# Evaluating Non-Hierarchical Overflow Loss Systems using Teletraffic Theory and Neural Networks

Yin-Chi Chan, *Member, IEEE,* Eric W. M. Wong, *Senior Member, IEEE,* and Chi Sing Leung, *Senior Member, IEEE*

***Abstract*—The Information Exchange Surrogate Approximation (IESA) is a powerful tool for estimating the blocking probability of non-hierarchical overflow loss systems (NH-OLSs), but can exhibit significant approximation errors in some cases. This letter proposes a new method of evaluating the blocking probability of generic NH-OLSs by combining machine learning with IESA. Specifically, we modify IESA by using neural networks (NN) to tune a newly introduced parameter in the IESA algorithm. Extensive numerical results for a simple NH-OLS show that our new hybrid method, which we call IESA+NN, is more accurate and robust than both base IESA and direct NN-based approximation of NH-OLS blocking probability, while remaining much more computationally efficient than computer simulation. Furthermore, due to the generic nature of our technique, IESA+NN is also easily extensible to more specialized stochastic models for communications and service systems, where base IESA has previously been applied.**

***Index Terms*—Teletraffic, neural networks, overflow loss systems**

## I. Introduction

MANY communications and service systems, such as cellular networks [1], [2], content distribution networks [3], and healthcare systems [4], [5] can be modeled as non-hierarchical overflow loss systems (NH-OLSs), in which servers are divided into groups, each request requires one server, and each server group serves some subset of the request types in the system. An incoming request *overflows* from one server group to the next until a suitable server is found, or is *blocked* and cleared from the system immediately if no such server is available; the probability of this is called the *blocking probability* and is a main performance metric of NH-OLSs.

A common feature in many NH-OLSs is mutual overflow [6], where congestion at one server group causes overflow to other server groups, which in turn become congested and yield overflow back to the original server group. While generally providing better performance than systems without mutual overflow [7], [8], such systems are difficult to analyze due to the curse of dimensionality: the state space of the system is exponential in the number of server groups, and no product-form solution exists for the steady-state probabilities [9]. Although the blocking probability of NH-OLSs with mutual overflow can be evaluated using simulation, this can be quite computationally expensive and infeasible for solving optimization problems where a large number of system configurations must be evaluated and compared, e.g. adaptive systems where such optimizations may occur at frequent intervals.

A major approximation approach for evaluating blocking probability in NH-OLSs with mutual overflow is to *decompose* the system into *independent* server groups [10], [11]; the well-known Erlang Fixed-Point Approximation (EFPA) [12] is an example of this. In [13], an improved decomposition method, the Information Exchange Surrogate Approximation (IESA), was proposed, which transforms the NH-OLS using a fictitious *surrogate model* before applying decomposition. However, as demonstrated in this letter, significant approximation errors remain in many cases.

Another method for evaluating blocking probability in NH-OLSs is using neural networks (NNs), thus offloading the computational effort to the training phase and allowing fast evaluation of the blocking probability once a trained NN is obtained. As an example, variants of the Extreme Learning Machine (ELM) algorithm were used in [14], [15] to train a neural network for the evaluation of blocking probability in optical networks. However, as demonstrated in this letter, direct estimation of blocking probability using NNs (hereafter called "direct NN") is not robust, especially when attempting to extrapolate outside the range of the training set.

In this letter, we propose a new blocking probability evaluation method for NH-OLSs with mutual overflow by introducing a new *tuning parameter* to IESA, using an NN to estimate the tuning parameter rather than directly estimating the blocking probability of the NH-OLS. Extensive numerical results demonstrate that our newly proposed method, which we call IESA+NN, is more accurate and robust than IESA or direct NN alone. Furthermore, since IESA+NN differs from IESA only in the introduction of a tuning parameter, it can therefore be applied to a wide range of scenarios where IESA has already been applied, e.g. emergency healthcare [5] and cellular networks [2], and to NH-OLSs with non-Poisson arrival traffic [16] or processor-sharing queues [17]. Finally, since computing the output of a trained NN is quite efficient, the amortized computational complexity of IESA+NN (i.e. excluding the one-time cost of generating training samples and building the NN) is low; in fact, base IESA, IESA+NN, and direct NN all have polynomial complexity in the number of server groups and/or the number of NN hidden nodes.

## II. System Model

As in [16], we consider an NH-OLS model with $G$ server groups each containing $N$ identical servers. Requests to the

Y. C. Chan, E. W. M. Wong, and C. S. Leung are with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong, e-mail: {ycchan26,eeewong,eeleungc}@cityu.edu.hk.

This work was jointly supported by the Key-Area Research and Development Program of Guangdong Province, China, under Grant 2019B010157002 and by the Research Grants Council (RGC) of Hong Kong under GRF Grant CityU 11104620.

system arrive according to a Poisson process with an intensity of $aGN$ Erlangs, require the service of any server in the system, and may attempt up to $k$ server groups in the system at random. The service times of the requests are independent and exponentially distributed with unit mean. Note that although simple formulas for the $N = 1$ case have been known for a full century [18], no scalable exact formula exists for the blocking probability in the general case, as no product form solution exists for the state probability distribution [9].

## III. Information Exchange Surrogate Approximation

A simple and classic method for overcoming the "curse of dimensionality" in overflow loss systems and networks is to apply two major simplifying assumptions. First, state dependencies between server groups are ignored, *decomposing* the system into a set of independent server groups [10]. Second, the offered traffic to each server group, including overflow traffic from other server groups, is often treated as if it were Poisson [12]. However, these two simplifying assumptions can lead to very large approximation errors in many cases when mutual overflow is present, and completely fail to capture the effect of $G$, the number of server groups in the NH-OLS, on the blocking probability [13], [16].

Therefore, in this letter, we will use an alternative method called the Information Exchange Surrogate Approximation (IESA) [13], [16] to estimate NH-OLS blocking probability. IESA is a decomposition-based approximation, meaning that it treats each server group as independent of the other groups. To compensate for approximation errors caused by this decomposition, IESA applies decomposition not to the NH-OLS model directly, but to a fictitious surrogate model, hereafter referred to as the IESA model.

In the IESA model, each request carries two attributes: $\Delta$, the set of previously attempted server groups, and $\Omega$, an estimate of the number of fully occupied server groups in the system. The $\Omega$ attribute is used for information exchange of the congestion level in the NH-OLS: in addition to increasing by one for each overflow, an incoming request to a server group will exchange its $\Omega$ attribute with that of the highest-$\Omega$ request in service if higher than that of the incoming request. Additionally, an overflowing $(\Delta, \Omega)$-request will, with probability $\pi_{n,\Omega}$, abandon the NH-OLS immediately without attempting any additional server groups, where

$$\pi_{\Omega,n} = \begin{cases} 0, & \Omega < k \\ \frac{\binom{\Omega}{k-n}}{\binom{G}{k-n}}, & k \le \Omega \le G. \end{cases} \tag{1}$$

Note that $\pi_{\Omega,n} = 1$ if $k = n$ or $\Omega = G$.

A full description of IESA for the current NH-OLS model can be found in [16]. A summary is as follows:

$$a_{j,n} = \begin{cases} \lambda, & n = j = 0 \\ 0, & n = 0,\ j \neq 0 \\ w_{j,n}\left(1 - \pi_{j,n}\right), & \text{otherwise}, \end{cases} \tag{2}$$

$$\tilde{a}_{j,n} = \textstyle\sum_{i=n}^{j} a_{i,n}, \quad A_j = \textstyle\sum_{n=0}^{j} \tilde{a}_{j,n},$$

$$b_j = E\left(A_j, N\right) \tag{3}$$

$$w_{j,n} = a_{j-1,n} b_{j-1} + \tilde{a}_{j-2,n-1}\left(b_{j-1} - b_{j-2}\right) \tag{4}$$

TABLE I
Notation for IESA

| Symbol | Definition |
|---|---|
| $\lambda$ | Offered load to each server group composed of fresh requests |
| $a_{j,n}$ | Offered load to each server group composed of requests with $\lvert\Delta\rvert = n$ and $\Omega = j$ |
| $\tilde{a}_{j,n}$ | Offered load to each server group composed of requests with $\lvert\Delta\rvert = n$l and $\Omega = n, n+1, \ldots, j$ |
| $w_{j,n}$ | Overflow from each server group composed of requests with $\lvert\Delta\rvert = n$ and $\Omega = j$ |
| $A_j$ | Offered load to each server group composed of requests with $\Omega \le j$ |
| $b_j$ | Per-server-group blocking probability of requests at level j of the IESA hierarchy, i.e. only considering requests with $\Omega \le j$ |

$$\hat{P}_{\text{IESA}} = 1 - \frac{\sum_{n=1}^{k}\sum_{j=n}^{G} w_{j,n} P_{j,n}}{\lambda} = 1 - \frac{A_{G-1}\left(1 - b_{G-1}\right)}{\lambda}, \tag{5}$$

with notation defined as in Table I, where (3) denotes the Erlang B formula [18]. The above equations define a recursive algorithm for IESA in $\Omega = 0 \ldots G$ and $n = 0 \ldots k$. Finally, when referring to the application of IESA to a specific set of NH-OLS parameters $\boldsymbol{x}_i$, we will use the notation $\hat{P}_{\text{IESA}}\left(\boldsymbol{x}_i\right)$ to refer to the blocking probability of that particular NH-OLS.

## IV. Direct NN approach

Apart from simulation and the IESA approximation described in the preceding section, we can estimate the blocking probability of NH-OLSs using machine learning. In what we call here as the direct NN approach, we use a single-layer feedforward network (SLFN) architecture which we train using the random-search-enhanced error-minimized extreme learning machine (EEM-ELM) algorithm described in [15], [19]. Note that [20], [21] demonstrated that ELM algorithms, including EEM-ELM, have universal approximation ability, despite the random input weights and biases of the hidden nodes. As only the output weights of the SLFN need to be trained, ELM algorithms such as EEM-ELM do not require backpropagation [22], leading to a much more computationally efficient training algorithm with fewer hyperparameters (which are the parameters of the EEM-ELM algorithm itself rather than of the NH-OLS to be evaluated) than many other machine learning algorithms. For an overview on ELM algorithms, see [23].

Compared to the original ELM algorithm [22], EEM-ELM has two main differences. First, in EEM-ELM, hidden nodes added to the NN *incrementally* in an iterative process. Second, EEM-ELM generates *multiple* candidate groups of hidden nodes in each iteration and only adds to the NN the group with the largest resulting reduction of the estimation error. This has been shown to reduce the number of hidden nodes required for a given estimation error threshold [15], [19]. EEM-ELM thus has four hyperparameters: $L_0$, the initial number of hidden nodes in the NN, $L_{\max}$, the final number of hidden nodes, $j$, the number of candidate groups of new hidden nodes to consider in each iteration, and $\delta L$, the number of hidden nodes in each candidate group. We also define $N_T$ as the number of samples in the training set, $\boldsymbol{x}_i = (\log G_i, \log k_i, \log N_i, a_i)$ as the input vector of the $i$th training sample (with $G_i$, $k_i$, $N_i$, and $a_i$ defined as in Section II), and

$$o_i = \log P(\boldsymbol{x}_i) \tag{6}$$

as the target output of the $i$th training sample (where $P(\boldsymbol{x}_i)$ is the simulated blocking probability of the NH-OLS corresponding to the $i$th training sample). As in [15], we use a sigmoid activation function for the hidden nodes. The output of the trained NN is denoted $\hat{o}(\boldsymbol{x}) = \sum_\ell \beta_\ell / [1 + \exp(-\boldsymbol{w}_\ell^{\mathrm{T}} \boldsymbol{x} + a_\ell)]$ where $\boldsymbol{w}_\ell$, $a_\ell$, and $\beta_\ell$ are the input weight vector, activation bias, and output weight for the $\ell$th hidden node, respectively. The error function to be minimized is therefore $\varepsilon = \sum_i (o_i - \hat{o}(\boldsymbol{x}_i))^2$, summing over all training inputs $i$. Finally, the estimated blocking probability for any given NH-OLS with parameters $\boldsymbol{x}$ is $\hat{P}_{\mathrm{NN}}(\boldsymbol{x}) = \exp(\hat{o}(\boldsymbol{x}))$. For a full description of the EEM-ELM algorithm, see [15], [19].

## V. IESA+NN

Although more accurate than previous decomposition-based methods, numerical results [13], [16] have revealed areas where IESA still exhibits significant approximation error. To obtain the high approximation capability of NN, while retaining IESA's ability to capture part of the underlying structure of the NH-OLS, in this letter we propose a new approximation method which we call IESA+NN. To obtain IESA+NN, we add a tuning parameter to IESA, which we set using a NN trained with the EEM-ELM algorithm. Specifically, we change (1) and (2) to

$$\pi_{\Omega,n,\tau} = \begin{cases} 0, & \tau\Omega < k \\ \min\left\{ \frac{\binom{\tau\Omega}{k-n}}{\binom{G}{k-n}}, 1 \right\}, & \text{otherwise} \end{cases} \tag{7}$$

and

$$a_{j,n} = \begin{cases} \lambda, & n = j = 0 \\ 0, & n = 0,\ j \neq 0 \\ w_{j,n}(1 - \pi_{j,n,\tau}), & \text{otherwise,} \end{cases}$$

respectively, and replace $\hat{P}_{\mathrm{IESA}}(\boldsymbol{x}_i)$ in (5) with $P_{\mathrm{IESA+NN}}(\tau, \boldsymbol{x}_i)$. The value of $\tau$ such that $P_{\mathrm{IESA+NN}}(\tau, \boldsymbol{x}_i) = P(\boldsymbol{x}_i)$ is denoted as $\tau^*(\boldsymbol{x}_i)$, and can easily be found via bisection as $P_{\mathrm{IESA+NN}}(\tau, \boldsymbol{x}_i)$ is non-decreasing in $\tau$. Since bisection is of low computational complexity (linear in the number of significant digits), the NN training costs of both approaches are similar. Note that $P_{\mathrm{IESA+NN}}(1, \boldsymbol{x}_i) = \hat{P}_{IESA}(\boldsymbol{x}_i)$.

As in the direct NN approach, the NN input is the collection of NH-OLS parameters, i.e. $\boldsymbol{x}_i = (\log G_i, \log k_i, \log N_i, a_i)$. However, (6) is changed to $o_i = \log \tau^*(\boldsymbol{x}_i)$. Finally, the estimated blocking probability for any given NH-OLS with parameters $\boldsymbol{x}$ is $\hat{P}_{\mathrm{IESA+NN}}(\boldsymbol{x}) = P_{\mathrm{IESA+NN}}(\hat{\tau}(\boldsymbol{x}), \boldsymbol{x})$ ,where $\hat{o}(\boldsymbol{x})$ is the NN output and $\hat{\tau}(\boldsymbol{x}) = \exp(\hat{o}(\boldsymbol{x}))$ is the estimated tuning parameter to be substituted into (7).

## VI. Numerical results

Note that the neural networks in both direct NN and IESA+NN share the same input parameters and differ only in target output. For both direct NN and IESA+NN, a training set was constructed as follows:

- $G = 10$, 12, 16, 20, 30, 50, 70, 100, 120, 150, 200, 300, 500, 700, 1000
- $k = 4$, 6, 8, 10, 12, 16, 20, 30, 50, 70, 100, 120, 150, 200, 300, 400, ..., 1000 such that $k \leq G$
- $N = 1$, 2, 3, 5, 7, 10, 12, 15, 20, 30, 50, 70, 100
- $a = 0.3$ to 1.0 in increments of 0.025

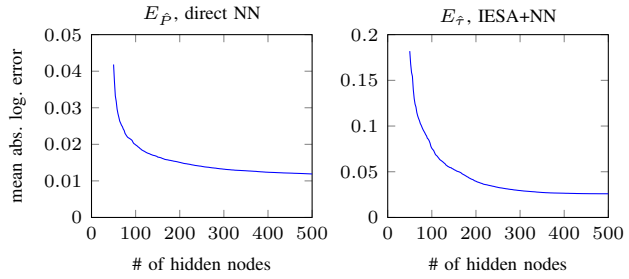


Fig. 1. Mean absolute log error of EEM-ELM as applied to the training set.

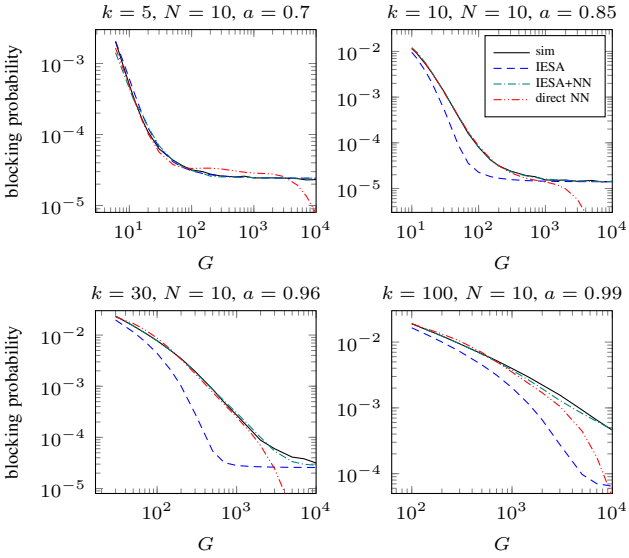


Fig. 2. Simulated and estimated blocking probabilities with respect to $G$.

We also require $\hat{P}_{\mathrm{IESA}} > 10^{-7}$ due to inaccuracy of the simulation result for low blocking probabilities, using the more computationally efficient IESA result rather than the simulation result for the filtering of the training set cases. The final size of the training set is $N_T = 20,637$.

Then, for both direct NN and IESA+NN, the NNs were trained using the EEM-ELM algorithm, with an initial size of $L_0 = 50$, $\delta L = 2$, $J = 50$, and $L_{\max} = 500$. Fig. 1 shows the mean absolute error of $\{o(\boldsymbol{x}_i)\}_{i=1}^{N_T}$ for each NN, i.e. $E = (1/N_T) \sum_{i=1}^{N_T} |\hat{o}(\boldsymbol{x}_i) - o(\boldsymbol{x}_i)|$. Note that this is equivalent to the mean absolute logarithmic error (MALE) of $\{\hat{P}_{NN}(\boldsymbol{x}_i)\}_{i=1}^{N_T}$ and $\{\hat{\tau}(\boldsymbol{x}_i)\}_{i=1}^{N_T}$, respectively: $E_{\hat{P}} = \frac{1}{N_T} \sum_{i=1}^{N_T} \left| \log_{10} \frac{\hat{P}_{NN}(\boldsymbol{x}_i)}{P(\boldsymbol{x}_i)} \right|$ and $E_{\hat{\tau}} = \frac{1}{N_T} \sum_{i=1}^{N_T} \left| \log_{10} \frac{\hat{\tau}(\boldsymbol{x}_i)}{\tau^*(\boldsymbol{x}_i)} \right|$. It is demonstrated that the MALE decreases as the number of hidden nodes increases. Note that $E_{\hat{P}}$ and $E_{\hat{\tau}}$ are *not* directly comparable, and the aim of Fig. 1 is simply to demonstrate that for both NNs, around 500 hidden nodes is sufficient.

In the following text, we compare the accuracy of base IESA, direct NN, and IESA+NN relative to simulation results, i.e. the values $\hat{P}_{\mathrm{IESA}}$, $\hat{P}_{\mathrm{NN}}$, and $\hat{P}_{\mathrm{IESA+NN}}$ relative to $P$, for a variety of test cases. The simulation values $P$ were obtained using Markov-chain simulation with $10^8$ arrivals per simulation run. Similar results were obtained using Levenberg–Marquardt backpropagation (LMBP); however, the results suggest slightly better extrapolation ability of EEM-ELM compared to LMBP when applying IESA+NN to cases outside the training set range. We thus only show the EEM-ELM-based results here.

Fig. 2 shows the blocking probability of an NH-OLS as

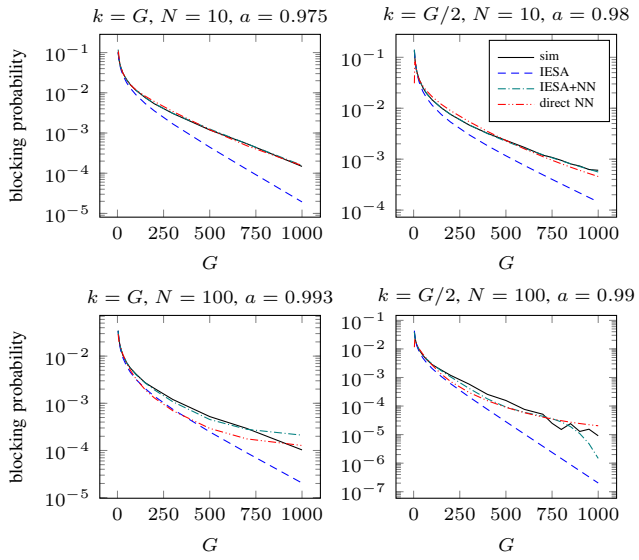


Fig. 3. Simulated and estimated blocking probabilities with respect to $G$ where both $G$ and $k$ increase in a fixed ratio.

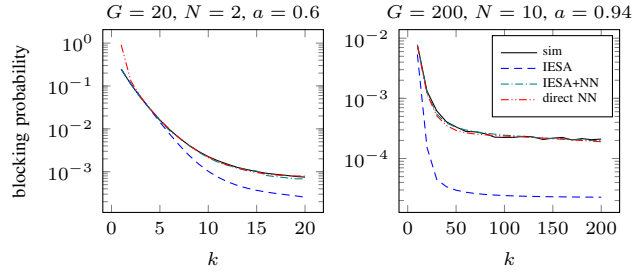


Fig. 4. Simulated and estimated blocking probabilities with respect to $k$.

$G$ increases with $k$ fixed. It was proved in [16] that base IESA is asymptotically exact in such cases as $G \to \infty$, i.e. $\hat{P}_{\text{IESA}} \to P$. This is supported by our numerical results, with the rate of convergence dependent on $k$. In addition, IESA+NN also appears to be asymptotically exact as $G \to \infty$, i.e. $\hat{P}_{\text{IESA+NN}} \to P$, whereas $\hat{P}_{\text{NN}}$ quickly diverges from $P$ for $G > 1000$, the maximum value of $G$ in the training dataset. This demonstrates the poor extrapolation ability of direct NN compared to the IESA+NN approach. Overall, the numerical results demonstrate that IESA+NN is significantly more accurate and robust than both direct NN and base IESA.

Fig. 3 shows the blocking probability of an NH-OLS as both $G$ and $k$ increase in a fixed ratio. The results demonstrate that IESA+NN is significantly more robust than both base IESA and direct NN. In particular, direct NN performs poorly in the $k = G/2$ case for small values of $G$ and $k$; however, direct NN also performs poorly in the bottom-left case even for larger values of $G$ and $k$, and is in fact less accurate than base IESA for cases around $G = k = 100$. On the other hand, in the bottom-right case, IESA+NN exhibits problems when $G$ and $k$ are both very large, due to the larger uncertainty in the simulation results and the low number of such cases in the training set (only cases with $a$ equal or close to 1 are admitted due to the low blocking probability of the other cases).

Fig. 4 shows the blocking probability of an NH-OLS as $k$ increases with $G$ fixed. The numerical results demonstrate that while both IESA+NN and direct NN are more accurate than base IESA as $k$ increases, pure NN is not as robust as

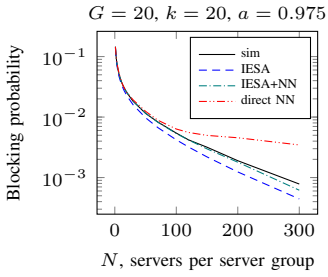


Fig. 5. Simulated and estimated blocking probabilities with respect to $N$.

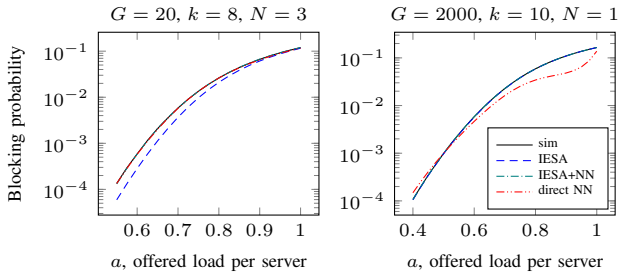


Fig. 6. Simulated and estimated blocking probabilities with respect to $a$.

IESA+NN when both $G$ and $k$ are small, as shown in the left side of Fig. 4.

Fig. 5 shows the blocking probability of an NH-OLS with respect to $N$. The results show that IESA+NN is much more accurate than IESA or direct NN for large $N$, particularly outside the training set range of $N \leq 100$.

Fig. 6 shows the blocking probability of an NH-OLS with respect to $a$. For the $G = 20$ case, direct NN and IESA+NN are both quite accurate for low loads, but pure IESA becomes less accurate as the offered load decreases. In the $G = 2000$ case, pure IESA and IESA+NN are both quite accurate for the entire range of $a$ shown, but direct NN is quite inaccurate for high loads.

## VII. NH-OLSs with Non-Poisson Inputs and Non-Exponential Service Times

In this section, we expand our methodology to evaluate the blocking probability of NH-OLSs with non-Poisson inputs and non-exponential service times. Although IESA has been extended in [16] to handle non-Poisson input traffic, it still cannot model the effect of non-exponential service times. We can use IESA+NN to resolve this issue, by applying the IESA part of IESA+NN as if the service times were exponentially distributed and relying on the NN part of IESA+NN to "correct" for the simplifying assumptions used.

We consider a set of NH-OLS configurations similar to that defined in Section II, but change the request service time distribution to lognormal with mean 1.0 and standard deviation $\sigma$. Note lognormal distributions occur frequently in human behavior models, e.g. call center service times [24] and ICU patient stays [25]. Additionally, each request must now attempt a *preferred* server group before randomly attempting up to $k-1$ other groups, such that the arrival process of fresh requests to each preferred server group forms an interrupted Poisson process [26] with intensity $Na$ and peakedness (variance-to-mean ratio) $z$.

Our training set is constructed as follows:

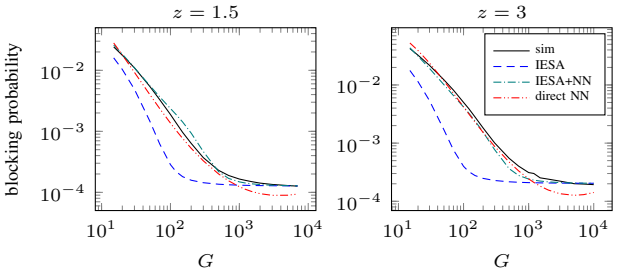


Fig. 7. Simulated and estimated blocking probabilities with respect to $G$, for an NH-OLS with $k = 12$, $N = 10$, $a = 0.9$, and $\sigma = 2$.

- $G = 10$, 15, 20, 50, 100, 200, 500, 1000, 20000
- $k = 4$, 6, 10, 15, 20, 70, 100, 150, 200 such that $k \leq G$
- $N = 1$, 2, 3, 5, 10, 20, 50, 100
- $a = 0.3$ to 1.0 in increments of 0.025
- $z = 1.1$, 1.25, 1.5, 2, 2.5, 3
- $\sigma = 0.5$, 1, 1.5, 2, 2.5, 3
- $P_{\text{IESA}} > 10^{-7}$ and $P > 5 \times 10^{-7}$

where $P_{\text{IESA}}$ and $P$ are the blocking probability of the NH-OLS as evaluated using the extended IESA method (i.e. [16]) and simulation, respectively. However, for $G = 20000$, instead of using simulation, which can be time-consuming and/or inaccurate for large $G$, we use previously proven asymptotic properties of IESA shown in [16] and assume $P = P_{\text{IESA}}$.

Fig. 7 shows the blocking probability of the NH-OLS with respect to $G$ for different values of $z$. While the results demonstrate that both direct NN and IESA+NN are significantly more accurate than IESA alone, IESA+NN is the only approximation of the three to maintain robustness across the entire range of $G$ shown, whereas direct NN is inaccurate for $G > 1000$.

## VIII. Concluding Remarks

In this letter, we considered a simple NH-OLS model with mutual overflow and showed that even accounting for symmetries, the number of distinct states is in general too many for exact analysis, whereas simulation is computationally expensive and existing approximation methods are not accurate and robust enough. We proposed a new approximation method, IESA+NN, based on introducing a control parameter into IESA which we tune using a neural network. Extensive numerical results demonstrate that IESA+NN is more accurate and robust than both base IESA and the direct NN approach. Furthermore, as noted in Section I, the amortized computational complexity (i.e. excluding one-time cost of generating training samples and building the NN) of IESA+NN is quite low and on par with IESA and direct NN. IESA+NN therefore has potential applications in self-adaptive systems where resource allocation, routing and/or admission control are adjusted on a periodic basis, whereas simulation would take too long for such purposes. Additionally, since IESA+NN differs from IESA only in the introduction of tuning parameter $\tau$, it can be applied to a wide range of scenarios where IESA has already been applied.

Finally, promising results were also obtained for the application of IESA+NN to NH-OLSs with non-Poisson inputs and non-exponential service times, although further research is desirable to further increase accuracy and robustness.

## References

[1] B. Eklundh, "Channel utilization and blocking probability in a cellular mobile telephone system with directed retry," *IEEE Trans. Commun.*, vol. 34, no. 4, pp. 329–337, Apr. 1986.
[2] J. Wu, E. W. M. Wong, and M. Zukerman, "Performance analysis of green cellular networks with selective base-station sleeping," *Perform. Evaluation*, vol. 111, pp. 17–36, May 2017.
[3] J. P. Muñoz-Gea, S. Traverso, and E. Leonardi, "Modeling and evaluation of multisource streaming strategies in P2P VoD systems," *IEEE Trans. Consum. Electron.*, vol. 58, no. 4, pp. 1202–1210, Nov. 2012.
[4] N. Litvak, M. van Rijsbergen, R. J. Boucherie, and M. van Houdenhoven, "Managing the overflow of intensive care patients," *Eur. J. Oper. Res.*, vol. 185, no. 3, pp. 998–1010, Mar. 2008.
[5] Y.-C. Chan, E. W. M. Wong, G. Joynt, P. Lai, and M. Zukerman, "Overflow models for the admission of intensive care patients," *Health Care Manag. Sci.*, vol. 21, no. 4, pp. 554–572, Dec. 2018.
[6] B. Hennion, "Feedback methods for calls allocation on the crossed traffic routing," in *Proc. ITC 9*, 1979.
[7] S. Gurumurthi and S. Benjaafar, "Modeling and analysis of flexible queueing systems," *Nav. Res. Logist.*, vol. 51, no. 5, pp. 755–782, Aug. 2004.
[8] S. C. Graves, "Flexibility principles," in *Building Intuition*, ser. Int. Ser. Oper. Res. Manag. Sci. Springer, 2008, vol. 115, ch. 3, pp. 33–49.
[9] A. Hordijk, "Insensitive bounds for performance measures," in *Proc. ITC 12*, Jun. 1988.
[10] R. B. Cooper and S. S. Katz, "Analysis of alternate routing networks with account taken of nonrandomness of overflow traffic," Bell Telephone Laboratories, Memo. MM64-3122-2, 1964.
[11] G. Koole and J. Talim, "Exponential approximation of multi-skill call centers architecture," in *Proc. QNETs*, Jul. 2000, pp. 23/1–10.
[12] F. P. Kelly, "Blocking probabilities in large circuit-switched networks," *Adv. Appl. Probab.*, vol. 18, no. 2, pp. 473–505, Jun. 1986.
[13] E. W. M. Wong, J. Guo, B. Moran, and M. Zukerman, "Information exchange surrogates for approximation of blocking probabilities in overflow loss systems," in *Proc. ITC 25*, Sep. 2013.
[14] H. C. Leung, C. S. Leung, E. W. M. Wong, and S. Li, "Extreme learning machine for estimating blocking probability of bufferless OBS/OPS networks," *IEEE J. Opt. Commun. Netw.*, vol. 9, no. 8, pp. 682–692, Jul. 2017.
[15] S. Li, H. C. Leung, E. W. M. Wong, and C. S. Leung, "Enhancement of extreme learning machine for estimating blocking probability of OCS networks with fixed-alternate routing," *IEEE Access*, vol. 7, pp. 52 319–52 330, Apr. 2019.
[16] Y.-C. Chan and E. W. M. Wong, "Blocking probability evaluation for non-hierarchical overflow loss systems," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2022–2036, May 2018.
[17] Y.-C. Chan, J. Guo, E. W. M. Wong, and M. Zukerman, "Performance analysis for overflow loss systems of processor-sharing queues," in *Proc. IEEE INFOCOM '15*, Apr. 2015, pp. 1409–1417.
[18] E. Brockmeyer, H. L. Halstrøm, and A. Jensen, *The Life and Works of A. K. Erlang*, ser. Trans. Danish Acad. Tech. Sci., Copenhagen, 1948, no. 2.
[19] Y. Lan, Y. C. Soh, and G.-B. Huang, "Random search enhancement of error-minimized extreme learning machine," in *Proc. ESANN*, Apr. 2010, pp. 327–332.
[20] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 879–892, Jul. 2006.
[21] R. Zhang, Y. Lan, G.-B. Huang, and Z.-B. Xu, "Universal approximation of extreme learning machine with adaptive growth of hidden nodes," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 2, pp. 365–371, 2012.
[22] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, Dec. 2006.
[23] G. Huang, G.-B. Huang, S. Song, and K. You, "Trends in extreme learning machines: A review," *Neural Netw.*, vol. 61, pp. 32–48, Jan. 2015.
[24] S. Gualandi and G. Toscani, "Human behavior and lognormal distribution. A kinetic description," *Math. Models Methods Appl. Sci.*, vol. 29, no. 04, pp. 717–753, Apr. 2019.
[25] J. C. Lowery, "Multi-hospital validation of critical care simulation model," in *Proc. 25th Winter Simulation Conference (WSC)*, 1993, pp. 1207–1215.
[26] A. Kuczura, "The interrupted Poisson process as an overflow process," *Bell Syst. Tech. J.*, vol. 52, no. 3, pp. 437–448, Mar. 1973.