/reduced capacity. Working of these online parallel and offline units is managed by the controlling unit, which also manages the preventive maintenance of all type of units, together with a 24/7 repair facility is modeled for reliability performance measurements. Taking exponential failure and repair rates of units and facilities a steady state transition diagram (depicting transition rates and states) is drawn using Markov process. The system parameters are modelled using Regenerative Point graphical Technique (RPGT) and optimized using Deep learning methods such as Adam, SGD, RMS prop. The results of the optimization may be used to validate and challenge existing models and assumptions about the systems.

Keywords: Optimization, RPGT, Deep learning, Adam, SGD, RMS prop

I. Introduction

In this paper, Optimization of System Parameters of 2:3 Good Serial System Using Deep Learning Optimization with packing unit in series, with priority in repair managed by a controlling unit and single server who never fails is carried out. There are three main dissimilar units of different capacities performing different jobs working in parallel, together with the terminal unit (usually packing unit) controlled by a controller unit for allocation and preventive maintenance of all units in the system, together with single server who never fails is carried out. Out of three parallel units if two units are working then the system is working at reduced capacity and failed unit is maintained by the controller, with priority in repair to controlling unit performing preventive maintenance and single server is conducted using Markov process, RPGT and Deep Learning optimization methods. Preventive maintenance is feasible only in the initial state when controlling facility is almost free. The system measures can be optimized with proper utilization of maintenance activities of any system, in most of the studies under taken so far by most of the earlier authors, it is assumed that system once installed for operation will continue to do so, but practically it is not so, sometimes, it is necessary to manage the operation of working unit and maintenance facilities with the help of some controlling unit or external supporting system which may also fail (or non-available), on failure of one or more units or controlling unit, there is need of repair facility.

In this paper two out of three units in which if three/two units are working then the system is working at full/reduced capacity. Working of these online units is managed by controlling which also manages for the preventive maintenance of all type of units, together with a repair facility modeled for reliability performance measurements. Taking exponential failure and repair rates of units, a steady state transition diagram (depicting transition rates and states) is drawn using Markov process. Various directed paths, primary, secondary, tertiary circuits, base state, simple paths w. r. t. initial and base states enumerated. Various path probabilities transition probabilities mean sojourn times and expressions for four reliability measures are modeled using RPGT, optimization of system parameters using deep learning methods such as Adam, SGD, RMS prop is provided, which gives valuable insights into the factors that affect system performance by drawing corresponding tables and graphs followed by discussion.

II. Assumptions and Notations

- The repair procedure arises soon after a unit flops.
- Repaired unit is as if a new one.
- Failure/repair rates of units are exponential
- Server facility is 24x7 hours.
- B = controlling unit
- A_i, B, D/ a_i, b, d – Working state / failed state of individual units. (1≤i≤3)
- m_1/ h_1 – represent failure /repair rates of units A_i (1≤i≤3), m_2/ h_2 = those of controlling unit, m_3/ h_3 = those of packing unit D,
- α, β- transition rates of controlling unit performing preventive maintenance$\mu_i^{'}$ waiting time for repair facility to arrive
- $\mu_i^{'}$ waiting time for repair facility to arrive
- $(i \rightarrow$ j → k → i) cycle
- D = Packing unit
- $Repair\ Priority\ order\ is\ D > B >$ Ai
- ◯ : Full Capacity Working State
- ⬭ : Reduced Capacity Working State
- ☐ : Failed State

III. Transition Diagram Description

Taking the transition failure and repair rates, the system may be stable in the states S_i $(0 \leq i \leq 10)$.

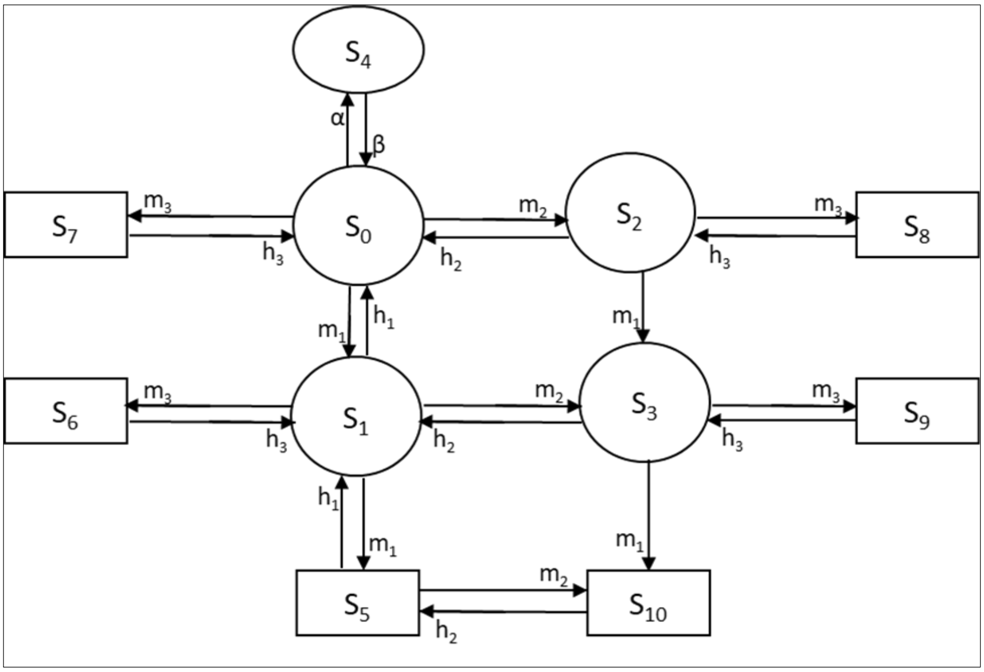

Figure 1: *Transition Diagram*

$S_0 = A_1A_2(A_3)$ BD, $S_1 = a_1A_2A_3$BD, $S_2 = A_1A_2(A_3)$ Bd, $S_3 = a_1A_2A_3$Bd,
S_4(under preventive maintenance) = $A_1A_2(A_3)$ BD,
$S_5 = a_1a_2A_3$BD, $S_6 = a_1A_2A_3$bD, $S_7 = A_1A_2A_3$bD, $S_8 = A_1A_2A_3$ bd, $S_9 = a_1A_2A_3$bd, $S_{10} = a_1a_2A_3$Bd

I. State Transition Probabilities

$q_{i\to j}(t)$
$q_{0\to 1}(t) = m_1 e^{-(m_1+m_2+m_3+\alpha)t}$
$q_{0\to 2}(t) = m_2 e^{-(m_1+m_2+m_3+\alpha)t}$
$q_{0\to 4}(t) = \alpha e^{-(m_1+m_2+m_3+\alpha)t}$
$q_{0\to 7}(t) = m_3 e^{-(m_1+m_2+m_3+\alpha)t}$
$q_{1\to 0}(t) = h_1 e^{-(m_1+m_2+m_3+h_1)t}$
$q_{1\to 3}(t) = m_2 e^{-(m_1+m_2+m_3+h_1)t}$
$q_{1\to 5}(t) = m_1 e^{-(m_1+m_2+m_3+h_1)t}$
$q_{1\to 6}(t) = m_3 e^{-(m_1+m_2+m_3+h_1)t}$
$q_{3\to 1}(t) = q_{2\to 0}(t) = h_2 e^{-(m_1+m_3+h_2)t}$
$q_{2\to 3}(t) = m_1 e^{-(m_1+m_3+h_2)t}$
$q_{2\to 8}(t) = q_{3\to 9}(t) = m_3 e^{-(m_1+m_3+h_2)t}$
$q_{3\to 10}(t) = m_1 e^{-(m_1+m_3+h_2)t}$
$q_{4\to 0}(t) = \beta e^{-\beta t}$
$q_{5\to 1}(t) = h_1 e^{-(m_2+h_1)t}$
$q_{5\to 10}(t) = m_2 e^{-(m_2+h_1)t}$
$q_{6\to 1} = q_{7\to 0} = q_{8\to 2} = h_3 e^{-h_3 t}$
$q_{10\to 3} = 0$
$q_{10\to 5} = h_2 e^{-h_2 t}$
$p_{i\to j} = q^*_{i\to j}(0)$
$p_{0\to 1} = m_1/(m_1+m_2+m_3+\alpha)$
$p_{0\to 2} = m_2/(m_1+m_2+m_3+\alpha)$
$p_{0\to 4} = \alpha/(m_1+m_2+m_3+\alpha)$
$p_{0\to 7} = m_3/(m_1+m_2+m_3+\alpha)$
$p_{1\to 0} = h_1/(m_1+m_2+m_3+h_1)$
$p_{1\to 3} = m_2/(m_1+m_2+m_3+h_1)$
$p_{1\to 5} = m_1/(m_1+m_2+m_3+h_1)$
$p_{1\to 6} = m_3/(m_1+m_2+m_3+h_1)$

$p_{2\to0}$= h2/ (m1+m3+h2)
$p_{3\to10}$ = $p_{2\to3}$= m1/ (m1+m3+h2)
$p_{3\to9}$ = $p_{2\to8}$= m3/ (m1+m3+h2)
$p_{3\to1}$= h2/ (m1+m3+h2)
$p_{2\to0}$= h2/ (m3+h2)
$p_{5\to10}$= m2/ (m2+h1)
$p_{4\to0}$ = $p_{6\to1}$= $p_{7\to0}$= $p_{8\to2}$=$p_{9\to3}$= $p_{10\to8}$= 1
$p_{0\to1}+p_{0\to2}+p_{0\to4}+p_{0\to7}=p_{1\to0}+p_{1\to3}+p_{1\to5}+p_{1\to6} = p_{2\to0}+p_{2\to3}+p_{2\to8}=p_{3\to1}+p_{3\to9}+p_{3\to10} = 1$

II. Mean Sojourn Times μ_i=R_i*(0)

$R_i(t)$
$R_0(t)= e^{-(m_1+m_2+m_3+\alpha)t}$
$R_1(t)= e^{-(m_1+m_2+m_3+h_1)t}$
$R_2(t)= e^{-(m_3+h_2)t}$
$R_3(t)= e^{-(m_1+m_3+h_2)t}$
$R_4(t)= e^{-\beta t}$
$R_5(t)= e^{-(m_2+h_1)t}$
$R_6(t)=R_7(t)=R_8(t)=R_9(t)= e^{-h_3 t}$
$R_{10}(t)= e^{-h_2 t}$
$\mu_0= 1/(m_1+m_2+m_3+\alpha)$
$\mu_1= 1/(m_1+m_2+m_3+h_1)$
$\mu_2= 1/(m_3+h_2)$
$\mu_3= 1/(m_1+m_3+h_2)$
$\mu_4= 1/\beta$
$\mu_5= 1/(m_2+h_1)$
$\mu_6= \mu_7= \mu_8= \mu_9= 1/h_3$
$\mu_{10}= 1/h_2$

III. Evaluation of Transition Path Probabilities:

Applying RPGT and '0' as the initial state of the system as under: The transition likelihood factors of altogether the reachable states after the initial state 'ξ' = '0' are:
$V_{0\to0} = 1$
$V_{0\to1,} = p_{0\to1}/\{(1-p_{1\to3}p_{3\to1})/(1-p_{3\to9}p_{9\to3})\}\{(1-p_{3\to10}p_{5\to10}p_{5\to1})/(1-p_{10\to5}\ p_{5\to10})\}(1-p_{1\to6}p_{6\to1})$
$(1-p_{1\to5}p_{5\to1})$
$V_{o\to2} = p_{2\to0} = /(1-p_{2\to8}p_{8\to2})$
$V_{0\to3}$ = ……., Continue
The transition likelihood factors of completely the reachable states since the base state 'ξ' = '1' are Probabilities since state '1' to dissimilar vertices stand given as
$V_{1\to0} = p_{1\to0}/\{(1-p_{0\to2}p_{2\to0})/(1-p_{2\to8}p_{8\to2})\}(1-p_{0\to4}p_{4\to0})(1-p_{0\to7}p_{7\to0})$
$V_{1\to1} = 1$
$V_{1\to2} = p_{0\to2}/\{(1-p_{0\to2}p_{2\to0})/(1-p_{2\to8}p_{8\to2})\}(1-p_{0\to4}p_{4\to0})(1-p_{0\to7}p_{7\to0})(1-p_{2\to8}p_{8\to2})$
$V_{1\to3}$ = ………, Continu

IV. Modeling system parameters

I.MTSF (F_0):

The states to which the arrangement can transit from original state '0', before joining down state are: 'i' = 0 to 4
$F_0 = (V_{0\to0}\ \mu_0+V_{0\to3}\ \mu_3+V_{0\to2}\ \mu_2+V_{0\to1}\ \mu_1+V_{0\to4}\ \mu_4)/[\{1-(0\to1\to0)-(0\to2\to0)-(0\to4\to0)\}]$
$= m_1^2 h_1 \alpha/m_1 + m_2 + m_3 + \alpha^2)(3h_1+2\alpha+m_2+m_3)$

II. Availability (Y_0):

The reformative states at which the organization is accessible are 'j' = 0 to 4 and the reformative states are 'i' = 0 to 8

$Y_0 = \left[\sum_j V_{\xi\to j}, f_j, \mu_j\right] \div \left[\sum_i V_{\xi\to i}, f_j, \mu_i^1\right]$

$= (V_{1\to1}\ \mu_1 + V_{1\to2}\ \mu_2 + V_{1\to3}\ \mu_3 + V_{1\to4}\ \mu_4)/D_1$

$Where\ D_1 = V_{1\to0}\ \mu_0 + V_{1\to1}\ \mu_1 + V_{1\to4}\ \mu_4 + V_{1\to3}\ \mu_3 + V_{1\to2}\ \mu_2 + V_{1\to5}\ \mu_5 + V_{1\to8}\ \mu_8 + V_{1\to7}\ \mu_7$
$+ V_{1\to6}\ \mu_6 + V_{1\to9}\ \mu_9 + V_{1\to10}\ \mu_{10}$

$Y_0 = m_1 m_2 / (m_1 + 3m_2 + \alpha + h_1)(3m_1 + m_2 + m_2^2 + \alpha h_1 + 3m_2)$

III. Busy Period of the Server (H_0) :

The states where the attendant is busy for doing some jobs are 'i' = 1 to 8, taking 'ξ' = '0', using RPGT busy period is given as

$H_0 = \left[\sum_j V_{\xi\to j}, n_j\right] \div \left[\sum_i V_{\xi\to i}, \mu_i^1\right]$

$Where\ D = V_{0\to0}\ \mu_0 + V_{0\to4}\ \mu_4 + V_{0\to2}\ \mu_2 + V_{0\to6}\ \mu_6 + V_{0\to1}\ \mu_1 + V_{0\to8}\ \mu_8 + V_{0\to3}\ \mu_3 + V_{0\to7}\ \mu_7 + V_{0\to5}\ \mu_5 + V_{0\to9}\ \mu_9$
$+ V_{0\to10}\ \mu_{10}$

$H_0 = [m_1/(m_1+m_2+m_3+\alpha)^2 + m_2(m_3+h_2) + 1/(m_1+m_2+m_3+\alpha) + (3m_1+m_2^2)(m_1+m_3+3h_1)$
$+ h_2/(m_1+m_3+h_1).\ h_1/(h_1+m_2) + 3h_1/(m_1^2+h_3) + m_2 h_2(h_2+3m_1+h_1+\alpha)$

IV. Expected Number of Examinations by the repair man T_0:

The reformative states where the repair person appointments afresh are j = 1, 2, 4, 7 the reformative states are i = 0 to 8, Taking 'ξ' = '0',

$T_0 = \left[\sum_j V_{\xi\to j}\right] \div \left[\sum_i V_{\xi\to i}, \mu_i^1\right]$

$= (V_{0\to4} + V_{0\to2} + V_{0\to1} + V_{0\to0})/D$

$T_0 = [(m_1+m_2)/(m_1+m_2+m_3+\alpha) + \alpha m_3(m_1+m_2+m_3+\alpha)(m_1+m_2) + m_1 m_3(m_1+m_2+m_3+\alpha)($
$m_1+m_2+m_3+h_1)(m_1+m_3+h_2) + m_1 h_2/(m_1+m_3+\alpha).\ (\alpha + h_1+m_2) + 3\alpha^2/(m_1+\alpha)(m_3+h_1)$

V. Optimization Using Deep Learning methods

Performing a optimization of a repairable using deep learning requires several steps in equation 1, 2, 3 and 4 to include for model to find different parameter [1, 2, 10]. Here is an example experiment that you could perform:

- Collection of data: Gather a dataset that contains information on the input parameters and the system's output. The input parameters could include factors such as the system's design, operating conditions, and maintenance schedule. The output could include metrics such as system availability, downtime, and failure rate in table 1 and table 2.
- Preprocess data: Clean and preprocess the dataset, splitting it into training, validation, and test sets.
- Train the model: Use a deep learning algorithm, such as a neural network, to model the connection among the input parameters and the output. Train the model using the training set and validate it using the set of values in table 1. You could use techniques such as early stopping and regularization to prevent over fitting.
- Appraise the model: After the model is proficient, appraise its performance by means of test set. Estimate metrics such as busy period.
- Perform sensitivity analysis: Using the trained model, vary the values of one parameter at a time while keeping the others constant. Record the effect on the system's output. Repeat this process for each input parameter, recording the impact of each parameter on the system's output.
- Interpret results: Analyze the consequences of the optimization examination to determine which input parameters need the most considerable influence on the system's output. You could use systems such as nose importance and fractional dependence plots to increase understandings into the mockup's behavior. Overall, performing a optimization of a repairable undertaken system using deep learning requires a combination of data collection, preprocessing, model training, and

analysis [3, 10]. It can be a powerful tool (Google Colab, Colab Notebook, Colab Python) for understanding the factors that contribute to the reliability of the system [4, 10]
.

VI. Dataset

Optimization is a way used to study how variations in the input parameters of an organization move the output. In the background of a repairable two out of three good system, optimization can help determine which parameters have the most significant impact on the system's reliability. To perform optimization using deep learning, you would need a dataset that contains information on the input parameters and the system's output [5, 6]. The input parameters could include factors such as the system's design, operating conditions, and maintenance schedule. The output could include metrics such as system availability, MTSF, and busy period

Table 1: *Table of parameter*

W (w1, w2, ---, wn)	$\lambda(\lambda 1, \lambda 2, \ldots, \lambda n)$	*S (s1, s2,---sn)*	*P*
(0-20,21-100)	*(0-30,31-100)*	*(0-100)*	*(0-80)*

Once you have a dataset, you could use a deep learning algorithm to model the relationship among the input parameters and the production. One approach could be to use a neural network, which can learn complex relationships between inputs and outputs. To perform optimization using a neural network, you could first train the network on the dataset, using a portion of the data for training and another portion for validation. Once the network trained, you could use it to make predictions on new input data, varying the values of one parameter at a time while keeping the others constant [7, 10]. By observing how changes in each parameter affect the system's output, you can determine which parameters have the most significant impact on the system's reliability to included dataset Table.1. Overall, optimization using deep learning can be an influential tool for understanding the issues that pay to the reliability of a repairable undertaken system. However, it requires a large and well-curated dataset, as well as expertise in deep learning techniques.

VII. Method

Optimization of a repairable system undertaken for analysis using deep learning typically involves the following steps:

- Data collection: Collect data on the input parameters and output metrics of the system. The input parameters could include factors such as the system's design, operating conditions, and maintenance schedule. The output metrics could include measures such as system availability, MTSF, and busy period in show table 2 included.
- Data preprocessing: Clean and preprocess the data, splitting it into training, validation, and test sets. Normalize the input variables to ensure that they are on the same scale.
- Model selection: Choose appropriate deep learning optimization techniques (Adam, SGD, RMS prop) for the sensitivity analysis. Some options contain feed forward neural systems, convolutional neural systems, and regular neural networks. Consider influences such as the size of the dataset, the difficulty of the input-output connection, and the computational capitals existing.
- Model training: Train the selected model on the training data. Use techniques such as stochastic gradient descent and back propagation to minimize the bust time. Monitor the performance of the model on the validation data, and adjust the hyper parameters as needed.
- Model evaluation: Evaluate the trained model on the test data. Calculate metrics such as mean absolute bust time and mean squared error to assess the model's performance of deep learning optimization in show table 1 and table 2.

- Optimization: Use the trained model to perform optimization on the input parameters. Vary the value of one input parameter at a time while holding the others constant. Record the effect on the output metric of interest. Repeat this process for each participation parameter to determine the of the output metric to changes in each parameter.
- Interpretation of results: Analyze the fallouts of the optimization examination to identify which input limits must the utmost impact on the output metric of interest. Use practices such as article importance and incomplete dependence plots to advance insights into the association amid the input limits and output metric [9, 10, 11].
- Several other methodologies were adopted by different scholars in the literature, some include optimization and estimations for the system parameters in reliability allocation and selective maintenance problems [12-15]. Others considered system availability under preventive maintenance [16] While others concentrated on the analysis of multiple hardware-software with Failure Interaction [17].

Overall, performing optimization of Repairable system under discussion using deep learning involves a combination of data collection, preprocessing, model selection, training, evaluation, and analysis.

Table 2: *Performance of model*

Model	-MTSF	Expected Number of Inspections by the repair man	-Busy Period	Availa bility
Adam	0.923	.9067	0.8012	0.9345
SGD	0.9123	0.9000	0.8123	0.9123
RMS prop	0.9012	0.8912	0.8103	0.9245

It can be a commanding tool for understanding the influences that underwrite to the reliability of the system.

VIII. Results and discussion

The results and discussion of a Optimization of undertaken repairable system parameters using deep learning will depend on the specific system and dataset analyzed. However, here are general insights that could be gained from such an analysis:

- Identification of critical system parameters: The optimization could reveal which input parameters require the greatest effect on the output metric of interest. For example, it could show that system availability is most optimization to the frequency of care or the quality of the components used in the organization.
- Understanding of the non-linear relationship amongst input strictures and output metrics: The deep learning model used in the analysis can capture non-linear relationships amongst input restrictions and output metrics, which could not detect using traditional statistical methods. The optimization can provide insights into the shape and magnitude of these relationships.
- Validation of existing models and assumptions: The results of the optimization applied to validate or challenge existing models and assumptions about the system. For example, the analysis could show that a certain parameter has a much greater impact on system performance than previously thought.
- Prediction of system behavior under different scenarios: The deep learning model applied to predict system performance under different setups, such as vagaries in operating conditions or

maintenance schedules. This can support decision-makers assess the impact of changed strategies and style informed verdicts.

Overall, Optimization of System Parameters of 2: 3 Good Serial System Using Deep Learning Methods can provide valuable insights into the factors that affect system performance, **(MTSF),** Expected Number of Inspections by the repair man, **Busy Period** and **Availability of the System are shown in figure 2, 3, 4 and 5**.

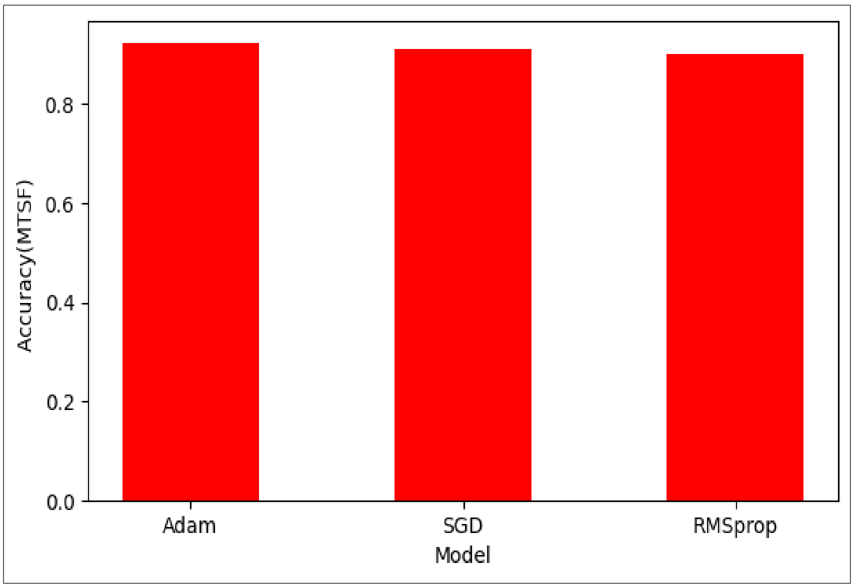

Figure 2: *comparing between models according to MTSF*

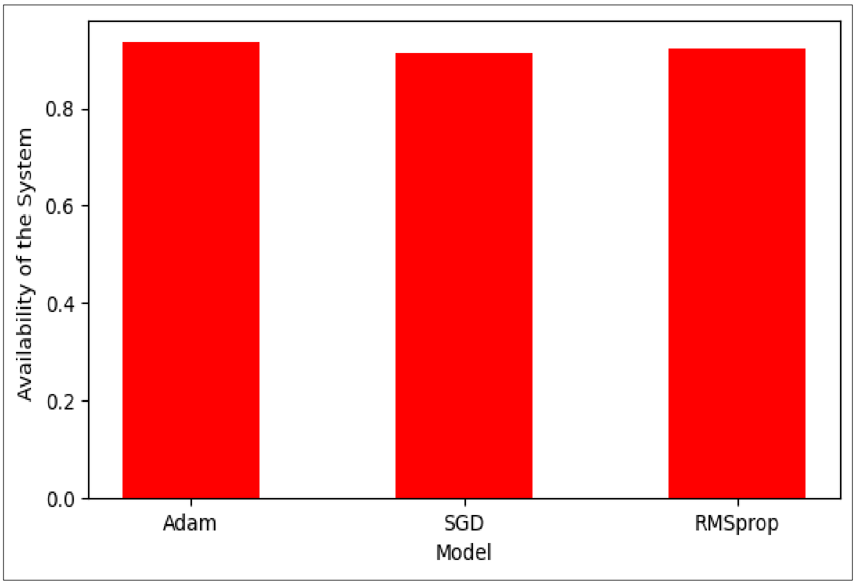

Figure 3: *comparing between models according to Availability*

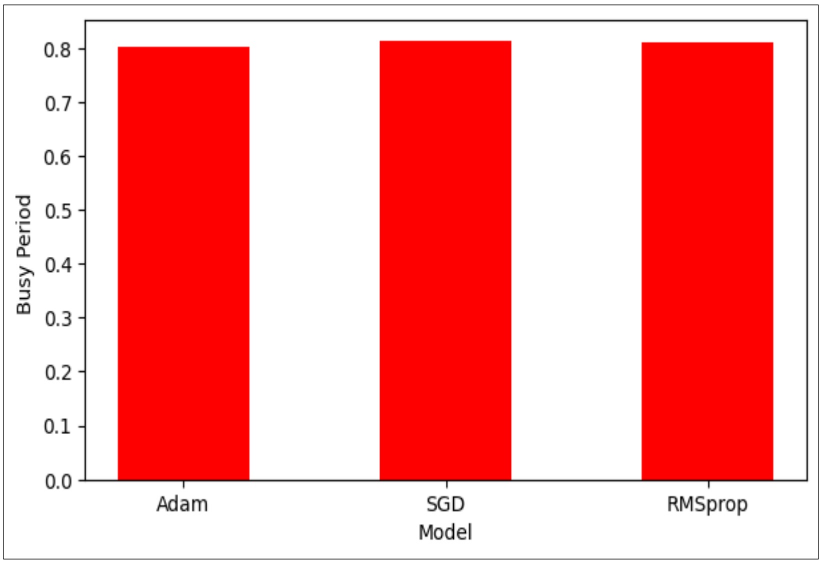

Figure 4*: comparing between models according to busy period*

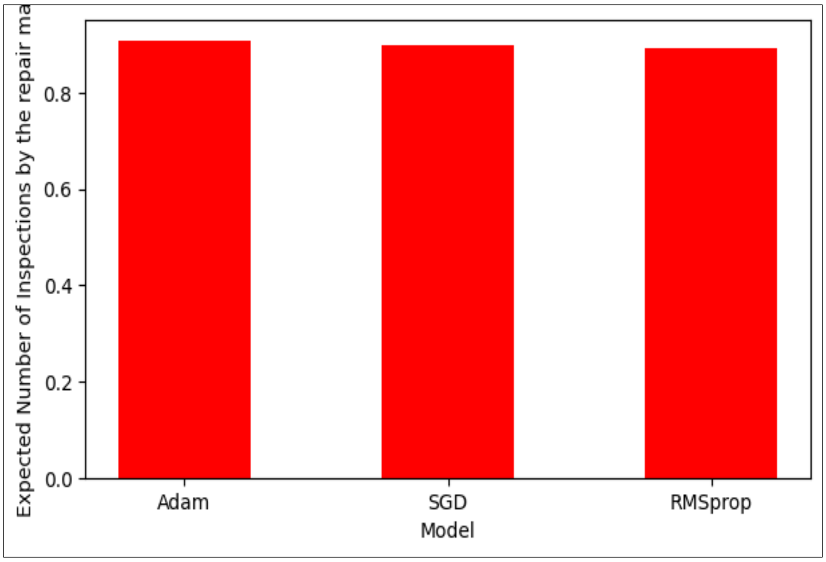

Figure 5: *comparing between models according to Expected Number of Inspections by the repair man*

MTSF between the different model is Adam is best performance among them. And busy time of Adam is better among them of model.

IX. Conclusion

In early research on the repair-replacement problem, the study of the model for a straightforward repairable system was the main focus of the repair-replacement models. However, it appears more logical to predict that the system's subsequent working times after repairs will get shorter and shorter while the system's subsequent repair durations after failure go longer and longer for a simple system that is deteriorating. In this paper, Optimization of System Parameters of 2: 3 Good Serial System Using Deep Learning Methods is carried out, along with external supporting systems for preventive maintenance and a single server that may also fail. The system consists of three similar units, out of which two units are working, at which point the system is operating at full capacity, and the third unit is kept in cold standby, which is switched in with the help of a perfect switch over system. The results of the optimization applied to validate or challenge existing models and assumptions about the system. For example, the analysis could show that a certain parameter has a much greater impact on system performance than previously thought. It can help optimize maintenance strategies, improve system design, and reduce downtime and maintenance costs.

References

[1] Fries, C., Wiendahl, H.,H. and Assadi, A.A. (2020). Design concept for the intralogistics material supply in matrix productions. Procedia CIRP 2020; 91: 33–8.

[2] Hsieh, Y.,C. and Chen, T.C. (2004). Reliability lower bounds for two-dimensional consecutive-k-out-of-n: F systems. Computers & Operations Research 31(8):1259–72

[3] John, Y. M., Sanusi, A., Yusuf, I., and Modibbo, U., M. (2023). Reliability Analysis of Multi-Hardware–Software System with Failure Interaction. *Journal of Computational and Cognitive Engineering*, *2*(1), 38-46.

[4] Kumar, A. (2020). Reliability And Sensitivity Analysis of Linear Consecutive 2-out-of- 4: F System. *European Journal of Molecular & Clinical Medicine*, *7*(07), 2020

[5] Kumar, A. (2022). Sensitivity Analysis Of Urea Fertilizer Plant. Journal of Reliability Theory and Applications Volume 17, RT&A No. 2 (68)

[6] Kumar, A. (2021) Performance Analysis of the Water Treatment Reverse Osmosis Plant. Journal of Reliability Theory and Applications Volume 16, RT&A No. 3 (63)

[7] Kim, H.K., Lee, S. and Yun, K.S. (2011). Capacitive tactile sensor array for touch screen application. Sensors and Actuators A: Physical 165:2–7.

[8] Koutras, M.V., Padadopoulos, G.K. and Papastavridis, S.G. (1994). Reliability of 2-Dimensional Consecutive-k-out-of-n: F Systems. IEEE Transactions on Reliability, 658–61.

[9] Kamal, M., Modibbo, U. M., AlArjani, A., and Ali, I. (2021). Neutrosophic fuzzy goal programming approach in selective maintenance allocation of system reliability. *Complex & intelligent systems*, *7*(2), 1045-1059.

[10] Khan, M. F., Modibbo, U. M., Ahmad, N., and Ali, I. (2022). Nonlinear optimization in bi-level selective e maintenance allocation problem. *Journal of King Saud University-Science*, *34*(4), 101933.

[11] Singla, S., Lal, A.K., and Bhatia,S.S (2011). Comparative study of the subsystems subjected to independent and simultaneous failure. *Eksploatacja I Niezawodnosc-Maintenance and Reliability*, *4*, 63-71.

[12] Raghav, Y. S., Varshney, R., Modibbo, U. M., Ahmadini, A. A. H., and Ali, I. (2022). Estimation and optimization for system availability under preventive maintenance. *IEEE Access*, *10*, 94337-94353.

[13] Singha, A. K., Pathak, N., Sharma, N., Gandhar, A., Urooj, S., Zubair, S., and Nagalaxmi, G. (2022). An Experimental Approach to Diagnose Covid-19 Using Optimized CNN. Intelligent Automation & Soft Computing, 34(2).

[14] Singla, S., and Kumari, S.(2022). Behavior and profit analysis of a thresher plant under steady stateB International Journal of System Assurance Engineering and Management 13, 166–171 .

[15] Singla, S., Lal, A. K., and Bhatia, S.S. (2021). Reliability analysis of poly tube industry using supplementary variable Technique *Applied Mathematics and Computation* 281, 3981–3992.

[16] Salvia, A. A. and Lasher, W.C. (1990). 2-dimensional consecutive-k-out-of-n: F models. IEEE Transactions on Reliability 39(3):382–5

[17] Singla, S., Modibbo, U. M., Mijinyawa, M., Malik, S., Verma, S., and Khurana, P. (2022). Mathematical Model for Analysing Availability of Threshing Combine Machine Under Reduced Capacity. *Yugoslav Journal of Operations Research*, *32*(4), 425-437.

[18] John, Y. M., Sanusi, A., Yusuf, I. and Modibbo, U. M. (2023). Reliability Analysis of Multi-Hardware–Software System with Failure Interaction. *Journal of Computational and Cognitive Engineering*, *2*(1), 38-46.

[18] John, Y. M., Sanusi, A., Yusuf, I., & Modibbo, U. M. (2023). Reliability Analysis of Multi-Hardware–Software System with Failure Interaction. *Journal of Computational and Cognitive Engineering*, *2*(1), 38-46.