# Node-Reliability: Monte Carlo, Laplace, and Stochastic Approximations and a Greedy Link-Augmentation Strategy

Xinhan Liu, Robert Kooij*, and Piet Van Mieghem, *Fellow, IEEE*

***Abstract*— The node-reliability polynomial $nRel_G(p)$ measures the probability that a connected network remains connected given that each node functions independently with probability $p$. Computing node-reliability polynomials $nRel_G(p)$ exactly is NP-hard. Here we propose efficient approximations. First, we develop an accurate Monte Carlo simulation, which is accelerated by incorporating a Laplace approximation that captures the polynomial's main behavior. We also introduce three degree-based stochastic approximations (Laplace, arithmetic, and geometric), which leverage the degree distribution to estimate $nRel_G(p)$ with low complexity. Beyond approximations, our framework addresses the reliability-based Global Robustness Improvement Problem ($k$-GRIP) by selecting exactly $k$ links to add to a given graph so as to maximize its node reliability. A Greedy Lowest-Degree Pairing Link Addition (Greedy-LD) Algorithm, is proposed which offers a computationally efficient and practically effective heuristic, particularly suitable for large-scale networks.**

***Index Terms*— network robustness, node failure, probabilistic graph, reliability polynomial**

## I. INTRODUCTION

RELIABILITY research in network science is concerned with the estimation of the probability that the residual network remains operational after the failure of some components [1]. In 1956, Moore and Shannon [2] proposed a probabilistic model for network reliability. Based on the types of components that can fail, network reliability can be classified into two categories:

**Network reliability w.r.t. link failures:** defined as the probability that the nodes of a connected graph $G$ remain connected if each link is independently operational with probability $p$, assuming the nodes of the graph are perfectly reliable [3]. This type of network reliability can be expressed as a so-called reliability polynomial:

$$\mathrm{Rel}_G\left(p\right) = \sum_{j=0}^{L} F_j(G)\left(1-p\right)^j p^{L-j}, \tag{1}$$

where $F_j(G)$ is the number of sets of $j$ links whose removal leaves $G$ connected, and $F_0(G) = 1$.

Xinhan Liu is with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands (email: X.Liu-22@tudelft.nl).

Corresponding author: Robert Kooij is with Unit ICT, Strategy & Policy, TNO, The Netherlands, and also with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands (email: R.E.Kooij@tudelft.nl).

Piet Van Mieghem is with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands (email: P.F.A.VanMieghem@tudelft.nl).

**Network reliability w.r.t. node failures:** defined as the probability that the operational nodes of a connected graph $G$ remain connected if each node is independently operational with probability $p$, assuming that the links of the graph are perfectly reliable [3]. This type of network reliability can be expressed as the node reliability polynomial:

$$\mathrm{nRel}_G(p) = \sum_{k=0}^{N} S_k(G)p^k(1-p)^{N-k}, \tag{2}$$

where $S_k(G)$ denotes the number of induced connected subgraphs with $k$ nodes.

Most studies on network reliability focus on link failures. This paper investigates node failures, assuming that each node remains operational independently with a uniform probability $p$. This assumption is justified in scenarios where nodes share similar physical characteristics, such as battery depletion in wireless sensor networks, random hardware failures in data centers, or telecommunication switching systems [4], [5]. The uniform probability provides a baseline scenario for evaluating network robustness, allowing straightforward comparisons across network topologies. Although the real-world node failure probabilities may vary, the uniform assumption simplifies initial robustness analysis and serves as a foundation for extensions to heterogeneous scenarios. Additionally, the assumption that links are perfectly reliable is appropriate when node robustness dominates reliability concerns, and link failures are comparatively negligible. Examples include wireless sensor networks, where link existence depends solely on node status, or data center networks, where node failures significantly outweigh link failures.

Compared to robustness metrics such as connectivity, algebraic connectivity, [6] natural connectivity, [7] network criticality, [8] path diversity, and spectral gap, [9] the node reliability polynomial provides distinct advantages. First, it explicitly incorporates the probabilistic nature of random node failures, intuitively interpreting network robustness as the probability of remaining interconnected under any node failure scenario. Traditional connectivity metrics, like algebraic connectivity and spectral gap, focus primarily on worst-case partitioning, lacking sensitivity to incremental probabilistic changes. Second, unlike natural connectivity and network criticality, node reliability quantitatively captures the cumulative effect of all possible failure scenarios, aiding precise evaluation and optimization of redundancy. Lastly, while path diversity assesses redundancy between specific node pairs, node reliability evaluates global connectivity involving all active nodes, thus offering a more comprehensive robustness assessment for practical communication networks and critical infrastructures.

The problems of computing the reliability polynomial $\text{Rel}_G(p)$ and node reliability polynomial $\text{nRel}_G(p)$ are NP-hard [10]–[12]. Closed-form analytic expressions for the node reliability polynomial only exist for some specific graph topologies [13]. Appendix D and Table I contains some examples. Various Monte Carlo methods give accurate estimations for the node reliability polynomial, but suffer from a high computational complexity [14]–[16]. For small- to medium-sized graphs, exact evaluation can also be performed with Binary Decision Diagram (BDD) techniques, such as the efficient vertex-reliability construction proposed by Kawahara *et al.* [17]. The reliability polynomial captures crucial information about a network's connectivity by encoding all possible cut sets—the sets of links or nodes whose removal would disconnect the network. The reliability polynomial thus serves as a comprehensive measure of a network's global robustness. Networks with higher values of the reliability polynomial, under the same operational probability $p$, tend to be more resilient to disconnection, allowing comparisons between different network topologies. In addition to a structural analysis, the reliability polynomial plays a key role in network design [1], [18], [19]. The reliability polynomial can identify critical nodes or links whose addition or removal significantly affects the overall reliability [19]. For instance, adding links can enhance reliability in communication networks by increasing redundancy, whereas removing specific links can effectively contain the spread of diseases [19]. The reliability polynomial finds practical applications in fields such as communication networks, infrastructure systems, and public health [19]–[21].

This paper first introduces a Laplace approximation for the node reliability polynomial in Section II. In Section III, we propose a Monte Carlo method for approximating node reliability polynomials. The proposed Monte Carlo method is inspired by a recent fast approach designed for network reliability polynomials [22]. Additionally, the Monte Carlo method is combined with the Laplace approximation to develop a new hybrid approach, referred to as the Laplace Monte Carlo method. Section IV introduces three degree-based stochastic approximations (Laplace, arithmetic and geometric) for the node-reliability polynomial. The relation between the reliability polynomial $\text{Rel}_G(p)$ and the node reliability polynomial $\text{nRel}_G(p)$ is also analyzed. Section V discusses a practical application of network reliability to link-augmentation ($k$-GRIP) optimisation. Notation and all variables are summarized in Appendix A. Additional analytical results for Erdős–Rényi and Random Geometric Graphs are provided in Appendix E.

## II. The Laplace approximation for the Node Reliability Polynomial $\text{nRel}_G(p)$

The number of combinations of $k$ different nodes out of $N$ nodes is the binomial coefficient $\binom{N}{k} = \frac{N!}{k!(N-k)!}$. We define the number $C_j(G)$ as the number of subsets of $j$ nodes whose removal disconnects the graph $G$. Every subset of $k$ nodes in $G$ must either be connected or disconnected, leading to the following relationship:

$$S_k(G) + C_{N-k}(G) = \binom{N}{k} \tag{3}$$

After substituting $S_k(G) = \binom{N}{k} - C_{N-k}(G)$ into Eq. (2), and applying Newton's binomial theorem, $(a+b)^N = \sum_{k=0}^{N}\binom{N}{k}a^{N-k}b^k$, we obtain:

$$\text{nRel}_G(p) = 1 - \sum_{k=1}^{N} C_{N-k}(G)p^k(1-p)^{N-k} \tag{4}$$

Hence, the (all-terminal) node reliability polynomial can be expressed both in the "$S$-form" and in the "$C$-form" as

$$\begin{aligned}\text{nRel}_G(p) &= \sum_{k=0}^{N} S_k(G)p^k(1-p)^{N-k} \\ &= 1 - \sum_{j=1}^{N} C_j(G)p^{N-j}(1-p)^j \end{aligned} \tag{5}$$

where $S_k(G)$ counts the number of induced connected subgraphs on $k$ nodes and $C_j(G)$ counts the number of vertex cut sets with $j$ nodes, which is the number of subsets of $j$ nodes whose removal disconnect the graph. We can express the node reliability polynomials in the "$S$-form" and "$C$-form" in *binomial forms* as well:

$$\begin{aligned}\text{nRel}_G(p) &= \sum_{k=0}^{N} \binom{N}{k} s_k(G)p^k(1-p)^{N-k} \\ &= 1 - \sum_{j=0}^{N} \binom{N}{j} c_j(G)p^{N-j}(1-p)^j \end{aligned} \tag{6}$$

where $s_k(G) = \frac{S_k(G)}{\binom{N}{k}}$ and $c_j(G) = \frac{C_j(G)}{\binom{N}{j}}$ are the fractions of induced connected subgraphs with $k$ nodes and vertex cut sets of graph $G$ with $j$ nodes amongst all possible node combinations of $k$ nodes and $j$ nodes from $N$ nodes. In other words, $s_k(G)$ equals the probability that the residual network remains connected after removing $N-k$ nodes and $c_j(G)$ is the probability that the residual part of graph $G$ is disconnected after removing $j$ nodes.

Although computing $S_k(G)$ and $C_j(G)$ is NP-hard, the node reliability polynomials $\text{nRel}_G(p)$ can still be approximated by estimating the probabilities $s_k(G)$ and $c_j(G)$. In this paper, we propose a Laplace Monte-Carlo approximation of $\text{nRel}_G(p)$ based on the "$C$-form" of the node reliability polynomial and a stochastic approximation of $\text{nRel}_G(p)$ based on the "$S$-form" of the node reliability polynomial.

The term $\binom{N}{k}p^k(1-p)^{N-k}$ represents the probability density function (pdf) of the binomial distribution. The Central Limit theorem states [23], [24] that the binomial distribution approaches the Gaussian distribution for large $N$. For large

$N$, the "$S$-" and "$C$-form" of the node reliability polynomials can then be approximated as:

$$\begin{aligned}\text{nRel}_G(p) &= \sum_{k=0}^{N} \binom{N}{k} s_k(G) p^k (1-p)^{N-k} \\ &\simeq \int_0^N s_k(G) \frac{\exp\left(-\frac{(Np-k)^2}{2Np(1-p)}\right)}{\sqrt{2\pi Np(1-p)}} dk \end{aligned} \tag{7}$$

Substituting $x = \frac{k}{N}$ transforms the integral in (7) into

$$\int_0^1 s_{Nx}(G) \frac{1}{\sqrt{2\pi}\sqrt{\frac{p(1-p)}{N}}} \exp\left(-\frac{(p-x)^2}{2\frac{p(1-p)}{N}}\right) dx \tag{8}$$

If we define $\tilde{\mu} = p$, and $\tilde{\sigma} = \sqrt{\frac{p(1-p)}{N}}$, then:

$$\text{nRel}_G(p) \simeq \frac{1}{\sqrt{2\pi}\tilde{\sigma}} \int_0^1 s_{Nx}(G) \exp\left(-\frac{(\tilde{\mu}-x)^2}{2\tilde{\sigma}^2}\right) dx \tag{9}$$

The Gaussian pdf $\frac{1}{\sqrt{2\pi}\tilde{\sigma}} \exp\left(-\frac{(\tilde{\mu}-x)^2}{2\tilde{\sigma}^2}\right)$ serves as an approximation of the Dirac delta function as [9, Sec. 7.1]:

$$\delta(x-\tilde{\mu}) = \lim_{\tilde{\sigma}\to 0} \frac{1}{\sqrt{2\pi}\tilde{\sigma}} \exp\left(-\frac{(\tilde{\mu}-x)^2}{2\tilde{\sigma}^2}\right) \tag{10}$$

Assuming that $\tilde{\sigma}$ tends to 0, the node reliability polynomial can be approximated by the following expression:

$$\begin{aligned}\text{nRel}_G(p) &\simeq \frac{1}{\sqrt{2\pi}\tilde{\sigma}} \int_0^1 s_{Nx}(G) \exp\left(-\frac{(\tilde{\mu}-x)^2}{2\tilde{\sigma}^2}\right) dx \\ &\simeq \int_0^1 s_{Nx}(G)\delta(x-\tilde{\mu})\, dx = s_{N\tilde{\mu}}(G) \end{aligned} \tag{11}$$

where $\tilde{\mu} = p$.

In summary, we call the approximation

$$\text{nRel}_G(p) \simeq \text{nRel}_G^{\text{Lap}}(p) = s_{Np}(G) \tag{12}$$

*the Laplace approximation* of the node reliability polynomial. Following a similar derivation, the Laplace approximation of the "C-form" of the node reliability polynomial is given by

$$\text{nRel}_G^{\text{Lap}}(p) = 1 - c_{N(1-p)}(G) \tag{13}$$

Based on the analysis in Appendix B, the total error of the Laplace approximation is $o\left(\frac{1}{\sqrt{N}} + \frac{s''_{Np}(G)}{N}\right)$, where $s''_{Np}(G)$ is the second derivative of $s_{Np}(G)$ about $p$. When $s_{Np}(G)$ varies more gently (i.e. $s''_{Np}(G)$ is of smaller order than $o(N)$), the second term decreases, and the approximation becomes more accurate. Meanwhile, for a fixed $p$, the error decreases as $N$ grows, further improving the accuracy. Hence, for sufficiently large $N$, the Laplace approximation in Eq. (12) yields increasingly accurate results.

## III. Monte Carlo Method for Approximating the Node Reliability Polynomial $\text{nRel}_G(p)$

The Monte Carlo method for estimating node reliability polynomials is based on a node deletion process, where at each time step a randomly selected node is removed. In a given graph, nodes are removed one by one until all nodes are eliminated. After each removal, the residual network is checked to determine whether it remains connected. By repeating the node deletion process $M$ times, the number of cases $R_j$ in which the removal of $j$ nodes disconnects the residual graph is obtained for each node $j \in [1, N]$. When the number of realizations $M$ is large, the probability that the removal of $j$ nodes disconnects the graph is approximately $\tilde{c}_j(G) = \frac{R_j}{M}$. Thus,

$$c_j(G) = \frac{C_j(G)}{\binom{N}{j}} \simeq \tilde{c}_j(G) \tag{14}$$

The $C-$form (4) of the node reliability polynomial $\text{nRel}_G(p) = 1 - \sum_{j=1}^{N} \binom{N}{j} c_j(G) p^{N-j}(1-p)^j$ can be approximated by:

$$\text{nRel}_{G,\text{MC}}(p) \simeq 1 - \sum_{j=1}^{N} \binom{N}{j} \tilde{c}_j(G) p^{N-j}(1-p)^j \tag{15}$$

where we denote the Monte Carlo approximation of $\text{nRel}_G(p)$ by $\text{nRel}_{G,\text{MC}}(p)$. The computational complexity of the Monte Carlo method is $\mathcal{O}(MN)$. Based on the analysis in Appendix C, the *mean squared error* (MSE) of the Monte Carlo approximation $\text{nRel}_{G,\text{MC}}(p)$ is $\text{MSE}[\text{nRel}_{G,\text{MC}}(p)] = o\left(\frac{1}{M}\right)$. Fig. 1 demonstrates the inverse power-law relationship between MSE and number of simulations $M$ by showing a linear decay trend in the log-log plot of $M$ versus MSE, consistent with a convergence rate close to $\mathcal{O}(1/M)$. The probabilistic error bound is $\Pr\left(|\text{nRel}_{G,\text{MC}}(p) - \text{nRel}_G(p)| > \epsilon\right) \leq 2\exp\left(-CM\epsilon^2\sqrt{Np(1-p)}\right)$, where $C$ is a positive constant. The error probability decays exponentially in $M$, $\epsilon^2$, and $\sqrt{Np(1-p)}$. For a network with a number of nodes $N$ and a prescribed error threshold $\epsilon$, the probability that the Monte Carlo estimate $\text{nRel}_{G,\text{MC}}(p)$ deviates from the true value $\text{nRel}_G(p)$ by more than $\epsilon$ decreases exponentially with a fixed number $M$ of simulations. For a fixed error threshold $\epsilon$, a larger network size $N$ leads to a smaller required number of simulations $M$ to achieve the same error probability.

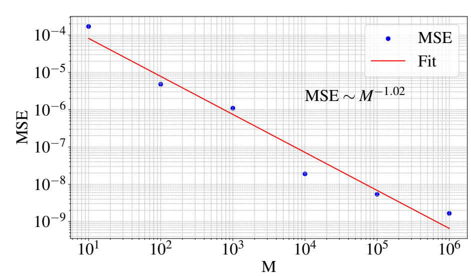



Fig. 1: MSE of the Monte Carlo approximation $\text{nRel}_{G,\text{MC}}(p)$ as a function of number of simulations $M$ for complete graph $K_9$ with a pendant node $K^*_{10}$. The slope of the red fit curve is $-1.02$

TABLE I: Network reliability polynomials for some simple networks

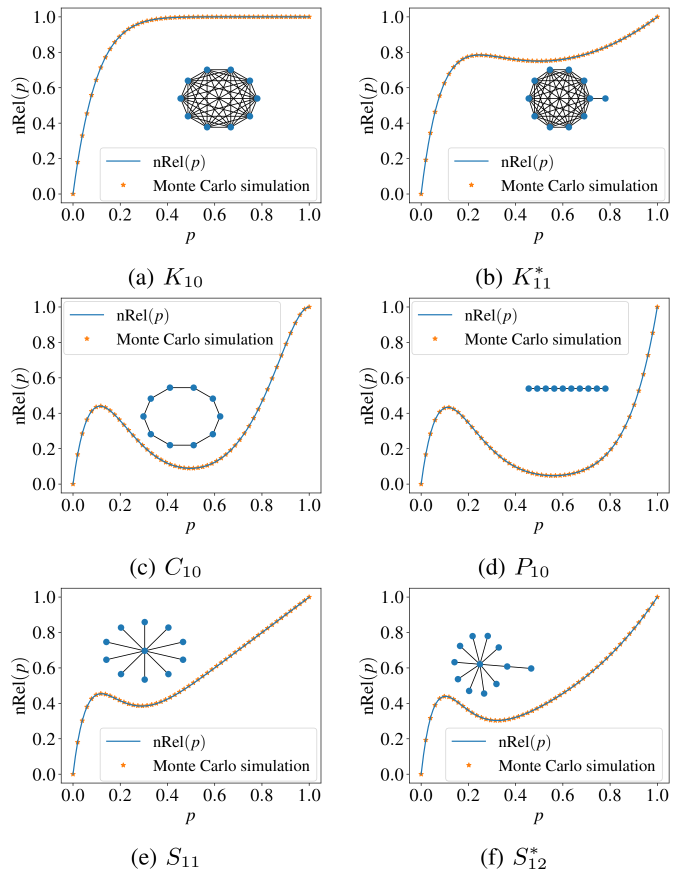
| Network (on $N$ nodes) | $\text{nRel}_G(p)$ |
|---|---|
| Complete graph $K_N$ | $\text{nRel}_G(p) = 1-(1-p)^N$ |
| Complete graph $K_{N-1}$ with a pendant node $K_N^*$ | $\text{nRel}_G(p) = p^2 + p(1-p)^{N-1} + (1-p)(1-(1-p)^{N-1})$ |
| Cycle graph $C_N$ | $\text{nRel}_G(p) = \frac{Np(p^N-(1-p)^N)}{2p-1} - (N-1)p^N \left(\text{nRel}_G(0.5) = \frac{N^2-N+1}{2^N}\right)$ |
| Path graphs $P_N$ | $\text{nRel}_G(p) = \frac{Np(1-p)^{N+1}-(N+1)p^2(1-p)^N+p^{N+2}}{(1-2p)^2} \left(\text{nRel}_G(0.5) = \frac{N(N+1)}{2^{N+1}}\right)$ |
| Star graph $S_N$ | $\text{nRel}_G(p) = p + (N-1)\,p\,(1-p)^{N-1}$ |
| Star graph $S_{N-1}$ with a pendant node $S_N^*$ | $\text{nRel}_G(p) = p^3 + p^2(1-p)^{N-2} + p(1-p)^{N-1} + (1-p)(p+(N-2)p(1-p)^{N-2})$ |

The Monte Carlo method is applied to several simple graphs, for which explicit closed-form analytical expressions of node reliability polynomials are known, see Table I and Appendix D. The graphs are the complete graph $K_N$, the complete graph $K_{N-1}$ with a pendant node, denoted by $K_N^*$ ($N$ nodes), the cycle graph $C_N$, the path graph $P_N$, the star graph $S_N$ ($N$ nodes) and the star graph $S_{N-1}$ with a pendant node, denoted by $S_N^*$ ($N$ nodes). The node reliability polynomials and the result of the Monte Carlo simulations are depicted in Fig.2 and Table III, which demonstrate that the Monte Carlo approximation is accurate for the considered graphs. In the remainder of this paper, the Monte Carlo method is used as a benchmark to evaluate other approximations of the node reliability polynomial.

The analysis in Section II has shown that, if the number of nodes $N$ in the graph $G$ is large, the node reliability polynomial can be approximated by Eq. (13) $\text{nRel}_G(p) \simeq 1 - c_{N(1-p)}$. The combination of the Monte Carlo method and the Laplace approximation is referred to as the Laplace Monte Carlo method, denoted as $\text{nRel}_{G,MC}^{\text{Lap}}(p) = 1 - c_{N(1-p)}$. The error of the Laplace Monte Carlo method combines the errors from the Laplace approximation and Monte Carlo sampling, and is therefore of order $o\left(\frac{1}{\sqrt{N}} + \frac{s''_{Np}(G)}{N} + \frac{1}{\sqrt{M}}\right)$. Fig.3 presents a comparison between the theoretical values of the node reliability polynomial $\text{nRel}_G(p)$ and the Laplace Monte Carlo approximation, represented as $1 - c_{N(1-p)}$. Fig. 3 indicates that the Laplace Monte Carlo approximation fits the theoretical values more closely as the size of the graph increases. Table III shows the MSE, mean absolute error (MAE), and maximum error of the Laplace Monte Carlo approximation $\text{nRel}_{G,MC}^{\text{Lap}}(p)$. Comparing Figs. 3a, 3b, and 3d, we observe that, with the number of simulations $M$ held fixed, both the MSE and MAE decline as the graph size $N$ increases. Conversely, a comparison between Figs. 3b and 3c shows that, for a fixed graph size $N$, increasing $M$ likewise reduces both MSE and MAE. The maximum error is contributed by the difference of $\text{nRel}_G(p)$ and $1 - c_{Np}$ at $p = \frac{1}{N}$, the larger the number of nodes $N$, the larger the second derivative value $s''_{Np}(G)$ at $p = \frac{1}{N}$, which contributes to a larger maximum error.

## IV. Stochastic Approximation for The Node Reliability Polynomial

Our previous work [25] has introduced a stochastic approximation for the reliability polynomial $\overline{\text{Rel}_G}(p)$. In this paper, we extend that approach towards a stochastic approximation for the node reliability polynomial $\overline{\text{nRel}_G}(p)$.

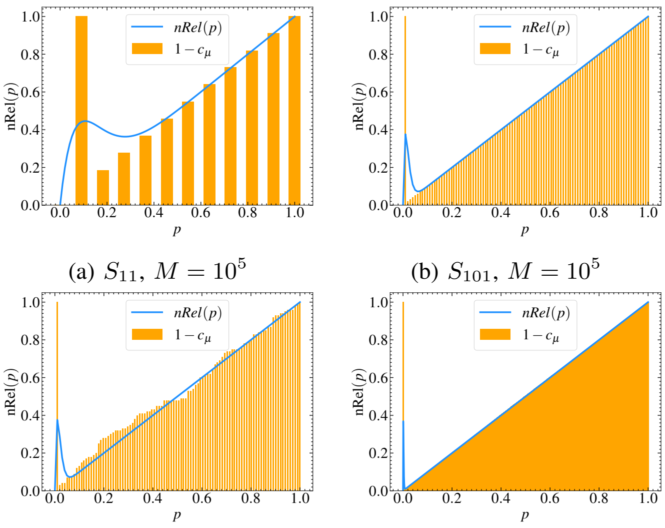



(a) $K_{10}$ (b) $K_{11}^*$ (c) $C_{10}$ (d) $P_{10}$ (e) $S_{11}$ (f) $S_{12}^*$

Fig. 2: Monte Carlo simulations ($M = 10000$) and exact values of node reliability polynomials for different graphs. The corresponding error metrics are summarized in Table III.

We denote the random residual graph with the failure of $N-k$ randomly selected nodes in $G$ as $\widehat{G}_k$. The probability that the residual graph $\widehat{G}_k$ is still connected equals $\Pr[\widehat{G}_k \text{ is connected}] = s_k(G)$.

For a given graph, the implication $\{G \text{ is connected}\} \Rightarrow \{D_{\min} \geq 1\}$, where the minimum degree is $D_{\min} = \min_{\text{all nodes} \in G} D$, is always true. However, the opposite implication does not generally hold, because it is possible for a network to be composed of several disconnected clusters where each node has a minimum degree greater than 1. Van der Hofstad [26] proves that for an Erdős–Rényi(ER) graph with large $N$ and certain link density $p_l$ which depends on $N$, the opposite implication $\{D_{\min} \geq 1\} \Rightarrow \{G \text{ is connected}\}$ holds almost for sure. For other network models with large $N$




(a) $S_{11}$, $M = 10^5$ (b) $S_{101}$, $M = 10^5$

(c) $S_{101}$, $M = 10^2$ (d) $S_{1001}$, $M = 10^5$

Fig. 3: The analytical expressions and Laplace Monte Carlo simulation results $1 - c_{N(1-p)}$ for the node reliability polynomial for star graphs $S_{N-1}$ under different $N$ and simulation sizes $M$. Error metrics are listed in Table III.

and high link density $p_l$, the equivalence $\{D_{\min} \geq 1\} \iff \{G \text{ is connected}\}$ also holds [27]–[29]. The main hypothesis of the stochastic approximation is that

$$\Pr[\widehat{G}_k \text{ is connected}] = \Pr[\widehat{D}_{\min} \geq 1] + o(1) \qquad (16)$$

where $\widehat{D}_{\min} = \min_{\text{all nodes} \in \widehat{G}_k} \widehat{D}$. Let $\Pr[D = k]$ be the probability that a randomly chosen node in the graph $G$ has degree $k$. The probability generating function (pgf) of the node degree $D$ in the graph $G$ is defined [23] as:

$$\varphi_D(z) = E[z^D] = \sum_{j=0}^{N-1} \Pr[D = j] z^j \qquad (17)$$

If the number of operational nodes is $k$, the probability that all neighbors of a node with degree $j$ fail independently of each other equals $(1 - \frac{k}{N})^j$. Consequently, the probability that a randomly chosen residual node $i$ in $\widehat{G}$ is isolated $\Pr[d_i = 0]$ equals $\varphi_D(1 - \frac{k}{N})$ [25]. The probability that the minimum degree $D_{\min}$ is larger than 0 is approximated by

$$\Pr[\widehat{D}_{\min} \geq 1] \simeq \prod_{i=1}^{k} (1 - \Pr[d_i = 0]) \simeq (1 - \Pr[d_i = 0])^k \qquad (18)$$

The probability that the residual graph $\widehat{G}_k$ remains connected is approximated as $\Pr[\widehat{G}_k \text{ is connected}] \simeq \Pr[\widehat{D}_{\min} \geq 1]$

$$\Pr[\widehat{G}_k \text{ is connected}] \simeq \left(1 - \varphi_D\left(1 - \frac{k}{N}\right)\right)^k \qquad (19)$$

The definition of the coefficients $S_k(G)$ in the "$S$-form" in (6) of the node reliability polynomial indicates that

$$S_k(G) = \binom{N}{k} s_k(G) \simeq \binom{N}{k} \left(1 - \varphi_D\left(1 - \frac{k}{N}\right)\right)^k \qquad (20)$$

Substituting (20) to the "$S$-form" of node reliability polynomial Eq. (2), leads to the approximation of the "$S-$form" node reliability polynomial $\text{nRel}_G(p)$ as

$$\text{nRel}_G(p) \simeq \sum_{n=0}^{N} \binom{N}{k} \left(1 - \varphi_D\left(1 - \frac{k}{N}\right)\right)^k p^k (1-p)^{N-k} \qquad (21)$$

When $N$ is large, the value of the node reliability polynomial $\text{nRel}_G(p)$ can be approximated by $s_{Np}(G)$ as:

$$\begin{aligned} \text{nRel}_G(p) \simeq s_{Np}(G) &= \Pr[\widehat{G}_{Np} \text{ is connected}] \\ &\simeq (1 - \varphi_D(1-p))^{Np} \end{aligned} \qquad (22)$$

Following the approach in our previous work [25], the stochastic approximation of the node reliability polynomial is denoted as:

$$\overline{\text{nRel}_G^{\text{Lap}}}(p) = (1 - \varphi_D(1-p))^{Np} \qquad (23)$$

The equivalence $\{G \text{ is connected}\} \iff \{D_{\min} \geq 1\}$, where the minimum degree is $D_{\min} = \min_{\text{all nodes} \in G}$ holds when the number of nodes $N$ and the link density $p_l$ is large, thus the accuracy of stochastic approximation $\overline{\text{nRel}_G^{\text{Lap}}}(p)$ increases with the increase of $N$ and $p_l$.

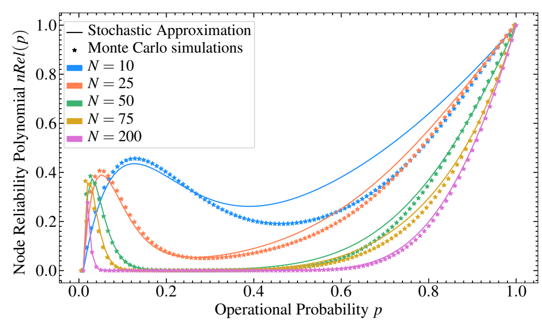



Fig. 4: Node reliability polynomial $\text{nRel}_G(p)$ obtained by the stochastic approximation and Monte Carlo simulations ($M = 10000$) for Erdős–Rényi graphs with different number of nodes $N$ and link probability $p_l = \frac{\log N}{N}$ depending on the number of nodes.

We first perform the simulations on Erdős–Rényi graphs with different number of nodes $N$ and link probability $p_l = \frac{\log N}{N}$. Fig. 4 demonstrates that the accuracy of the stochastic approximation increases with the size of the network. Derivations and analytical approximations of the node-reliability polynomial for Erdős–Rényi graphs are provided in Appendix E.

Fig. 5, 6, 7, and 8 depict the node reliability polynomial $\text{nRel}_G(p)$ obtained by stochastic approximation and Monte Carlo simulation for Barabási–Albert graphs, Erdős–Rényi graphs, 2D-lattice graphs and 3D-lattice graphs. We find that the accuracy of the stochastic approximation increases with link density $p_l$ and the number of nodes $N$.

Fig. 10 shows the node reliability polynomial $\text{nRel}_G(p)$ obtained by stochastic approximation and Monte Carlo simulation for some real-world networks. The corresponding parameters for these networks are provided. We find that the stochastic approximation demonstrates high accuracy in

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2025.3607004



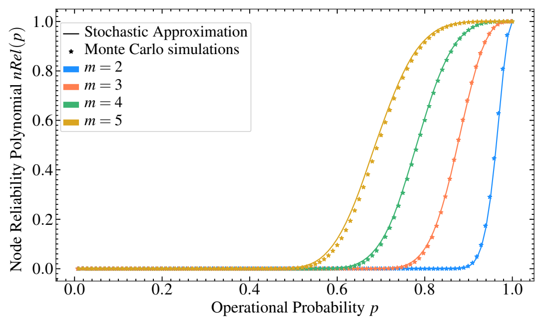



Fig. 5: Stochastic approximation and Monte Carlo simulations ($M = 10000$) of Barabási–Albert graphs with $N = 1000$ and different number of links added per step $m$

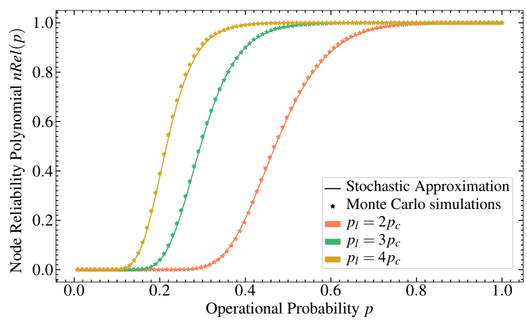



Fig. 6: Stochastic approximation and Monte Carlo simulations ($M = 10000$) for the Erdős–Rényi graphs with $N = 1000$ and critical link density $p_c \sim \frac{\log N}{N} = 0.0069$.

approximating the node reliability polynomial in real-world networks.

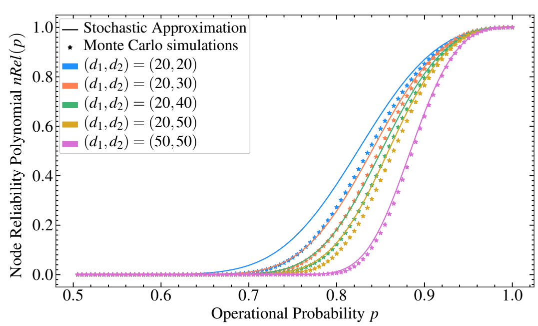



Fig. 7: Stochastic approximation and Monte Carlo simulations ($M = 10000$) for 2D-lattices with width $d_1$ and height $d_2$

We show in [25] that for graphs with a large number of nodes and high link density, the link reliability polynomial can be accurately approximated by:

$$\overline{\text{Rel}_G}(\tilde{p}) \simeq (1 - \varphi_D (1 - \tilde{p}))^N \tag{24}$$

where the variable $\tilde{p}$ denotes the probability of links being operational.

To evaluate the performance of our reliability approximation methods on structured communication topologies, we consider three representative network models: Torus [30], HyperX [31], and Fat-Tree [32]. These topologies are widely used in parallel

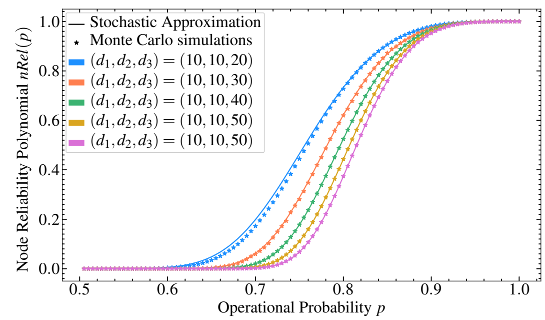



Fig. 8: The stochastic approximation and Monte Carlo simulations ($M = 10000$) for 3D-lattices with width $d_1$, length $d_2$ and height $d_3$

computing and data center networks due to their scalability. Fig. 9 shows the node reliability polynomial $\text{nRel}_G(p)$ obtained by stochastic approximation and Monte Carlo simulation. From the MSE and MAE value given in Table II, we find that the stochastic approximation shows a small error in computing the node reliability polynomial $\text{nRel}_G(p)$ of the Torus network and HyperX network, but perform poorly on the Fat-Tree network. For a given node functional probability $p$, the node reliability $\text{nRel}_G(p)$ of Fat-Tree network is always larger than the stochastic approximation $\overline{\text{nRel}_G^{\text{Lap}}}(p)$. The Fat-Tree topology provides high path redundancy, which makes it highly reliable against node failure.

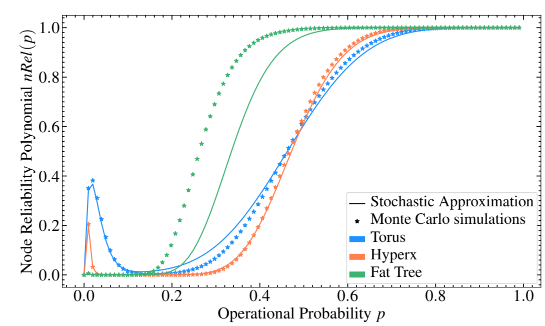
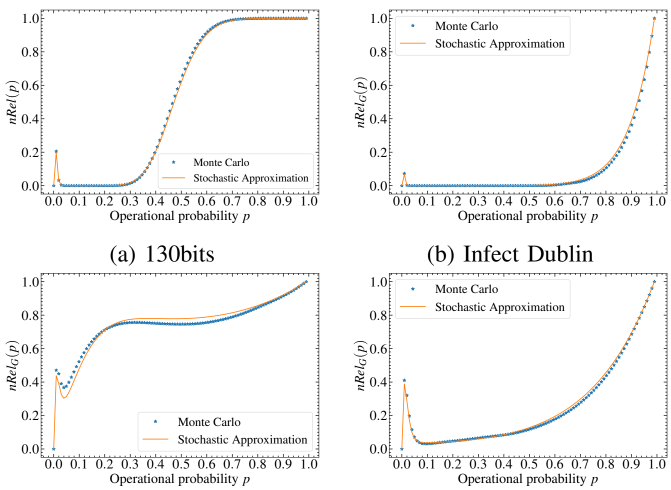



Fig. 9: Stochastic approximation and Monte Carlo simulations ($M = 10000$) for the node reliability of three structured network topologies: 3D Torus ($6 \times 6 \times 6$), HyperX (4D, 4 switches per dimension), and $k$-ary Fat-Tree ($k = 24$). The corresponding error metrics are reported in Table II.

TABLE II: Error metrics (MSE, MAE, $\Delta p_{99.999\%}$) for the structured topologies shown in Figure 9.

| Network | MSE | MAE | $\Delta p_{99.999\%}$ |
|---|---|---|---|
| Torus | $4.835 \times 10^{-4}$ | $1.646 \times 10^{-2}$ | 0.042 |
| HyperX | $8.798 \times 10^{-5}$ | $5.392 \times 10^{-3}$ | 0.027 |
| Fat-Tree | $1.953 \times 10^{-2}$ | $7.052 \times 10^{-2}$ | 0.102 |

We define $p_{99.999\%}$ as the minimum node activation probability $p$ such that the node reliability satisfies $nRel_G(p) \geq 0.99999$. This value is critical for meeting high availability targets, such as the "five nines" standard in telecom systems.




(a) 130bits (b) Infect Dublin

(c) Infect Hyper (d) Aves Songbird Social

Fig. 10: Monte Carlo simulations and stochastic approximations for various real-world graphs. The corresponding error metrics are reported in Table III.

Our results show that the stochastic approximation closely estimates $p_{99.999\%}$, offering a fast and practical tool for guiding network design, redundancy planning, and equipment selection. We define $\Delta p_{99.999\%}$ as the absolute difference between the estimated values of $p_{99.999\%}$ obtained by the stochastic approximation and by Monte Carlo simulation. For the 3D Torus and HyperX, $\Delta p_{99.999\%}$ is 0.042 and 0.027, respectively, indicating good accuracy. For the Fat-Tree, the difference increases to 0.102, suggesting reduced precision on Fat-Tree topologies.

TABLE III: Error metrics (MSE, MAE, Maximum Error) for Figures 2, 3, and 10.

| Subfigure | MSE | MAE | Max Error |
|---|---|---|---|
| 2a | 0 | 0 | 0 |
| 2b | $1.182 \times 10^{-6}$ | $9.907 \times 10^{-4}$ | $1.753 \times 10^{-3}$ |
| 2c | $2.373 \times 10^{-6}$ | $1.043 \times 10^{-3}$ | $3.288 \times 10^{-3}$ |
| 2d | $8.512 \times 10^{-7}$ | $8.262 \times 10^{-4}$ | $1.733 \times 10^{-3}$ |
| 2e | $2.485 \times 10^{-5}$ | $4.307 \times 10^{-3}$ | $7.247 \times 10^{-3}$ |
| 2f | $3.146 \times 10^{-6}$ | $1.676 \times 10^{-3}$ | $2.149 \times 10^{-3}$ |
| 3a | $3.228 \times 10^{-2}$ | $8.123 \times 10^{-2}$ | 0.559 |
| 3b | $4.787 \times 10^{-3}$ | $1.188 \times 10^{-2}$ | 0.624 |
| 3c | $5.817 \times 10^{-3}$ | $3.584 \times 10^{-2}$ | 0.624 |
| 3d | $5.031 \times 10^{-4}$ | $2.785 \times 10^{-3}$ | 0.631 |
| 10a | $1.328 \times 10^{-3}$ | $1.862 \times 10^{-2}$ | $2.215 \times 10^{-2}$ |
| 10b | $1.712 \times 10^{-4}$ | $7.576 \times 10^{-3}$ | $9.617 \times 10^{-3}$ |
| 10c | $7.827 \times 10^{-4}$ | $2.346 \times 10^{-2}$ | $3.151 \times 10^{-2}$ |
| 10d | $1.803 \times 10^{-4}$ | $1.062 \times 10^{-2}$ | $1.291 \times 10^{-2}$ |

### A. Arithmetic Stochastic Approximation $\overline{nRel_G^{arith}}(p)$ and Geometric Stochastic Approximation $\overline{nRel_G^{geom}}(p)$

The main hypothesis of stochastic approximation is $\Pr[\widehat{G}_k \text{ is connected}] = \Pr[\widehat{D}_{\min} \geq 1] + o(1)$. Sec. IV gives a way to estimate the probability $\Pr[\widehat{D}_{\min} \geq 1]$ that the minimum degree in the residual network $\widehat{G}$ is larger than 0. In addition to the Laplace stochastic approximation $\overline{\mathrm{nRel}_G^{\mathrm{Lap}}}(p)$, we provide two other stochastic approximations for the node reliability polynomials, the arithmetic stochastic approximation $\overline{\mathrm{nRel}_G^{\mathrm{arith}}}(p)$ and geometric stochastic approximation $\overline{\mathrm{nRel}_G^{\mathrm{geom}}}(p)$. Given a node $i$ with degree $d_i$, we define the event $\{\text{Node } i \text{ fails}\} \cup \{\text{Node } i \text{ is active and at least one neighbor of } i \text{ is active}\}$ as $X_i$. Its probability is denoted by $f_i = 1 - p(1-p)^{d_i}$. Under the independence assumption, the probability that every node is not isolated (i.e., each node experiences $X_i$) is $\Pr\left(\bigcap_{i=1}^{N} X_i\right) = \prod_{i=1}^{N} f_i$. However, when the events $\{X_i\}$ are not independent, we can correct the simple product by incorporating the dependencies via joint cumulants. In particular, according to the analysis in Appendix F, we have

$$\Pr\Big[\bigcap_{i=1}^{N} X_i\Big] = \prod_{i=1}^{N} f_i \times \left\{1 + \sum_{k=2}^{N} \sum_{1 \leq i_1 < \cdots < i_k \leq N} \frac{\kappa\big(X_{i_1}, X_{i_2}, \ldots, X_{i_k}\big)}{\prod_{m=1}^{k} f_{i_m}}\right\}. \tag{25}$$

where, $\kappa\big(X_{i_1}, X_{i_2}, \ldots, X_{i_k}\big)$ denotes the joint cumulant of the indicator variables of the events $X_{i_1}, X_{i_2}, \ldots, X_{i_k}$, which quantifies the pure $k$th-order dependence among these events.

When the events $\{X_i\}$ are mutually independent, all joint cumulants for $k \geq 2$ vanish, and the expression in (25) reduces to the product $\prod_{i=1}^{N} f_i$. The resulting geometric stochastic approximation for the node reliability polynomial is

$$\overline{\mathrm{nRel}_G^{\mathrm{geom}}}(p) = \prod_{i=1}^{N} \left(1 - p(1-p)^{d_i}\right) \tag{26}$$

When calculating the node reliability of large networks, concatenating the probabilities that each node is not isolated becomes computationally demanding. The arithmetic mean probability that a randomly selected node is not isolated node in $\widehat{G}$ is

$$P_{AM} = \frac{1}{N} \sum_{i=1}^{N} \Pr[X_i] \tag{27}$$

Substituting $\Pr[X_i] = f_i = 1 - p(1-p)^{d_i}$ into Eq. (27), we obtain

$$P_{AM} = \frac{1}{N} \sum_{i=1}^{N} f_i = \frac{1}{N} \sum_{i=1}^{N} \left(1 - p\,(1-p)^{d_i}\right) \tag{28}$$

which we rewrite as a sum over the nodal degrees by denoting $n_j$ as the number of nodes with degree $j$ and realizing that $\Pr[D = j] = \frac{n_j}{N}$,

$$\begin{aligned} P_{AM} &= \frac{1}{N} \sum_{j=0}^{N-1} n_{d=j} \left(1 - p\,(1-p)^j\right) \\ &= \sum_{j=0}^{N-1} \Pr[d = j] \left(1 - p\,(1-p)^j\right). \end{aligned} \tag{29}$$

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2025.3607004



Using the definition of pgf of the node degree (17), we obtain:

$$P_{AM} = \sum_{j=0}^{N-1} \Pr[d=j]\left(1-p\,(1-p)^{j}\right) = 1 - p\varphi_D(1-p) \tag{30}$$

By raising $P_{AM}$ to the $N$th power, the arithmetic stochastic approximation is derived as:

$$\overline{\text{nRel}_G^{\text{arith}}}(p) = \left(1 - p\varphi_D(1-p)\right)^N \tag{31}$$

The geometric mean of $1 - f_i$ over all nodes is $P_{GM} = \sqrt[N]{\prod_{i=1}^{N}(1-f_i)}$, and by the definition of $f_i$:

$$P_{GM} = \sqrt[N]{\prod_{i=1}^{N}\left(1-p\,(1-p)^{d_i}\right)}. \tag{32}$$

Since the arithmetic mean is always larger than or equal to the geometric mean, it holds that $\overline{\text{nRel}_G^{\text{arith}}}(p) \geq \overline{\text{nRel}_G^{\text{geom}}}(p)$.

The computational complexity of the arithmetic stochastic approximation, $\overline{\text{nRel}_G^{\text{arith}}}(p) = \exp\left(N\log\left(1-p\varphi_D(1-p)\right)\right)$, depends on the computation of $\varphi_D(1-p)$, which has a complexity of $o\left(\|D\|\right)$, where $\|D\|$ represents the number of distinct degrees in the graph. In contrast, the geometric stochastic approximation, $\overline{\text{nRel}_G^{\text{geom}}}(p)$, involves calculating the term $\left(1-p\,(1-p)^{d_i}\right)$ for each of the $N$ nodes, resulting in a computational complexity of $o(N)$. Therefore, in networks where $\|D\| \ll N$, the arithmetic stochastic approximation requires significantly fewer computational resources, leading to a notable reduction in computation time and increased efficiency. As the network size increases, this advantage becomes even more significant, making the arithmetic stochastic approximation a practical choice for analyzing large-scale networks with a limited number of distinct degrees. For cases requiring higher precision, the geometric stochastic approximation $\overline{\text{nRel}_G^{\text{geom}}}(p)$ can be used, while the arithmetic stochastic approximation $\overline{\text{nRel}_G^{\text{arith}}}(p)$ is preferred for its computational efficiency.

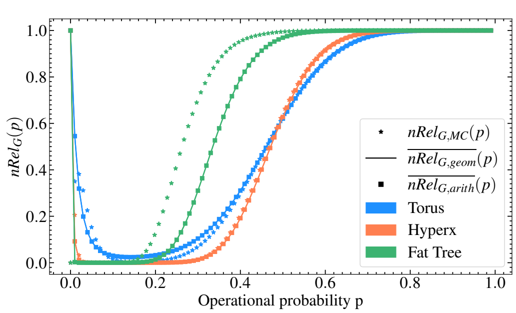



Fig. 11: Arithmetic Stochastic approximation $\overline{\text{nRel}_G^{\text{arith}}}(p)$, geometric stochastic approximation $\overline{\text{nRel}_G^{\text{geom}}}(p)$ and Monte Carlo simulations ($M = 10000$) for the node reliability of three structured network topologies: 3D Torus ($6 \times 6 \times 6$), HyperX (4D, 4 switches per dimension), and $k$-ary Fat-Tree ($k = 24$).

### B. Limitations of the stochastic approximations

Fig. 9 and 11 show that all three stochastic approximations match the Monte-Carlo benchmark for the *Torus* and *HyperX* networks but clearly underestimate the node reliability of the *Fat-Tree*. The discrepancy comes from treating the events $X_i$ $(i = 1, \ldots, N)$ as independent. In large, moderate-degree graphs each event involves (almost) disjoint neighbour sets, so the independence product is close to truth. In a Fat-Tree many leaves share the same aggregation and core switches; the events then overlap and the product formula is too small.

The example network in Fig. 12 illustrates this overlap for two nodes that share neighbours: the exact joint probability $\Pr[X_i \cap X_j]$ exceeds the independence product $\Pr[X_i]\Pr[X_j]$. The same mechanism, amplified over thousands of leaves, explains the gap observed on the Fat-Tree.

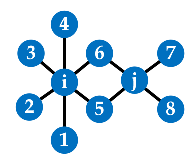



Fig. 12: Illustration of dependence between events $X_i$ and $X_j$. Each node is independently work with probability $p$. The probability of event $X_i = \{\text{Node } i \text{ fails}\} \cup \{\text{Node } i \text{ is active and at least one neighbor of } i \text{ is active}\}$ is $\Pr[X_i] = 1 - p(1-p)^6$. For node $j$ the probability is $\Pr[X_j] = 1 - p(1-p)^4$. The joint probability is $\Pr[X_i \cap X_j] = 1 - p(1-p)^6 - p(1-p)^4 + p^2(1-p)^8$. If the two events were independent, the probability is $\Pr[X_i \cap X_j] = \Pr[X_i]\Pr[X_j] = 1 - p(1-p)^6 - p(1-p)^4 + p^2(1-p)^{10}$.

## V. Analytical Optimization Strategy for the Node-Reliability-Based $k$-GRIP Problem

The reliability is an important robustness measure of graph $G(V, E)$. Given a connected graph $G = (V, E)$ and a budget of $k$ links to be added, a set $\mathcal{S} \subset \binom{N}{2} \setminus L$ of size $k$ that optimizes the robustness of $G$ is a common problem. Predari *et al.* refer to this optimisation problem as $k$-GRIP problem, short for graph robustness improvement problem [33]. We select the reliability polynomial $\text{Rel}_G(p)$ (node reliability polynomial $\text{nRel}_G(p)$) as robustness measure for $k$-GRIP. Within the stochastic approximations of reliability polynomial $\overline{\text{Rel}_G(p)}$ and node reliability polynomial $\overline{\text{nRel}_G(p)}$, we give optimal solutions $\mathcal{S}$ of $\overline{\text{Rel}_G(p)}$ and $\overline{\text{nRel}_G(p)}$. The optimal solution $\mathcal{S}$ are approximate optimal solutions for reliability-based $k$-GRIP and node-reliability-based $k$-GRIP problems.

By using the reliability polynomial as an objective measure, we design networks that optimize for specific reliability criteria. For example, to enhance a network's reliability, we might seek to add or reinforce connections (links) in such a way that the reliability polynomial achieves its highest possible values under expected operational probabilities. Since computing the exact expression of the reliability polynomial is NP-hard, it is not possible to provide an analytical solution to the reliability-based $k$-GRIP problem. However, a stochastic approximation of the reliability polynomial can be optimized analytically,

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2025.3607004



which offers a practical approach to solving the reliability-based $k$-GRIP problem.

We recall the stochastic approximation of the reliability polynomial in Eq. (24),

$$\overline{\text{Rel}_G}(p) = (1 - \varphi_D(1-p))^N,$$

and the stochastic approximation of the node reliability polynomial in Eq. (23),

$$\overline{\text{nRel}_G}(p) = (1 - \varphi_D(1-p))^{Np},$$

that both depend on the value of term $1 - \varphi_D(1-p)$. The function $f(x) = x^c$, where $c$ is a positive number, is monotonically increasing for $x$ in domain $[0,1]$. Thus a larger $1 - \varphi_D(1-p)$ contributes to a higher reliability and node reliability. Consequently, the problem of optimizing the stochastic approximations $\overline{\text{Rel}_G}(p)$ and $\overline{\text{nRel}_G}(p)$ reduces to maximizing the value of $1 - \varphi_D(1-p)$. We denote the graph obtained by adding the links of $\mathcal{S}$ into $G$ as $G' := G \cup \mathcal{S}$ and the degree distribution of graph $G'$ as $D'$, where $\mathcal{S} \subset \binom{N}{2} \setminus L$. Based on the analysis in our previous work [25], the term $1 - \varphi_D(1-p)$ can be expressed as

$$1 - \varphi_D(1-p) = \frac{1}{N} \sum_{i=1}^{N} \left(1 - (1-p)^{d_i}\right) \tag{33}$$

where $d_i$ is the degree of node $i$. Here we denote the degree vector of graph $G$ as $\mathbf{d} = [d_1, d_2, ..., d_N]$ and the degree change vector after $k$ links are added into $G$ as $\mathbf{a} = [a_1, a_2, ..., a_N]$, where $a_i \geq 0$. Then the degree vector of graph $G'$ becomes $\mathbf{d} = [d_1 + a_1, d_2 + a_2, ..., d_N + a_N]$. The reliability-based $k$-GRIP problem is then transformed into:

**Objective:**

$$\begin{aligned} &\max_{\mathbf{a}} 1 - \varphi_{D+\mathbf{a}}(1-p) \\ &= \max_{\mathbf{a}=[a_1,a_2,...,a_N]} \sum_{i=1}^{N} \left(1 - (1-p)^{d_i + a_i}\right) \end{aligned} \tag{34}$$

**Subject to:**

$$s.t. \sum_{i=1}^{N} a_i = 2k, a_i \geq 0, a_i \in \mathbb{Z} \tag{35}$$

Suppose there are two sets

$$\begin{aligned} \mathbf{a}_1 &= [a_1, a_2, ..., a_m, ..., a_n, ..., a_N] \\ \mathbf{a}_2 &= [a_1, a_2, ..., a_m + 1, ..., a_n - 1, ..., a_N] \end{aligned} \tag{36}$$

where $a_m, a_n \geq 1$.

The only difference between $\mathbf{a}_1$ and $\mathbf{a}_2$ is that the $m$th element of $\mathbf{a}_2$ is the $m$th element of $\mathbf{a}_1$ plus one, and the $n$th element of $\mathbf{a}_2$ is the $n$th element of $\mathbf{a}_1$ minus one. The difference of $\sum_{i=1}^{N} \left(1 - (1-p)^{d_i + a_i}\right)|_{\mathbf{a}_2}$ and $\sum_{i=1}^{N} \left(1 - (1-p)^{d_i + a_i}\right)|_{\mathbf{a}_1}$ is

$$\begin{aligned} \Delta &= \sum_{i=1}^{N} \left(1 - (1-p)^{d_i + a_i}\right)|_{\mathbf{a}_2} - \sum_{i=1}^{N} \left(1 - (1-p)^{d_i + a_i}\right)|_{\mathbf{a}_1} \\ &= p\left((1-p)^{d_n + a_n - 1} - (1-p)^{d_m + a_m}\right) \end{aligned} \tag{37}$$

The value of $\Delta$ is larger than 0 only when $d_n + a_n - 1 > d_m + a_m$. Here $d_m + a_m$ and $d_n + a_n$ are the degree of node $m$ and node $n$ after $k$ links are added to the graph according to set $\mathbf{a}_1$. When the degree $d_n + a_n$ of the node $n$ is larger that $d_m + a_m$ of node $m$ plus one, reconnecting the end of one link connect to node $n$ to node $m$ contribute to a better link-adding set $\mathcal{S}$. Here we define a $k$-GRIP descending restructuring, which disconnects the end of an added link to a node $n$ with degree $d_n + a_n$ in the graph $G'$ and reconnects that end of the link to another node $m$ with degree $d_m + a_m$, where $d_n + a_n - 1 > d_m + a_m$ and $a_n > 0$. The ascending restructuring is defined as the inverse of the descending restructuring. The above analysis shows that $1 - \varphi_{D+\mathbf{a}}(1-p)$ after descending restructuring is always larger than that before descending restructuring.

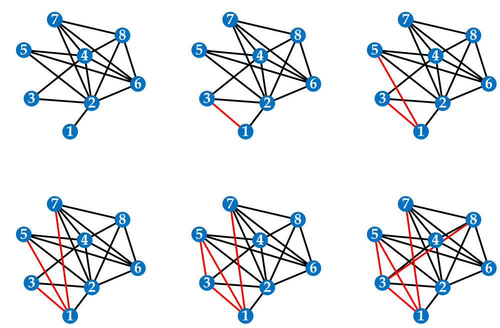



Fig. 13: Schematic illustration of the **Greedy-LD** algorithm, which iteratively adds 5 links (shown in red) to a graph with $N = 8$ nodes and $L = 16$ links. Each panel shows the intermediate graph after one additional link is added, highlighting how the algorithm prioritizes nodes with the lowest degree at each step.

We denote the set of all possible graphs, where $k$ links are added into graph $G$, as $\langle G \rangle$. Start from a random graph $G'$ in $\langle G \rangle$, any graph in $\langle G \rangle$ can be obtained by multiple descending and ascending restructurings. Since descending restructuring always contribute to a higher $\sum_{i=1}^{N} \left(1 - (1-p)^{d_i + a_i}\right)$, the optimal graph $G^*$ is a graph where no descending restructuring could occur. To construct the optimal graph $G^*$, links can be greedily added by connecting pairs of nodes with the lowest degrees, provided the link does not already exist. In this paper, the algorithm of greedily adding $k$ links between pairs of nodes with the lowest degrees that are not already connected is referred to as the **Greedy Lowest-Degree Pairing Link Addition (Greedy-LD) Algorithm**.

With a bucket list that stores nodes by current degree, the initial degree scan costs $\mathcal{O}(L)$ time, where $L$ is the number of links in the graph. Each of the $k$ link insertions then requires only constant time for bucket updates and an $\mathcal{O}(1)$ adjacency check, giving an overall running time of $\mathcal{O}(L + k)$. Fig. 14 shows the distribution of the average node reliability over $p \in [0,1]$ for all possible link addition configurations, where 5 links are added to base graphs with $N = 7, 8, 9$ nodes and $L = 11, 16, 24$ links, respectively. In all three cases, the **Greedy-LD** algorithm yields results that rank within the top 2% of all configurations, indicating that it consistently produces high-reliability structures with

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2025.3607004



**Algorithm 1** Greedy Link Addition Based on the Minimum Degree–Sum

**Input:** connected graph $G = (V, E)$, number of links to add $k$

**Output:** augmented graph $G^*$

compute initial degrees $\{d_v \mid v \in V\}$
**for** $t \leftarrow 1$ **to** $k$ **do**
**(a) score every missing link**
  **for all** $\{u, v\} \subset V$ **s.t.** $(u, v) \notin E$ **do**
    $s(u, v) \leftarrow d_u + d_v$ ▷ degree sum of the pair
  **end for**
**(b) identify best pairs**
  $s_{\min} \leftarrow \min_{\{u,v\}\notin E} s(u, v)$
  $C \leftarrow \big\{\{u, v\} \notin E \mid s(u, v) = s_{\min}\big\}$ ▷ all pairs with the smallest degree sum
**(c) random tie-break**
  uniformly pick one link $\{i, j\}$ from $C$
**(d) update graph and degrees**
  $E \leftarrow E \cup \{(i, j)\}$; $d_i \leftarrow d_i + 1$; $d_j \leftarrow d_j + 1$
**end for**
**return** $G^* = (V, E)$

simple computation. To further evaluate the performance of the **Greedy-LD Algorithm** in improving network reliability, we applied **Greedy-LD Algorithm** and other two link-adding strategies to real-world networks from the Network Repository [34] and compared their effectiveness. The first strategy, **Random Pairing Strategy**, adds links randomly between node pairs that are not already linked. The second strategy, **Greedy Highest-Degree Pairing Strategy**, focuses on adding links between the highest-degree nodes that are not yet connected, aiming to strengthen the already well-connected nodes. Figures in Fig. 15 illustrate that **Greedy-LD Algorithm** has the most significant effect in enhancing network reliability and node reliability, far outperforming random link addition and adding links between high-degree nodes. Adding links between high-degree nodes shows the worst performance, with almost no noticeable impact on network reliability. The simulation demonstrates that one of the most effective ways to enhance network robustness from the perspective of network connectivity is to add links between low-degree nodes.

## VI. Conclusion

Calculating the exact node-reliability polynomial is known to be NP-hard, prompting the need for efficient approximation techniques. The node-reliability polynomial of a graph has been approximated by Laplace's method. For the node-reliability polynomial, utilizing "the $C$-form" and "$S$-form" representations. We demonstrated that, by approximating these forms with probabilistic methods, significant computational efficiency can be achieved, while maintaining reasonable accuracy when approximating the node-reliability polynomial $\mathrm{nRel}_G(p)$.

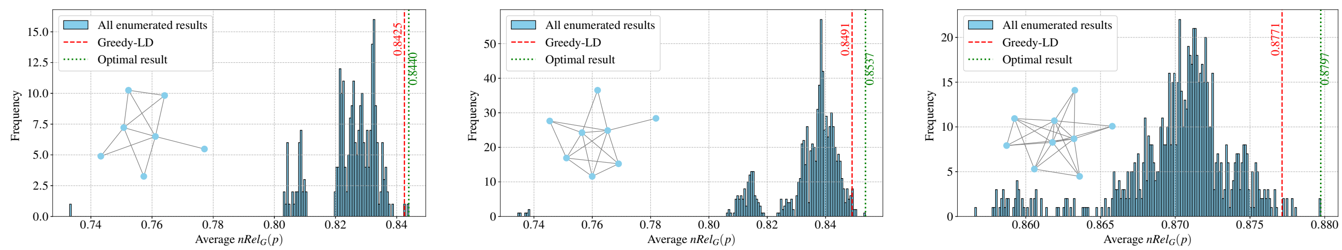
We proposed two simulation approaches: a standard Monte-Carlo estimator and a Laplace–Monte-Carlo variant. Because the Laplace term already captures the main behaviour of $nRel(p)$, the Laplace–Monte-Carlo method requires far fewer samples and is therefore computationally lighter while achieving the same level of accuracy as the plain Monte-Carlo approach. Additionally, we introduced three degree-based stochastic approximations—Laplace, arithmetic, and geometric—leveraging the probability-generating function of node degrees. These stochastic approximations provide quick and reasonably accurate estimates, particularly effective for large and dense networks.

Finally, we introduced a Greedy Lowest-Degree Pairing Link Addition (**Greedy-LD**) algorithm that simply connects pairs of nodes with the smallest current degree. Despite its simplicity and $\mathcal{O}(k + L)$ time complexity, Greedy-LD consistently ranks within the top $2\,\%$ of all possible link-addition configurations in our experiments, and provides larger reliability gains than both random link insertion and highest-degree pairing strategies.

**Acknowledgements** Van Mieghem has been funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 101019718).

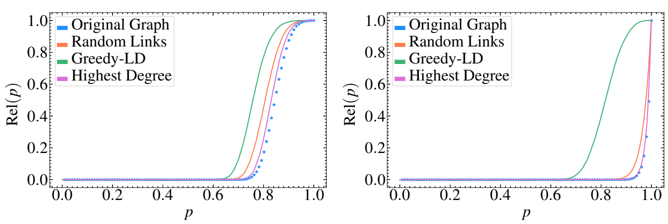
## References

[1] Hebert Pérez-Rosés. Sixty years of network reliability. *Mathematics in Computer Science*, 12:275–293, 2018.
[2] Edward F Moore and Claude E Shannon. Reliable circuits using less reliable relays. *Journal of the Franklin Institute*, 262(3):191–208, 1956.
[3] Michael O Ball. Complexity of network reliability computations. *Networks*, 10(2):153–165, 1980.
[4] Ian F Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. Wireless sensor networks: a survey. *Computer networks*, 38(4):393–422, 2002.
[5] Bianca Schroeder, Eduardo Pinheiro, and Wolf-Dietrich Weber. Dram errors in the wild: a large-scale field study. *ACM SIGMETRICS Performance Evaluation Review*, 37(1):193–204, 2009.
[6] Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak mathematical journal*, 23(2):298–305, 1973.
[7] WU Jun, Mauricio Barahona, Tan Yue-Jin, and Deng Hong-Zhong. Natural connectivity of complex networks. *Chinese physics letters*, 27(7):078902, 2010.
[8] Arpita Ghosh, Stephen Boyd, and Amin Saberi. Minimizing effective resistance of a graph. *SIAM review*, 50(1):37–66, 2008.
[9] Piet Van Mieghem. *Graph spectra for complex networks*. Cambridge University Press, 2023.
[10] Klaus Sutner, Appajosyula Satyanarayana, and Charles Suffel. The complexity of the residual node connectedness reliability problem. *SIAM Journal on Computing*, 20(1):149–155, 1991.
[11] Leslie G Valiant. The complexity of enumeration and reliability problems. *siam Journal on Computing*, 8(3):410–421, 1979.
[12] J Scott Provan and Michael O Ball. The complexity of counting cuts and of computing the probability that a graph is connected. *SIAM Journal on Computing*, 12(4):777–788, 1983.
[13] Charles J Colbourn, Appajosyula Satyanarayana, C Suffel, and Klaus Sutner. Computing residual connectedness reliability for restricted networks. *Discrete Applied Mathematics*, 44(1-3):221–232, 1993.
[14] Louis D Nel and Charles J Colbourn. Combining monte carlo estimates and bounds for network reliability. *Networks*, 20(3):277–298, 1990.
[15] Ilya B Gertsbakh and Yoseph Shpungin. *Models of network reliability: analysis, combinatorics, and Monte Carlo*. CRC press, 2016.
[16] Héctor Cancela and Mohamed El Khadiri. The recursive variance-reduction simulation algorithm for network reliability evaluation. *IEEE Transactions on Reliability*, 52(2):207–212, 2003.
[17] Jun Kawahara, Koki Sonoda, Takeru Inoue, and Shoji Kasahara. Efficient construction of binary decision diagrams for network reliability with imperfect vertices. *Reliability Engineering & System Safety*, 188:142–154, 2019.
[18] Jason I Brown, Charles J Colbourn, Danielle Cox, Christina Graves, and Lucas Mol. Network reliability: Heading out on the highway. *Networks*, 77(1):146–160, 2021.

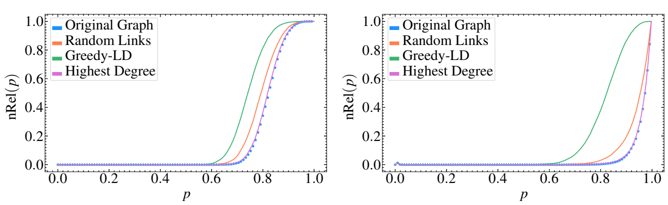



(a) $N = 7$, $L = 11$, $k = 5$, rank: 2/252 (b) $N = 8$, $L = 16$, $k = 5$, rank: 12/792 (c) $N = 9$, $L = 24$, $k = 5$, rank: 10/792

Fig. 14: Distribution of the average node reliability polynomial over $p \in [0, 1]$ for all possible configurations obtained by adding 5 links to a base graph with: (a) $N = 7$, $L = 11$, (b) $N = 8$, $L = 16$, and (c) $N = 9$, $L = 24$ links. The red dashed line shows the result of the **Greedy-LD** algorithm, while the green dotted line marks the optimal reliability among all configurations. In these three cases, the greedy strategy ranks 2nd out of 252, 12th out of 792, and 10th out of 792, respectively, demonstrating consistently near-optimal performance across different network sizes.




(a) n2c6 b3 $\text{Rel}_G(p)$ (b) USAir97 $\text{Rel}_G(p)$




(c) n2c6 b3 $\text{nRel}_G(p)$ (d) USAir97 $\text{nRel}_G(p)$

Fig. 15: Monte Carlo simulations results ($M = 10000$) of reliability polynomial $\text{Rel}_G(p)$ and node reliability polynomial $\text{nRel}_G(p)$ in two real-world graphs and graphs constructed by adding links to these graphs for three different strategies. 'n2c6 b3' is a simplicial complex network with $N = 1365$ nodes and $L = 5263$ links [34]. $l = 500$ links are added into the 'n2c6 b3' graph for three different strategies. 'USAir97' is the 1997 U.S. flight network with $N = 332$ nodes and $L = 2126$ links [34]. $l = 100$ links are added into the 'USAir97' graph in three different strategies. (a) Reliability polynomial $\text{Rel}_G(p)$. (b) Reliability polynomial $\text{Rel}_G(p)$. (c) Node reliability polynomial $\text{nRel}_G(p)$. (d) Node reliability polynomial $\text{nRel}_G(p)$.

[19] Stephen Eubank, Mina Youssef, and Yasamin Khorramzadeh. Using the network reliability polynomial to characterize and design networks. *Journal of complex networks*, 2(4):356–372, 2014.
[20] Willem Pino, Teresa Gomes, and Robert Kooij. A comparison between two all-terminal reliability algorithms. *Journal of Advances in Computer Networks*, 3(4):284–290, 2015.
[21] Velin Kounev, Martin Lévesque, David Tipper, and Teresa Gomes. On smart grid communications reliability. In *2015 11th International Conference on the Design of Reliable Communication Networks (DRCN)*, pages 33–40. IEEE, 2015.
[22] Jason I Brown, Theodore Kolokolnikov, and Robert E Kooij. New approximations for network reliability. *Networks*, 84(1):51–63, 2024.
[23] Piet Van Mieghem. *Performance analysis of complex networks and systems*. Cambridge University Press, 2014.
[24] William Feller. *An introduction to probability theory and its applications, Volume 2*, volume 2. John Wiley & Sons, 1991.
[25] Piet Van Mieghem and Xinhan Liu. Degree-based approximations for network reliability polynomials. *Journal of Complex Networks*, 13(2):cnaf001, 2025.
[26] Remco Van Der Hofstad. *Random graphs and complex networks*. Cambridge university press, 2024.
[27] Lorenzo Federico and Remco Van der Hofstad. Critical window for connectivity in the configuration model. *Combinatorics, Probability and Computing*, 26(5):660–680, 2017.
[28] Tomasz Luczak. Sparse random graphs with a given degree sequence. In *Random graphs*, volume 2, pages 165–182, 1992.
[29] Michael Molloy and Bruce Reed. A critical point for random graphs with a given degree sequence. *Random structures & algorithms*, 6(2-3):161–180, 1995.
[30] William J. Dally. Performance analysis of k-ary n-cube interconnection networks. *IEEE transactions on Computers*, 39(6):775–785, 1990.
[31] Jung Ho Ahn, Nathan Binkert, Al Davis, Moray McLaren, and Robert S Schreiber. Hyperx: topology, routing, and packaging of efficient large-scale networks. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, pages 1–11, 2009.
[32] Charles E Leiserson. Fat-trees: Universal networks for hardware-efficient supercomputing. *IEEE transactions on Computers*, 100(10):892–901, 1985.
[33] Maria Predari, Lukas Berner, Robert Kooij, and Henning Meyerhenke. Greedy optimization of resistance-based graph robustness with global and local edge insertions. *Social Network Analysis and Mining*, 13(1):130, 2023.
[34] Ryan A. Rossi and Nesreen K. Ahmed. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015.

**Xinhan Liu** is pursuing his PhD degree since November 2022 at Delft University of Technology, The Netherlands. He obtained the M.Sc. degree in Wireless Communication and Sensing from the same institution, in 2022. In 2019, he obtained a B.Sc. degree in Electronic Information Engineering at Tianjin University, China. In 2018, he was a visiting student at Technion - Israel Institute of Technology (funded by Technion), where he studied digital image processing. His main research interests include graph theory, network dynamics, opinion models, network reliability, shortest path problem, digital signal processing, and robustness of complex networks.

**Robert Kooij** has a background in mathematics: he received both his MSc and PhD degree cum laude at Delft University of Technology, in 1988 and 1993, respectively. From 1997 until 2003 he was employed at the research lab of KPN, the largest telecom operator in the Netherlands. From 2003 until 2018 he was employed at the ICT Unit of TNO, the Netherlands Organization of Applied Scientific Research. In 2011 he became principal scientist, conducting and managing research on Critical ICT Infrastructures. Since 2005 Robert is part-time affiliated with the Delft University of Technology, at the faculty of Electrical Engineering, Mathematics and Computer Science. Since 2010 he is a part-time full professor with the chair "Robustness of Complex Networks". From 2018 until 2020 professor Kooij lived in Singapore, where he got a position as principal research scientist at the Singapore University of Technology and Design, working on a project related to cyber resilience for critical infrastructures. Currently he is the head of the department of Quantum and Computer Engineering (QCE) at Delft University of Technology. He is also part-time affiliated with the Cyber Security Technologies group at TNO.

**Piet Van Mieghem** (Fellow, IEEE) is professor at the Delft University of Technology and chairman of the section Network Architectures and Services (NAS) since 1998. He is the author of four books: *Performance Analysis of Communications Networks and Systems*, *Data Communications Networking*, *Graph Spectra for Complex Networks*, and *Performance Analysis of Complex Networks and Systems*. He is a board member of the Netherlands Platform of Complex Systems, a steering committee member of the Dutch Network Science Society, an external faculty member at the Institute for Advanced Study (IAS) of the University of Amsterdam and an IEEE Fellow. He was awarded an Advanced ERC grant 2020 for ViSiON, Virus Spread in Networks.

Professor Van Mieghem received a Master degree and a Ph.D. degree in Electrical Engineering from the K.U.Leuven (Belgium) in 1987 and 1991, respectively. Before joining Delft, he worked at the Interuniversity Micro Electronic Center (IMEC) from 1987 to 1991. During 1993 to 1998, he was a member of the Alcatel Corporate Research Center in Antwerp. He was a visiting scientist at MIT (1992-1993) and a visiting professor at UCLA (2005), at Cornell University (2009), at Stanford University (2015) and at Princeton University (2022).

Currently, he serves on the editorial board of the OUP *Journal of Complex Networks*. He was a member of the editorial board of *Computer Networks* (2005-2006), the *IEEE/ACM Transactions on Networking* (2008-2012), the *Journal of Discrete Mathematics* (2012-2014) and *Computer Communications* (2012-2015).