Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/NA11.20260041
Published: June 9, 2026

FAST Reliability Estimation Using Neural Networks for k-out of-n Systems

Subbarao V. Majety
Assistant Professor, Director of Design Studio
Department of Mechanical and Civil Engineering
Purdue University Northwest
Hammond, IN, USA
majety@purdue.edu

Jayanth Chillarige
Department of Electrical and Computer Engineering
Graduate Student
Purdue University Northwest
Hammond, IN, USA
jchillar@purdue.edu

Abstract

Reliability estimation is essential for the design and operation of engineered systems. Calculating the reliability of large k-out of-n systems becomes computationally intensive as it involves summing numerous combinations of functioning and failed components. Direct enumeration methods can be used, but their runtime grows quickly, limiting their usefulness for repeated analysis. This study investigates artificial neural networks (ANN) as surrogate models to provide fast and accurate reliability estimates of a k-out of-n system. A large synthetic dataset of components' reliability is generated that is used for the training and evaluation of the neural network model. A multi-layer perceptron model is then trained to learn the nonlinear mapping from components' reliability to overall system reliability, with the output constrained to the [0,1] interval. This use of ANN is assessed using standard regression metrics and a deviance-based metric. Its computational results are compared against direct analytical computation across multiple k-out of-n settings. In addition, runtime is evaluated by comparing neural-network inference against direct analytical computation, showing that the proposed approach can deliver faster reliability estimates for larger k-out of-n systems. The ANN approach helps in estimating the reliability of k-out of-n systems with reasonable accuracy at much faster computational times.

Keywords
Reliability, k-out of-n system, Artificial Neural Networks and Multi-layer perceptron.

1. Introduction
System reliability is the probability that a system can perform the task without failure over its lifespan. A system's reliability depends on the reliability of the components that make up the system with a given architecture. Although the components are independent of each other, the system architecture would affect system reliability.

1.1 Architecture of a k-out of-n System

In a k-out of-n system, the system consists of n components, and the system functions if at least k components are operational. This structure provides fault tolerance by allowing n−k components to fail without causing system failure. Let N be the set of all components:

$$N = \{1, 2, 3, \dots, n\}$$

Let P_r be a subset of r components from the set N, such that:

$$P_r = \{i_1, i_2, i_3, \dots, i_r\}, \quad \text{with } i_1 < i_2 < \dots < i_r, \quad \text{and } i_j \in N$$

Let C be the collection of all such subsets P_r, where exactly r≥K components function. Assuming components are independent, and that each component i has reliability R_i, the system reliability R_{sys} is given by:

$$R_{sys} = \sum_{r \geq K}^{N} \left[\sum_{P_r \in C} \left(\prod_{i \in P_r} R_i \prod_{j \notin P_r} (1 - R_j) \right) \right] \quad \dots (1)$$

Where:

- $P_r \subset N$ is a subset of r functioning components.
- R_i is the reliability of component i.

Example: 2-out of-3 System (Figure 1)

For a system with 3 components where at least 2 need to function, the system's reliability is given as:

$$R_{sys} = R_1 \times R_2 \times R_3 + (1 - R_1) \times R_2 \times R_3 + R_1 \times (1 - R_2) \times R_3 + R_1 \times R_2 \times (1 - R_3) \quad \dots (2)$$

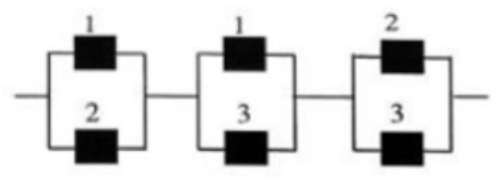

Figure 1. A 2-out of-3 system

1.1. Objective

The purpose of this study is to explore computation of reliability of a k-out of-n system efficiently with acceptable accuracy. As part of study, ANN is considered as a method.

2. Literature Review

Liu et al. (1993) conducted one of the earliest exploration studies on applying neural networks to reliability data analysis. This study shows the distribution estimation when the reliability data is compiled. Chen et al. (2023) employed deep neural networks to evaluate the reliability of manufacturing networks, illustrating the growing role of machine learning in system reliability analysis. But it is limited to manufacturing networks based on network topology. Taluja et al. (2020) demonstrated the use of ANNs as fast surrogates for the estimation of network reliability, trading off small prediction errors for large computational savings. This approach is more topology-based on a computer network showing the application of system reliability prediction. Huang et al. (2023) developed an artificial neural network model to predict the reliability of random capacitated-flow networks. Their approach replaces traditional minimal-path enumeration with a data-driven prediction method. The ANN learns how network topology and link capacities affect overall reliability. Results showed high prediction accuracy and a major reduction in computation time. This method provides a fast, general framework for assessing reliability in complex flow-based systems.

Bermejo et al. (2019) conducted a comprehensive review of artificial neural network (ANN) models applied to energy and reliability prediction across renewable energy systems, including solar photovoltaic (PV), hydraulic, and wind sources. The authors highlighted the potential of ANNs for system reliability estimation but noted key challenges such as limited data availability and lack of standardized reliability metrics. Agarwal et al. (2022) explored the application of Artificial Neural Networks (ANN) for reliability analysis in electrical distribution networks. Their study demonstrated that ANNs can accurately predict the reliability of complex systems with minimal computational overhead. The authors highlighted ANN's potential for enhancing decision-making in power system design and maintenance.

Zio (2013) introduced Monte Carlo simulation as an essential tool for system reliability and risk analysis, demonstrating that it is well suited for analyzing complex systems when accurate analytical methods are impossible or computationally expensive. Additionally, the study highlighted the necessity of more effective modeling techniques for large-scale reliability. Rai et al. (1987) proposed two recursive algorithms to calculate the reliability of k-out-of-n systems. They demonstrated that, in comparison to simple combinational methods, recursive approaches can lessen the computational burden of exact reliability evaluation.

3. Methodology

The reliability of a k-out of-n system is calculated using the equations from the General System section. To compute reliability on a computer, a Windows OS with an Intel i5 13th Gen processor is used, along with Python 3.13.2 running in a Jupyter Notebook.
The time required to compute the reliability of a k-out of-n system increases not only when n is larger but also when k gets smaller than n. Artificial Neural Networks (ANN) are used to estimate the system reliability. The ANNs are trained with a dataset prior to the deployment and are tested. The relation between the input and output is nonlinear as well as complex. This makes it a regressor problem. The ANN is a good choice for regression since the model can learn and adapt.

In the present study, separate ANN models were trained for each k-out-of-n configuration; therefore, the reported results demonstrate within-configuration generalization rather than the performance of a single universal model across varying values of k and n.

3.1 Machine Learning

A neural network is a parameterized machine learning model composed of multiple layers of interconnected artificial neurons. Supervised learning is a machine learning approach that is defined by its use of labeled data sets. Unsupervised learning is a class of machine learning methods that operate on unlabeled datasets, aiming to uncover underlying structures, patterns, or groupings within the data without external supervision. Reinforcement Learning is a branch of machine learning where an agent learns an optimal policy for interacting with an environment by taking actions in a state space to maximize cumulative expected reward. Supervised learning is employed for this study, since input and output is known for the neural network. System reliability is predicted in a range of [0,1], thus it is a regression problem.

4. Dataset and Target Variable

The dataset is a collection of data of all components' reliability. There are 30 variables in the dataset representing 30 components' reliability. More than 500,000 synthetic samples were generated for training and testing. Component reliabilities were sampled using a mixed strategy to cover both moderate and high-reliability regimes: a portion of the samples was drawn uniformly from [0.3, 1.0], while another portion was drawn from the narrower interval [0.9, 1.0] to better represent highly reliable engineering components. The target system reliability for each sample was computed analytically using Eq. (1).

4.1 MLP Architecture

The input tensor size is the number of components in the k-out of-n system. The input layer varies according to the model that is trained. The ANN model (Figure 2) that is employed has five hidden layers with a top-down approach. The first layer has 512 neurons and keeps decreasing so that it can converge into a single output. The second layer has 256 neurons, third layer and fourth 128, fifth has 64 connecting to an output layer. A non-linear function, typically the Rectified Linear Unit (ReLU), is applied after each hidden layer. The network concludes with a single dense unit using a sigmoid activation function. The output layer has a different activation function to restrict the predicted values between [0,1]. This layer takes the complex features learned by the hidden layers and scales them to produce the final, continuous regression output: the estimated overall system reliability.

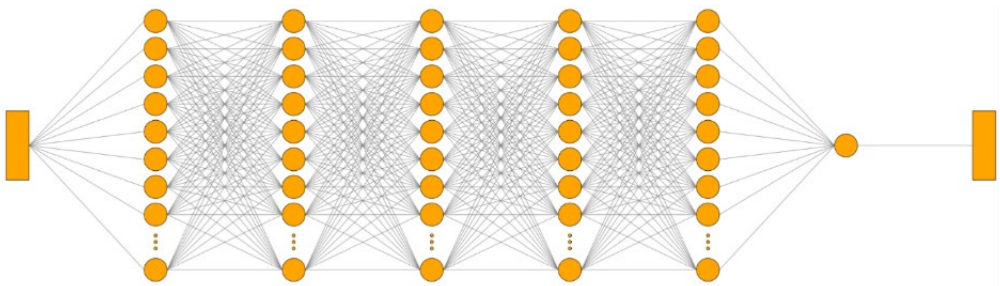

Figure 2. Neural Network Architecture

Figure 2 illustrates the MLP structure used in this work, with gradually decreasing hidden-layer width to learn the nonlinear mapping from component reliability to system reliability.

4.2 Evaluation Metrics

The ANN predicts the estimated system reliability $\hat{R}_{\text{sys},i}$ from component reliability, whereas exact reliability $R_{\text{sys},i}$ is computed using Eq. (1) for comparison. Model performance was evaluated on M test samples using error-based regression metrics mean squared error (MSE), root mean squared error (RMSE) and mean absolute error (MAE) together with the mean Poisson deviance.

$$\text{MSE} = \frac{1}{\text{M}}\sum_{\text{i}=1}^{M}\left(\text{R}_{\text{sys,i}} - \widehat{\text{R}_{\text{sys,i}}}\right)^2. \qquad \dots (3)$$

$$\text{RMSE} = \sqrt{\text{MSE}}. \qquad \dots (4)$$

$$\text{MAE} = \frac{1}{M}\sum_{\text{i}=1}^{M}\left|\text{R}_{\text{sys,i}} - \widehat{\text{R}_{\text{sys,i}}}\right|. \qquad \dots (5)$$

$$\bar{\text{D}}_{\text{Poisson}} = \frac{2}{M}\sum_{\text{i}=1}^{M}\left[\text{R}_{\text{sys,i}}\ln\left(\frac{\text{R}_{\text{sys,i}}}{\hat{\text{R}}_{\text{sys,i}}}\right) - \left(\text{R}_{\text{sys,i}} - \hat{\text{R}}_{\text{sys,i}}\right)\right], \qquad \dots (6)$$

By squaring the errors, MSE disproportionately penalizes larger errors. This makes it the ideal primary metric in reliability prediction, where large prediction errors (outliers) concerning a critical system's operational state are often deemed unacceptable. Root means squared error metric is highly valued because it returns the error to the original unit of the target variable, providing a more intuitive measure of the average error magnitude compared to MSE. MAE is more robust to occasional extreme count values, which may arise during abnormal operational periods or rare high-impact failure scenarios. In addition, model is assessed using the mean Poisson deviance.

4.3 Model Training

The model is trained using the Adam (Adaptive Moment Estimation) optimizer as it adaptively adjusts the learning rate for each parameter using estimates of the first and second moments of the gradient. To prevent overfitting and ensure good generalization, early stopping is applied during training. The validation loss is monitored; training is halted when there is no improvement and the model automatically restores the best-performing weights. A Reduce-on-Plateau learning rate scheduler is also used to improve convergence. This allows the optimizer to take smaller, more precise steps once progress slows, helping the model converge to a better local minimum.

5. Results and Discussion

5.1 Numerical Results

The reliability analysis of the proposed k-out of-n system model assumes that large systems composed of n essential components can be effectively represented by incorporating a limited number of redundant components. For the purposes of this study, redundancy is assumed to be introduced at a ratio of one-third relative to the essential components. Four models are trained for each k-out of-n system. The systems that are trained are shown in Table 1 along with the neural network metrics. The ANN achieved very small prediction errors, with MSE values below $10^{-6}$and RMSE values below 10^{-3}. For example, the 4-out-of-5 model produced an RMSE of $9.29 \times 10^{-4}$and MAE of 6.94×10^{-4}, while the 24-out-of-30 model achieved the best accuracy with RMSE 2.24×10^{-4} and MAE 1.74×10^{-4}. These results indicate that the network learned the nonlinear mapping from components' reliability to system reliability with high fidelity over the tested reliability ranges.

Table 1.ANN Model performance for K-out of-n systems

Model	MSE	RMSE	MAE	Poisson
4 out of 5	8.629942e-07	9.289748e-04	6.940378e-04	2.906479e-06
13 out of 20	6.352914e-07	7.970517e-04	7.906710e-04	6.356402e-07
16 out of 25	3.383222e-07	5.816547e-04	5.760938e-04	3.384575e-07
24 out of 30	5.023575e-08	2.241333e-04	1.739618e-04	5.028505e-08

Table 1 shows that the ANN maintained low prediction error across all tested configurations. The best performance was obtained for the 24-out-of-30 model, which achieved RMSE of 2.24×10^{-4} and MAE of 1.74×10^{-4}, indicating highly accurate approximation of analytical reliability values.

Computational efficiency was assessed by comparing exact reliability computation time with ANN prediction time. The performance of the models was tested on 100 samples of reliability values. For small systems such as 4-out-of-5, direct computation was effectively instantaneous at 0.0044s, whereas ANN prediction was slower, indicating that surrogate prediction is not advantageous for low-dimensional or easily enumerable systems. However, for larger systems the exact computation time increased sharply due to combinatorial growth. The 13-out-of-20 case required 201.71s for exact computation compared to 21.79s for ANN prediction, and the 16-out-of-25 case required 8290.75s compared to 21.96s for ANN prediction. The 24-out-of-30 case required 5203.25s for exact computation versus 50.16s for ANN prediction. These comparisons demonstrate that the surrogate provides substantial runtime savings for larger k-out of-n systems while maintaining high accuracy.

Table 2. Comparison of Time for Computation and Prediction

Model	Computation time (for 100 samples, in seconds)	Prediction Time (for 100 samples, in seconds)	Mean Deviation
4 out of 5	0.0044	35.2242	0.000853
13 out of 20	201.7108	21.7872	0.000666
16 out of 25	8290.7521	21.9646	0.000367
24 out of 30	5203.2515	50.1595	0.000183

Table 2 shows that the computational benefit of the ANN increases strongly with system size. For the 4-out-of-5 case, direct computation is faster because the exact evaluation is trivial. However, for larger systems the exact method becomes increasingly expensive, while ANN prediction remains comparatively stable, leading to large runtime savings.

The choice of the optimal regression model for reliability prediction is fundamentally dependent on the structure of the input data. By employing the Multi-Layer Perceptron (MLP), this study demonstrates that a trained multi-layer perceptron can serve as an accurate surrogate for computing k-out of-n system reliability from components' reliability. Using a large dataset with analytically generated targets, the ANN achieved consistently low prediction errors across multiple configurations, with RMSE values well below 10^{-3} and the best-performing case reaching RMSE

2.24×10^{-4}. The results confirm that the proposed network can approximate combinatorial reliability computation with high precision.

5.2 Graphical Results

Figure 3 shows that exact computation time increases sharply with system size, whereas ANN inference remains comparatively low, demonstrating the advantage of the surrogate in larger systems.

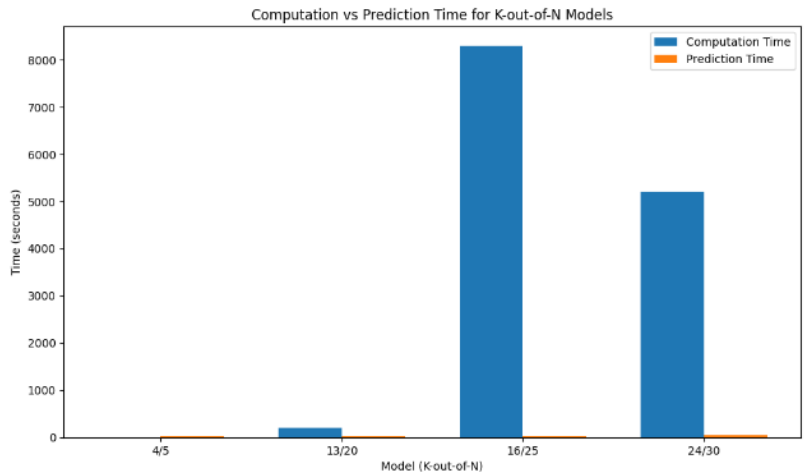

Figure 3. Comparison table for Estimated (Predicted) and Enumerated (Computed) time

5.3 Proposed Improvements

Prediction of the reliability of a system can be done using other machine learning techniques also. It is a regression problem and regressor techniques such as state vector regression can be used and compared with the ANN results.

Table 3. SVR Model performance for k-out of-n systems

SVR Model	MSE	RMSE	MAE	Poisson
4 out of 5	0.000020	0.004522	0.003789	0.000021
13 out of 20	2.881366e-07	5.367835e-04	4.261283e-04	2.882903e-07
16 out of 25	1.871158e-07	4.325688e-04	3.421295e-04	1.871453e-07
24 out of 30	5.886798e-07	7.672547e-04	5.829257e-04	5.892279e-07

For bigger k-out-of-n systems, SVR performance is nearly as accurate as ANN as seen from Table 3. With RMSE values less than 6×10^{-2}, SVR attains MSE = 2.88×10^{-2} (13/20) and 1.87×10^{-2} (16/25). Higher SVR error (MSE = 2×10^{-1}) is shown in the 4/5 scenario, which can happen in small-n domains when the output fluctuates dramatically with input reliability patterns. This behavior usually benefits from either hyperparameter adjustment specific to small-n distributions or denser sampling near border areas.

SVR achieved lower errors in the intermediate 13/20 and 16/25 cases, but overall, the MLP model demonstrated the best aggregate performance across all the tested k-out-of-n configurations. This shows that both nonlinear models work well, with MLP offering the best overall accuracy throughout the entire set of tests.

5.4 Validation (case studies)

5.4.1 Drivetrain of a Car

A combination of a k-out of-n system and series system is the drivetrain of an automobile. It can be understood from Figure 4. The computation time for getting the reliability of a simple system such as this is 0.002601 seconds. Figure 4 represents a relatively simple mixed architecture for which exact computation is already inexpensive, so ANN acceleration is unnecessary.

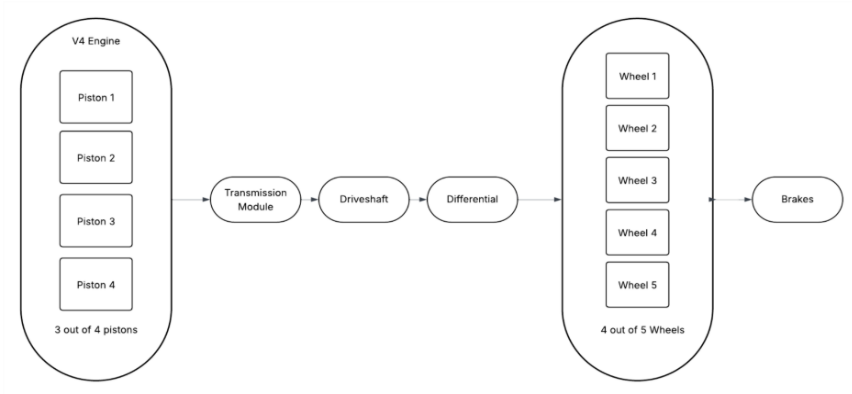

Figure 4. Block Diagram of a Drivetrain in a car

5.4.1.1 Power Distribution system

A series connection of k-out of-n systems represents the power distribution system. A similar representation can be found in the article by Taboada et al. (2008). In the system considered here, each stage is represented by k-out of-n system. The system is shown in Figure 5.

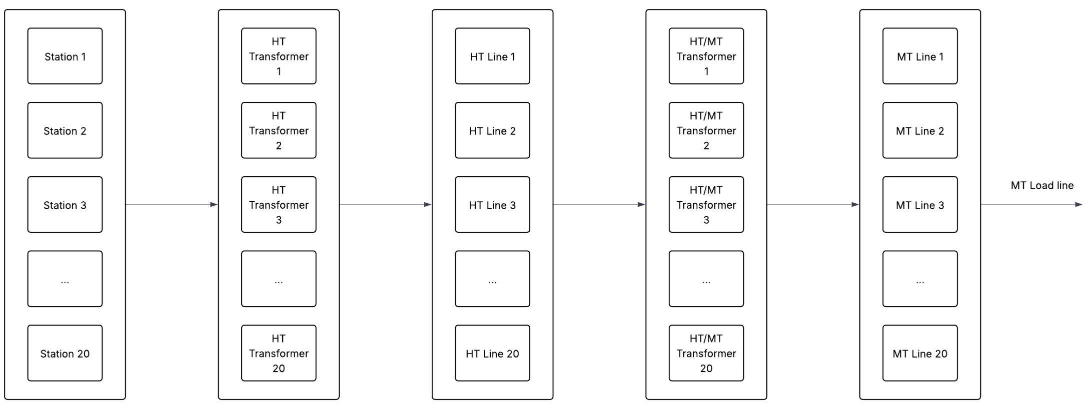

Figure 5. Power Distribution System

Each subsystem is a 13-out of-20 system having 7 redundant components (one-third) and subsystems are connected in series. Figure 5 illustrates a more complex series arrangement of k-out-of-n subsystems. For such systems, with more redundant components, the computation time to calculate the overall system reliability is high. For exact enumeration, the time taken for execution is 2.7842 seconds. To reduce this computation time, ANN is employed, and the estimation time is 0.3684 seconds with a deviation of 0.000136 from the true value. Hence, it can be concluded that neural networks provide accurate estimations for complex systems faster.

6. Conclusion

The primary advantage of the ANN approach is computational efficiency for large and highly redundant systems. While direct computation remains preferable for small architectures due to negligible runtime, the exact enumeration approach becomes impractical as n increases, and k decreases relative to n. However, as the number of components increases and the required number of functioning units becomes smaller relative to the total system size, exact enumeration becomes increasingly impractical. In such cases, the ANN surrogate substantially reduces computation time while maintaining close agreement with the analytical reliability values. In the larger configurations tested, the model achieved speedups of approximately 9×, 104×, and 377×, demonstrating its effectiveness for repeated reliability evaluation. Therefore, neural networks are especially useful in applications such as optimization, sensitivity analysis, redundancy allocation, and real-time decision support for complex systems.

At the same time, the method has some limitations. The ANN requires an initial training effort, and its predictive accuracy depends on the quality, range, and representativeness of the dataset used for training. In addition, the present work trains separate models for specific k-out-of-n configurations, so further investigation is needed to determine how well a unified model could generalize across different system sizes and architectures. The method proposed in this paper is an estimation which, in our opinion, works better than Monte Carlo simulation and also better than recursive methods computationally. Neural networks are a beneficial tool in applications requiring repeated reliability evaluations, such as optimization, sensitivity studies, redundancy allocation, and real-time decision support for complex systems. Overall, the results show that ANN-based surrogates provide a practical and accurate alternative to exact reliability computation for large k-out-of-n systems.

References

A. Taluja, A. K. Solanki, H. Kumar, and K. K. Singh, “Calculation and Evaluation of Network Reliability using ANN Approach,” in *Proc. Int. Conf. on Computational Intelligence and Data Science (ICCIDS) / Procedia Computer Science*, vol. 167, pp. 2153–2163, 2020.

C.-H. Huang, D.-H. Huang and Y.-K. Lin, "Network reliability prediction for random capacitated-flow networks via an artificial neural network," *Reliability Engineering & System Safety*, vol. 238, p. 109533, 2023, doi: 10.1016/j.ress.2023.109378.

E. Zio, “The Monte Carlo Simulation Method for System Reliability and Risk Analysis”, Springer, London, 2013, doi: 10.1007/978-1-4471-4588-2.

H. A. Taboada, J. F. Espiritu, & D. W. Coit, “Design allocation of multistate series-parallel systems for power systems planning: A multiple objective evolutionary approach”, *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 222(3), 381–391, 2008, https://doi.org/10.1243/1748006XJRR151

J. F. Bermejo, J. F. Gómez Fernández, F. Olivencia Polo, and A. Crespo Márquez, “A Review of the Use of Artificial Neural Network Models for Energy and Reliability Prediction: A Study of the Solar PV, Hydraulic and Wind Energy Sources”, *Applied Sciences*, vol. 9, no. 9, p. 1844, 2019.

M. C. Liu, W. Kuo, and T. Sastri, “An Exploratory Study of a Neural Network Approach for Reliability Data Analysis”, *Computers & Industrial Engineering*, vol. 25, nos. 1–4, pp. 185–188, 1993.

S. Rai, A. K. Sarje, E. V. Prasad, and A. Kumar, “Two Recursive Algorithms for Computing the Reliability of k-out-of-n Systems”, *IEEE Transactions on Reliability*, vol. 36, no. 2, pp. 261–265, 1987, doi: 10.1109/TR.1987.5222362.

U. Agarwal, N. Jain and M. Kumawat, "Applicability of ANN for Reliability Analysis of Distribution Network", *IEEE Delhi Section Conference (DELCON)*, New Delhi, India, 2022, pp. 1-7, doi: 10.1109/DELCON54057.2022.9753027

Y.-F. Chen, Y.-K. Lin, and C.-F. Huang, “Using Deep Neural Networks to Evaluate the System Reliability of Manufacturing Networks”, *Reliability Engineering & System Safety*, 2023.

Biographies

Subbarao V. Majety earned his PhD from University of Pittsburgh, Pittsburgh, PA. His research interests include reliability allocation, reliability optimization and reliability testing.

Jayanth Chillarige is a graduate student in Electrical and Computer Engineering Department at Purdue Northwest University. He earned his Bachelors in Electronics and Communications engineering from India.