A Computationally Efficient Evolutionary Algorithm for Multi-Objective Network Robustness Optimization

[[1]](#footnote-1)

Shuai Wang, Jing Liu, *Senior Member*, *IEEE*, and Yaochu Jin, *Fellow*, *IEEE*

*Abstract*—The robustness of complex networks is of great significance. Great achievements have been made in robustness optimization based on single measures, however, such networks may still be vulnerable to multiple attack scenarios. Therefore, recently, multi-objective robustness optimization of networks has received increasing attention. Nevertheless, several challenges remain to be addressed, including the different computational complexities in evaluating the objectives, insufficient diversity in the obtained networks, and high computational costs of the search process. In this paper, we address the aforementioned challenges by developing a computationally efficient multi-objective optimization algorithm. Based on the unique features of complex networks, a new parallel fitness evaluation method guided by a network property parameter is designed, and embedded in a reference vector guided multi-objective evolutionary algorithm. In addition, a surrogate ensemble with heterogeneous inputs is constructed based on graph embedding information to efficiently estimate multiple robustness of networks. The proposed algorithm is validated on synthetic and real-world network data and our results show that the designed algorithm outperforms the state-of-the-art method with a marked improvement on the computational efficiency. Compared with other single-objective optimization methods, the proposed algorithm demonstrates a considerable exploitation ability.

*Index Terms*—Complex networks, robustness, multi-objective optimization, surrogates, graph embedding

# INTRODUCTION

D

ifferent kinds of networked systems, such as the Internet [1], power grids [2], and gene regulatory networks [3], are commonly present in biological and societal systems. These systems tend to have complex structures, owing to which, their dynamics is hard to analyze. To represent systems in an intuitive manner, the complex network model [4] was recently developed. On a macro level, the topology of a network represents the structure of a system; and on a micro level, nodal connections in a network represent the interaction relationship among system components. With the help of complex networks, researchers have discovered a number of network properties, such as the community structure [3], degree distribution [5], and mixing patterns [6], thereby clarifying the underlying important nature of modern networked systems.

Another aspect that adds to the complexity of networked systems is that the operating environment of such networks involves uncertainty and disturbances. Complex networks are expected to be robust against uncertainty, disturbances and attacks; that is, networks must be able to function properly even in the presence of attacks and errors. Several measures have been designed to evaluate the robustness of networks considering different damage scenarios, including node-based attacks [2], link-based attacks [7], and cascading failures [8]. These measures provide criteria for judging whether a network system can function reliably in the presence of disturbances. The measures can also guide an optimization algorithm to enhance the network robustness. Encouraging results have been achieved by rewiring network topologies to enhance the robustness. For example, heuristic optimization techniques [2, 7] have been shown to be effective in enhancing the robustness of networks against attacks with a low space complexity. Moreover, population-based optimization methods can achieve satisfactory results using problem-directed operators [9, 10].

The researchers in [11] reported that, networked systems may be subjected to diversified destructive attacks, and thus, the single-objective robustness optimization methods for networks may not be sufficient in real-world applications. In this regard, it is necessary to consider multiple types of attacks when optimizing the robustness. Further, different robustness measures may conflict with one another [11, 12], thereby rendering it challenging to simultaneously enhance the robustness of networks against multiple types of attacks. Given many efficient evolutionary algorithms available to solve multi-objective optimization problems (MOPs) [13, 14], increasing attention has been paid to design robust networks using such optimization techniques. A multi-objective evolutionary algorithm has been designed in [15] to enhance the robustness of scale-free networks [5] against both node-based and link-based attacks. In general, a set of solutions with various structures can be generated to reach a trade-off between the two objectives. In terms of the computational efficiency, the multi-objective approach exhibits a notable superiority over single-objective ones, and each non-dominated solution can be considered as a result of the weighted optimization between the two objectives. These solutions provide diverse candidates against a wide range of damage scenarios, and this aspect is validated in this work as well. Such an approach has been shown to be beneficial in the design of networked systems such as water infrastructures [16] and power transmission facilities [17]. Similar methods have also been applied to realize the controllability promotion [18] and cooperative optimization [19] of networks.

However, several challenges still remain to be addressed in the multi-objective design of robust networks against multiple attacks. First, the computational latency [20] for evaluating the robustness under node-based and link-based attacks is non-uniform, i.e., the computational complexities for evaluating the robustness against node-based and link-based attacks are evidently different, since the number of links tends to be significantly larger than that of nodes in a network [7, 11]. Thus, a strategy that can mitigate the difference in the computational complexity is therefore highly desired to improve the efficiency of optimization processes. Moreover, the solution space in network topology optimization is extremely large, since plenty of structural variants exist in even small-size networks. Many of these variants always get similar assessments under a certain robustness measure, which may hinder optimization processes. Furthermore, decision variables for the network optimization problem are network structures, and this problem is discrete and highly non-linear; consequently, the multi-objective optimization of networks a challenging problem. Finally, it is time-consuming to evaluate the robustness of networks especially for large-scale ones, and the computational cost for robustness optimization is often prohibitive [2, 9, 15]. As shown in [15], it takes days of running time to handle networks with 500 nodes, making most existing optimization algorithms cannot be used to solve real-world problems.

Many researchers attempted to address the above challenges in the context of evolutionary optimization. To solve MOPs with non-uniform latencies, several parallel strategies were designed in [20, 21] to improve the efficiency when utilizing computational resources. However, the techniques that how to properly adapt these techniques to perform the multiple robustness optimization of networks remain an open question. A reference vector guided evolutionary algorithm was introduced in [22] to incorporate user preferences into multi-objective optimization processes and further improve the search performance in solving MOPs with more than three objectives. However, to date, such algorithms have hardly been studied in multi-objective network robustness optimization [15, 18, 19].

Another aspect that prevents a wider application of existing multi-objective network optimization methods to the robustness optimization of complex networks is the associated high computational cost when evaluating the robustness. In the context of evolutionary optimization, a large body of research has been carried out to solve computationally expensive numerical optimization problems using surrogate models [23 - 25]. By means of making approximations of the original expensive real objective functions, surrogate models can guide the search process at a reduced computational cost. However, most of existing surrogate-assisted methods focused on solving continuous numerical optimization problems, and few approaches considered network data which are discrete, non-linear, and contain structural information. Interestingly, the machine learning community has exhibited an increasing interest in learning the latent representations of vertices in networks, which is known as graph embedding (GE) [26]. A variety of methods [27 - 29] have been devised to represent networks with low-dimensional vectors that keep their main connective information. Through GE, the decision variables for network optimization problems can be transformed from discrete connection information into continuous variables. In this manner, existing surrogate techniques can be adopted to mitigate the difficulty caused by the high computational complexity in network robustness optimization. For example, a surrogate-assisted single-objective network optimization method was presented in [30]. However, this method [30] did not consider challenges resulting from the multi-objective robustness optimization of complex networks.

This paper is aimed at developing an efficient evolutionary algorithm for the multi-objective network robustness optimization problem, while addressing the associated high computational complexity and non-uniform latency. By combining the GE technique with surrogates, we propose a surrogate-assisted reference vector guided multi-objective evolutionary algorithm with a parallel evaluation strategy, termed SP-RV-MOEANet, to establish networks with a stronger tolerance against both node-based and link-based attacks. A network-parameter-assisted search procedure is devised in the parallel evaluation strategy; and the heterogeneous input is adopted in the constructed surrogate ensemble. Empirical results on several synthetic and realistic networks demonstrate that proposed algorithm can outperform the state-of-the-art method reported in [15], and in addition, reach a dramatic decrease (up to 50%) in runtime.

The rest of this paper is organized as follows. Related work on multi-objective network robustness optimization, non-uniform latencies, reference-vector-guided search, surrogate-assisted optimization, and graph embedding is reviewed in Section II. Section III presents the proposed non-uniform latency mitigation strategy and surrogate modeling. The details of the SP-RV-MOEANet are given in Section IV. Experimental results are reported in Section V. Finally, Section VI presents the concluding remarks.

# Related work

## Multi-objective Network Robustness Optimization

Different from numerical optimization problems [13, 14, 20, 22], decision variables for network optimization are composed of network data instead of numerical values. A network *G* can be represented as a matrix *G* = (**V**, **E**), where **V** = {1, 2,…, *N*} consists of *N* vertices and **E** = {*eij* | *i*, *j* ∈ **V**} consists of *M* edges. If there is a connection between vertices *i* and *j*, then *eij* is set as 1; otherwise, *eij* is 0. The first step in robustness optimization of networks is to define and evaluate their robustness. Generally, the definition of the robustness of a networked system is considered as the tolerance of this network against attacks and failures [1, 2]. The stability of systems is of great significance, and robust networks are expected to remain invulnerable. Given a specific network, its robustness can be estimated through different perspectives; for instance, changes on the degree distribution of a network reflect its functional loss under structural failures [1]. In general, the network topology has been shown to be closely related to its robustness [1, 2, 5], and it is important to evaluate the robustness when the network structure is not totally collapsed [2]. A measure *R* was proposed in [2] to numerically evaluate the network robustness through recording the structural integrity changes when losing connections. *R* indicates insensitivity of the network to different scales and properties, and has been proven to be reliable in network performance evaluation. The definition of *R* is given as follows:

 (1)

where *s*(*q*) is the fraction of the largest connected component in this damaged network after losing *q* vertices. Here, the removal can be either random or malicious. Random removal means that the removed nodes are selected from the network stochastically, and malicious removal is referred to removing the most important nodes in the current network. 1/*N* works as the normalization factor to make networks with different scales comparable. In this way, *R* gives a numerical estimation of the robustness of networks, and robust networks are likely to have large values of *R*.

*R* mainly focuses on the structural perturbations caused by node-based attacks. As indicated in [7], more damage types should be considered in applications. A measure *Rl* that can evaluate the network robustness against link-based attacks was designed as follows:

 (2)

where *s*(*p*) is the fraction of the largest connected component in the damaged network after losing *p* links. 1/*M* is the normalization factor here. Similar to *R*, *Rl* gives a numerical estimation of the robustness of networks against link-based attacks. A popular way to define the importance of links is based on the betweenness centrality, which has been examined in related studies [11, 15].

With the help of these measures, the optimization of network performance has been put a lot of value on in recent decades. In [2], a heuristic-based algorithm was devised to successfully enhance network robustness guided by *R*, and more competitive results can be obtained using the population-based method proposed in [9]. Meanwhile, as shown in [7], singly enhancing *R* cannot guarantee the promotion on *Rl*, and there exists a contradictory relation between the optimization on *R* and *Rl* [11]. Since it is hard to optimize *R* and *Rl* simultaneously, a better way to design such comprehensively robust networks is via multi-objective optimization methods as in [13, 14]. A two-phase multi-objective evolutionary algorithm was proposed in [15] to handle the MOP between *R* and *Rl*, and validated on networks with up to 500 nodes in the experiment. Inspiring results have been achieved in designing robust networks against multiple attacks, but several challenges remain to be tackled as indicated in the Introduction part.

## Non-uniform Latencies and Reference-Vector-Guided Search

Generally, it is assumed that different objectives have the same latency in solving MOPs; as a result, the number of fitness evaluations is the same for all objectives. However, as shown in [20], the runtime latencies for evaluating different objectives vary a lot in many real-life applications. To improve the computational efficiency, three schemes have been proposed to interleave the evaluations of different objectives [20]. Experimental results on several benchmark problems confirm a better efficiency of these schemes over the standard approach. In addition, a surrogate model was introduced to solve MOPs with non-uniform latencies [21], and a parallel strategy which combines optimization procedures and population generation processes has been designed to further improve the search performance. These studies reveal that delays may be caused by different durations of objective evaluations, and the mitigation of such delays can effectively improve the computational efficiency of multi-objective evolutionary algorithms (MOEAs).

Another issue that is directly related to the solution of MOPs is the quality of the obtained Pareto fronts, i.e., the non-dominated solutions. For a multi-objective optimization algorithm, the obtained non-dominated solutions, should have good performance in terms of both convergence and diversity, and the balance between these two performance indicators is crucial. In [22], a reference-vector-guided strategy was proposed to solve many-objective optimization problems. The strategy considers two factors in the environmental selection, including the angle between solution and its associated reference vector, and the distance from solution to the ideal point. Based on the angle and distance information, convergence is emphasized in the early stage of the search process, and the diversity is in priority in the later stage. In other words, solutions that are close to the ideal point are preferred first. As the search process proceeds, solutions with a large angle with the reference vector will be prioritized. To reduce the computational cost of reference-vector-guided search, surrogate models were introduced in [31]. Empirical results demonstrate that both convergence and diversity performance are competitive [22, 31].

## Surrogate-Assisted Optimization

Evolutionary algorithms (EAs) often need a large number of fitness evaluations to achieve good solutions. When EAs are applied to expensive optimization problems, the computational cost for fitness evaluations becomes non-trivial, preventing EAs from being applied to a wider range of real-world problems [32]. To address this challenge, much work has been reported to study the construction and management strategies of surrogate models to assist numerical optimization algorithms [21, 23, 24, 33]. Surrogate models have already contributed to the solution of many realistic optimization problems. For example, the design of trauma systems can benefit from a surrogate technique that adaptively selects a representative subset of the raw data [33], and as a result, the computational time can be significantly reduced.

A good surrogate is expected to guide the evolutionary search to find an acceptable solution at a reduced computational cost. For the numerical optimization problem, a variety of surrogate models have been proposed to approximate continuous objective or constraint functions. Gaussian processes [34], also known as Kriging, have shown to be effective in fitness approximation [21, 31, 33], partly because they are able to provide additional uncertainty information for surrogate management. However, Kriging will become computationally intensive when the number of training sample increases. Radial basis function (RBF) networks [35] are another popularly used surrogate model. RBF is relatively insensitive to high-dimensional data, and works well as a surrogate in assisting optimization of expensive problems [23 - 25]. Some other surrogates, including the inverse distance weighting (IDW) interpolation function [36] and the least square (LS) function [37], are also attractive options for surrogates.

## Graph Embedding

Networked systems have often hundreds or even thousands of nodes, making it hard to dig out useful information. Some structural properties, such as the degree distribution [5] and the number of shortest paths between nodes [3], are helpful to reflect the properties of a network. However, evaluation of the network structural properties is not an easy task, and huge computational resources are required, especially for those large-scale networks. A potential solution is to represent networks in a low-dimensional space in order to alleviate the high computational cost, and the graph embedding (GE) [26] has attracted increasing attention. Basically, GE intends to convert the discrete connection information of a node into numerical variables; in this way, a network with an *N* × *N* connection matrix can be represented as an *N* × *d* vector, and *d* tends to be much smaller than *N*. Several kinds of embedding methods have been proposed in recent studies. The random walk strategy [27], the autoencoder [28], and the matrix factorization [29] show feasibility in preserving network connection information in low-dimensional vectors. Based on GE, information excavation from networks can be more efficient with reduced runtime, which has been validated on tasks like community detection [28, 29], network reconstruction [28], and link prediction [27, 29].

In addition to the aforementioned information excavation tasks, GE shows potential to solve network optimization problems. Numerical-vector-based representations of networks can be obtained from GE; meanwhile, the robustness values of these networks can also be evaluated. In this way, surrogate models are available to learn the relationship between embedded graphs and their robustness values. A surrogate ensemble has been designed in [30], which remarkably improves the computational efficiency of single-objective network robustness optimization processes.

## New Contributions of the Proposed Algorithm

Focusing on the multi-objective network robustness optimization problem (guided by *R* or *Rl*), this work addresses three open challenges inspired by techniques in the context of numerical optimization problems.

Firstly, two optional parallel evaluation strategies, one population-based and the other individual-based, are designed based on the idea in [21], considering the distinct cost difference in the evaluation of *R* and *Rl* that may incur longer idle time and additional computational costs in the optimization process. The population-based strategy evaluates *R* or *Rl* of current individuals in parallel. The individual-based strategy completes the evaluation on the high-cost objective (*Rl*) and the search on the low-cost objective (*R*) of an individual in parallel. Here, the search is guided by a network structural parameter (*r*) for its positive correlation with *R* and low evaluation cost. The strategies are helpful to reduce the required running time. Also, the reference-vector-guided search is introduced into the optimization process, based on which, promising individuals are selected to exchange the structural information and generate new candidates. This operation contributes to the diversity of final results.

Secondly, surrogates are adopted to work as a low-cost performance estimator and provide the uncertainty information. By extending the idea in [30], which was designed for the single-objective (*R*) optimization of networks, an improved ensemble strategy is proposed. This strategy is aimed at estimating the robustness of networks and to guide the mutation operator. Compared with [30], the designed surrogate ensemble utilizes a heterogeneous input to reach a more reliable estimation of the two objectives, and intends to explore more promising candidates for the following local search operator at a low cost. The model management and update strategies are also tailored for this multi-objective optimization problem. Empirical results verify the effectiveness of the designed surrogate model in promoting both the convergence and the diversity of the solutions.

Last, a network structural parameter(*r*) is introduced into the initialization process. The previous study [15] manages to generate the initial population guided by one of the objectives, i.e., *R*, which is effective but requires a high computational cost. Considering *r* shows a positive correlation with *R* and negative with *Rl* [39], individuals with diverse performances on the two objectives can be generated at a low cost guided by this network parameter. The importance of this operator is validated in experiments.

Equipped with these strategies, the proposed algorithm significantly outperforms existing methods in solving the multi-objective network optimization problem.

# Parallel Fitness Evaluation and Approximation

In this work, we focus on efficiently multi-objective robust optimization of networks against multiple attacks. Before we describe the proposed algorithm SP-RV-MOEANet, two strategies to address two challenges, namely the non-uniform latencies between two robustness measures *R* and *Rl*, and the high computational cost required to evaluate the robustness values, are presented in this section. Based on several pilot studies, the first strategy intends to evaluate the two robustness measures in parallel to better utilize the computational resources; and the second strategy focuses on construction of a surrogate model to approximate robustness values of networks. Details of these strategies are depicted in the following text.

## A Parallel Evaluation Strategy for Network Optimization

In multi-objective robust optimization of networks, *R* and *Rl* can be evaluated separately, and evaluating *Rl* is usually more expensive. The state-of-the-art algorithm [15] does not consider the non-uniform latencies of the two objectives and just conducts the evaluation process in sequence, which is computationally inefficient [20, 21]. It has been reported that the parallel fitness evaluation is an effective strategy when solving numerical optimization problems with non-uniform latencies. As in [21], the expensive objective evaluation and search procedures were conducted in parallel, and better results on some benchmark test problems were achieved. Although this strategy can be applied to network robustness optimization, certain modifications are necessary to consider the features of network-related problems. First, owing to the large solution space when optimizing networks, random searches cannot effectively generate valid candidates [2]. Moreover, single-objective optimization guided by the relatively cheaper objective, *R*, is also time-consuming [9]. As a result, only a limited number of searches can be conducted in the process of evaluating *Rl* and the optimized networks tend to have an inferior performance of *R*. These features prevent a direct application of the parallelization strategy in [21] to the network robustness optimization problem in this work.

To evaluate the robustness measures *R* and *Rl* efficiently, both population- and individual-based parallelization strategies are considered. The population-based parallel robustness evaluation strategy is mainly implemented after conducting the *R*-guided local search on individuals in the population. Since the search is guided by *R*, the *Rl* values of all these newly generated individuals are evaluated in parallel. In the mutation operator, both robustness measures are evaluated in parallel. This strategy is denoted as *Parallel*-*Eva*. The individual-based parallelization strategy is implemented when *Rl* of a certain individual is being evaluated, and the search on *R* of another individual is conducted simultaneously. The search is guided by a network parameter assortativity (*r*) [4], instead of *R* itself. *r* is defined as follows:

 (3)

where *eij* is a link in **E**, *ki* is the degree of vertex *i* attached by *eij*, and the value of *r* is equal to the Pearson correlation coefficient of the degrees. *r* has a positive correlation with *R* as indicated in [39]. In other words, networks with a larger *r* are likely to be more robust in terms of *R*, and increasing *r* can be seen as an approximated optimization on *R*. Although the promotion effect may not be significant, more searches can be conducted because of the low computational cost for calculating *r*. For two input individuals *indi* and *indj*, the *Rl* value of *indi* is evaluated, and the *R* value of *indj* can be promoted parallelly. This strategy contains both search and evaluation processes, and is denoted as *Parallel*-*Sea*&*Eva*. Details of the strategy are given in Algorithm 1. The topological rewiring operator is shown in Algorithm 2.

|  |
| --- |
| **Algorithm 1: Parallel-Sea&Eva** |
| **Input**:  *Indi, Indj*: The original individuals;  *MaxSearch*:The maximum search step;  **Output**:  *Indi, Indj*: The operated individuals; |
| ***while*** *Rl* of *Indi* is evaluated***do***  Save the graph in *Indj* as *g*, and copy *g* in *gp*;  ***for*** *Re*=1 ***to*** *MaxSearch* ***do***  *gp* ← *Topological rewiring* (*g*, 1);  ***if*** *r*(*gp*) > *r*(*g*):  Update *g* with *gp*;  ***end if***;  ***end for***;  Save *g* in *Indj*, and evaluate *R* of *Indj*;  ***end while***; |

The determination of the parameter *MaxSearch* is usually problem specific, and a suitable value makes sure there is no time waste in *Parallel*-*Sea*&*Eva*. We recommend setting *MaxSearch* equal to *N*, which is the number of nodes in the network.

|  |
| --- |
| **Algorithm 2: Topological Rewiring Operator** |
| **Input**:  *G*: The input network;  *ReNum*: The rewiring attempts;  **Output**:  *G*’: The rewired network; |
| Copy *G* in *G*’;  ***for*** *j*=1 ***to*** *ReNum* ***do***  Select two links *ekl* and *emn* from *G*’ randomly, where *m*, *n* are different from *k*, *l*;  ***if*** *ekm* and *eln* do not exist in *G*’:  Remove *ekl* , *emn* from *G*’, and add *ekm*, *eln* to *G*’;  ***end if***;  ***end for***; |

The parallelization strategy in [21] aims at solving numerical optimization problems by conducting evaluation operations and random search processes simultaneously. But when solving the network robustness optimization problem, the random search guided by one robustness measure, which has a high computational cost, may be insignificant in the period for evaluating another measure. Such a strategy does not contribute notably to the design of robust networks. Considering features of this problem as depicted in the Introduction part, two modified strategies, *Parallel*-*Eva* and *Parallel*-*Sea*&*Eva*, are designed in this work to address this discrete optimization problem. These two strategies either concentrate on the direct evaluation task, or the evaluation mingled with the optimization task. The network property parameter *r* is adopted as the optimization indicator for its low computational cost and correlation with the measure *R*. The detailed utilization of these two strategies in SP-RV-MOEANet is described in the subsequent section.

## Surrogate for Network Robustness Evaluation

We intend to construct a surrogate model to estimate the robustness of candidate networks at a relatively low cost, and use the estimated robustness to guide part of the search procedures. The relationship between the decision variables and their corresponding fitness value is to be learned using the surrogate model. Here, fitness values can be evaluated by robustness measures shown in Eqns. (1) and (2); however, the original network data is discrete and of high dimensions. Thus, it is difficult to directly utilize network data to construct surrogates. GE has been shown to be effective in representing networks with low-dimensional vectors, and has been widely applied in network-related tasks [27 - 29]. The Structural Deep Network Embedding (SDNE) proposed in [28] is reliable on different kinds of networks, and has shown a good performance in maintaining the structural information to guide optimization processes [30]. SDNE is implemented in this work to represent networks, and these representations serve as the vicarious decision variables for networks.

Following the work in [30], we set the output dimension of SDNE as two, which means that each node in a network is represented by a two-dimensional vector, to balance the estimation accuracy with the computational cost. With the help of SDNE, the dimension of nodal representations can be considerably decreased. Nevertheless, the dimension is still high for a representation of the whole network. For instance, in the cased of a 500-node network, the original data of this network is of 500×500 dimensionality. After being processed by SDNE, each node can be presented by a two-dimension vector, and the whole network is still of 500×2 dimensionality. As indicated in [23, 24], optimization problems with over 1000 decision variables are considered to be of high-dimensional. With these network representations as decision variables, network performances such as the robustness can be estimated with a lower computational budget, and rational surrogate models have been proven to be effective as demonstrated in [30]. Considering the features of network representations, the surrogates can reasonably manage high-dimensional data. Since ensemble surrogates have shown excellence in solving expensive optimization problems [25], and can provide computationally more scalable uncertainty estimation than Gaussian processes [34], this work adopts a surrogate ensemble to estimate the robustness. Similar to [30], the ensemble in this work consists of three different base learners, namely an RBF [35], an IDW [36], and an LS [37].

Also, as heterogeneous input contributes to promoting the diversity of the ensemble, implicit features are also extracted from the raw data [25]. Based on embedded graphs, several feature extraction techniques have been implemented to extract features to be used as the input of surrogate models. The first one of such approach is the principal components analysis (PCA) [40] which can well extract the main components at a low computational cost. The second one is to cluster the embedding information, with the number of centers *k* is set as twenty percent of *N*. K-means is adopted for clustering. As shown in [27], the clustered embedding data, which contains part of the key connection information, is critical to mine the information from networks. The third one is to select embedding vectors of the top *k* high-degree nodes, with *k* being the number of centers used in K-means clustering. The robustness of a network is closely related to its high-degree nodes [8, 10], and the malfunctions of several hubs may cause a serious congestion in a networked system. The representations of such high-degree nodes can provide additional information to estimate the robustness of the network.

Given the network data, the robustness measures (*R* and *Rl*) of the networks are evaluated first, and networks are recoded by SDNE to get their 2D representations. Subsequently, three feature extraction techniques, along with the original embedded graph, are used as the input of the surrogate ensemble. The surrogate ensemble consists of RBF, IDW, and LS, which is able to estimate the robustness values and the uncertainty of the estimates, as shown in Fig. 1. The surrogate ensemble can later be used to facilitate the network robustness optimization.

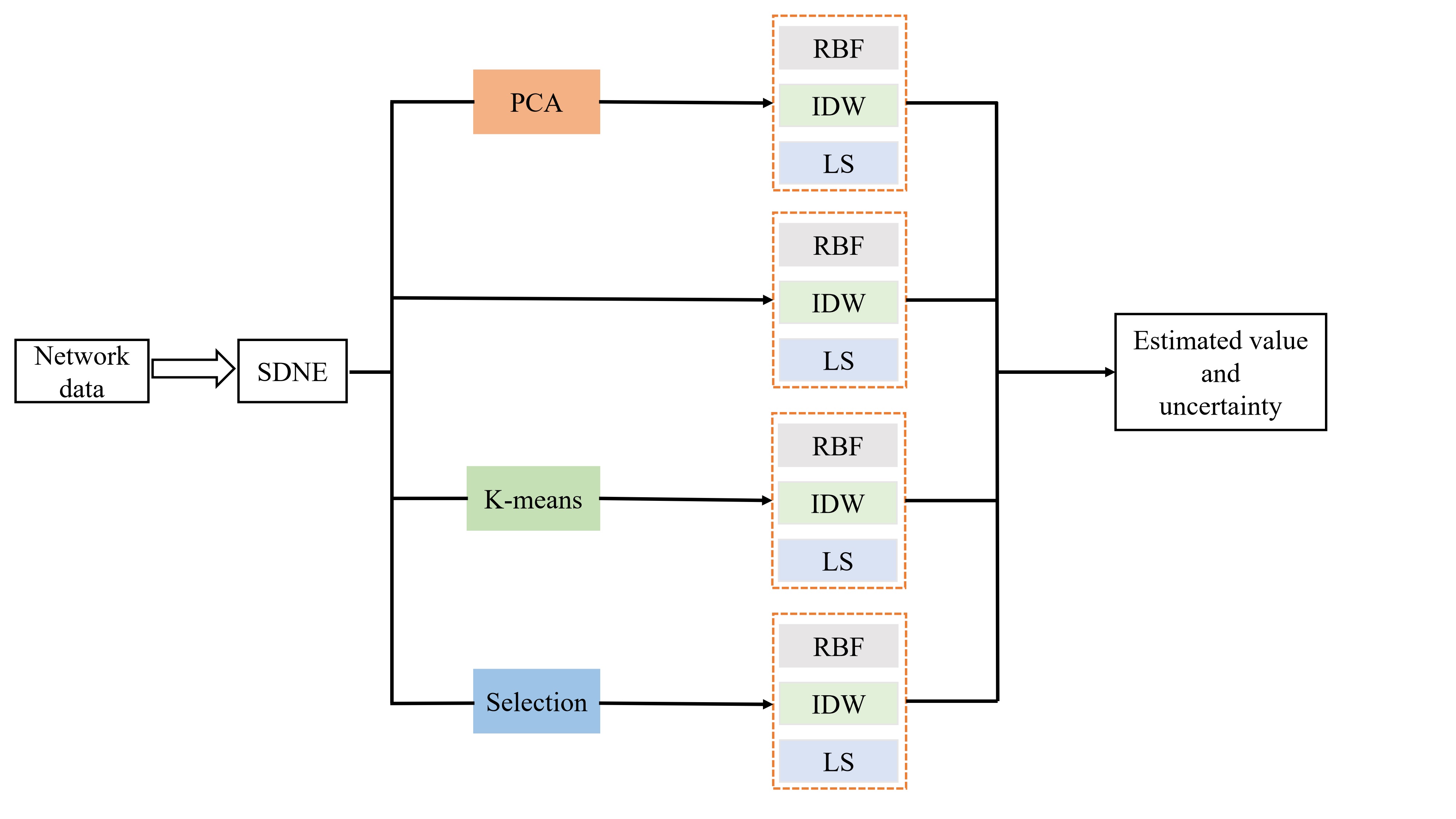


Fig. 1 The structure of the surrogate ensemble for estimating network robustness.

The related study in [30] concentrated on optimizing the measure *R* of networks, and the surrogate update mainly considered solutions with better or optimal *R* values. However, this study intends to optimize *R* and *Rl* simultaneously. The contradiction between these two objectives incurs difficulties in fitness predictions, and a number of representative candidates are expected to be considered in the update process. Therefore, the surrogate model is updated with obtained non-dominated solutions in each step of the generation process. In this way, surrogate can keep a closer track to solutions found in the search process.

# SP-RV-MOEANet

## The Framework

SP-RV-MOEANet intends to optimize both *R* and *Rl* of networks and outputs a set of non-dominated solutions. The algorithm begins with the initialization phase, in which the topological rewiring operation is implemented on the original network *G*0 to generate the initial population with a wide distribution of *R* and *Rl*. Then, the robustness values of the individuals in the initial population are evaluated using Eqns. (1) and (2) and the surrogate model is constructed using the networks in the initial population and associated robustness values as the training data. The trained surrogates are used to estimate the robustness measures in the following evolutionary search process. A set of unit reference vectors are determined for realizing the environmental selection. It should be noted that the degree distribution of all generated networks should be kept the same as that of *G*0.

|  |
| --- |
| **Algorithm 3: SP-RV-MOEANet** |
| **Input**:  *G*0: The input network;  *MaxGen*: The maximum number of genetic iterations;  **Output**:  ***EP***: The non-dominated solution set; |
| ***P***0 ← Initialization(*G*0);  Construct the surrogate model with ***P***0;  *t* = 0, ***EP*** = ∅;  ***while*** *t* < *MaxGen* ***do***:  Conduct the crossover operator on ***P****t* to generate ***Q****t*;  Conduct the mutation operator on ***Q****t*;  ***P****t*+1 ← Select better individuals from ***P****t* and ***Q****t*;  Conduct the local search operator on ***P****t*+1;  Update the surrogate model and reference vectors;  Update ***EP*** with ***P****t*+1;  *t* = *t* + 1;  ***end while***; |

Three genetic operators are employed in each generation: (1) a crossover operator to generate a temporary population through exchanging the information in part of the current population. The individuals to conduct this operator are selected considering the reference vectors; (2) a mutation operator to improve the quality of the temporary population through a structural search guided by either the network parameter *r* or the surrogate ensemble. The population is updated using the temporary population; (3) a local search operator to further improve *R* or *Rl* of some individuals selected from the population, with the selection guided by the reference vectors. All non-dominated solutions in the population are selected to update an external population ***EP***. ***EP*** is used to archive non-dominated solutions found in the search process, which present the output of the algorithm. The reference vectors and the surrogate model are updated in the genetic process. At a probability of *psu*, the surrogate model is updated using the solutions in ***EP***. Meanwhile, the reference vectors are adjusted every 20 generations following the recommendation in [22, 31]. The pseudocode of SP-RV-MOEANet is described in Algorithm 3. The details of the population initialization and genetic operators are presented in the following subsections. A comparison between the current state-of-the-art method MOEA-RSFMMA [15] and SP-RV-MOEANet can be found in Supplementary Materials I.

## Population Initialization

Population initialization aims at generating a set of initial networks for search processes, and therefore, a well-distributed initial population is essential. As the solution space of multi-objective network robustness optimization is extremely large, a randomly generated initial population may not be efficient, as shown in [15, 18]. An initialization strategy was devised in [15] to design robust networks against multiple attacks, which focused on adjusting *R* to get networks with both high and low *R*. Although this strategy works well for generating the initial population, the computational cost was high [15]. Even if the computational complexity of evaluating *R* is cheaper than that of *Rl*, a large runtime is still needed for generating the initial population in [15].

In this work, we manage to generate a promising initial population efficiently. As shown in [39], the assortativity (*r*) defined in Eqn. (3), which is computationally cheap, is related to the network robustness. Networks with a high value of *r* are more robust to node-based attacks, but fragile against link-based attacks based on the betweenness centrality. Thus, networks with both high and low *r* values can be satisfactory initial solutions to optimize *R* and *Rl*. To achieve this, the initialization operator contains the following operations.

1) The operator searches for structurally different *G*0 by minimizing *r*, and obtains *Gmin* with the minimal *r*.

2) Then the operator keeps increasing *r* of *Gmin* using the topological rewiring operator until reaching the maximum value; in this process, a candidate network *Gc* is preserved when a certain number of searches (e.g. one hundred) is carried out.

3) To enhance the diversity of the initial population, several networks (*Numrand*) are also generated based on random topological rewiring on *G*0. In this work, *Numrand* is set as ten.

4) A series of networks are now obtained, and their robustness values *R* and *Rl* are evaluated using the *Parallel*-*Eva* strategy. The better individuals are selected for the initialization population ***P***0 until *Initsize* individuals are generated. Based on ***P***0, the surrogate ensemble is trained to evaluate *R* and *Rl*.

## Crossover and Mutation Operators

The crossover operator is designed to exchange part of the chromosome information of the individuals in the current population, and several new individuals are generated before the mutation operations are performed. A dedicated crossover operator was designed in [9, 15]. In this work, we intend to select individuals that can further improve the effectiveness of the crossover operator. As shown in [22, 31], the reference vectors can guide the selection of individuals for crossover. The basic idea is that individuals close to the reference vectors are preferred in the early stage to accelerate the convergence of the search process, and others can be selected in the later stage to promote the diversity of the population. The distance between an individual and its associated reference vector is evaluated using the angle-penalized distance (APD) defined as follows [22, 31]:

 (4)

where *P*(*θ j*) is the angle penalty function that balances convergence and diversity, ||*f j*|| is the normalized function value. A detailed discussion of APD can be found in [22, 31].

With the help of the reference vectors, the crossover operator can exchange structural information between the selected individuals to generate more promising candidate networks at a possibility of *pc*. The detailed crossover operation between selected individuals (networks) follows the one proposed in [9, 15]. The extra population ***Q****t* is generated through the crossover operator, whose size, *Extrasize*, is a predefined parameter. For each individual in ***Q****t*, if the random number fits *pc*, the corresponding network is constructed by performing crossover between the two selected candidates in the current population. Candidates are determined based on the roulette wheel selection on their APD values. Otherwise, the new individual duplicates one existing candidate randomly.

Subsequently, the mutation operator is conducted on ***Q****t* using the surrogates. The mutation process involves three stages, as follows.

1) The first stage is to randomly rewire topologies of all individuals in ***Q****t* at a probability of *pm*.

2) The second stage begins with utilizing the surrogates to evaluate *R* and *Rl* of all individuals in ***Q****t* and certain representative individuals are selected to implement topological changes guided by *r*. The selection is based on clustering, and a number of *Numclu* cluster centers are determined by the estimated robustness values of all individuals in ***Q****t*. For each center, the individual that is closest to the center is chosen as a representative. Several other individuals (a total of *Numoth*) are selected as additional representatives based on their uncertainty information and distance to the existing representatives. The uncertainty information is provided by the surrogate ensemble, and the distance is evaluated by the Euclidean distance between the embedded vectors of the individual and the determined centers. Individuals with a larger degree of uncertainty are likely not considered in the previous searches and may become non-dominated solutions. The individuals with a larger distance to the existing representatives are likely to contribute to a diverse population. The fitness-proportional selection is adopted to select additional individuals. Some individuals are selected more than once in the abovementioned selection operation, and they will be are preserved in a temporary set ***P****mul*. For the selected representatives, *r* is increased and decreased in the generation iterations with even and odd numbers to make structural changes on these individuals.

3) The third stage is to rewire networks guided by the robustness values estimated by surrogates on individuals in ***P****mul*. The individual with the smallest uncertainty information in ***P****mul* is selected, and the rewiring process is guided by the estimated robustness value of either *R* or *Rl* in the even and odd generations.

In the end of the mutation operator, individuals in ***Q****t* are evaluated using the *Parallel*-*Eva* strategy. The procedure of the mutation operator is summarized in Algorithm 4. *Numclu* and *Numoth* are set as 3 and 2, respectively. *MaxSea* is set as 50 in the experiment.

|  |
| --- |
| **Algorithm 4: Mutation operator** |
| **Input**:  ***Q****t*: The external population (archive);  *pm*: Probability to conduct the random mutation operator;  *MaxSea*: The maximum number of rewiring operations;  *Numclu*: The number of cluster centers;  *Numoth*: The number of other representatives;  *gen*: Current generation iteration number; |
| /\*The first stage\*/  ***for*** each individual in |***Q****t*|:  ***if*** *rand*(0, 1) < *pm* **do**:  Conduct *Topological rewiring* (*g*, *MaxSea*) where *g* is the network of this individual;  ***end if***;  ***end for***;  /\*The second stage\*/  ***Q****rep* ← ∅;  Utilize surrogates to estimate *R* and *Rl* values of individuals in ***Q****t*, then determine *Numclu* representatives and save in ***Q****rep*;  Determine other *Numoth* representatives and save in ***Q****rep*, and get ***P****mul*;  ***for*** each individual in ***Q****rep* ***do***:  Increase (decrease) *r* of the network *g* of this individual using *Topological rewiring* (*g*, *MaxSea*) when *gen* is even (odd);  ***end for***;  /\*The third stage\*/  ***if*** ***P****mul* ≠ ∅ ***do***:  Find the individual in ***P****mul* with the smallest value of uncertainty information, save its network in *g*;  Conduct optimization using *Topological rewiring* (*g*, *MaxSea*) guided by its estimated *R* (*Rl*) when *gen* is even (odd);  ***end if***;  Update the corresponding individuals in ***Q****t*;  Evaluate *R* and *Rl* of individuals in ***Q****t* using *Parallel*-*Eva*; |

## Local Search Operator

The local search operator intends to enhance *R* or *Rl* of the individuals without using the surrogates, which is a time-consuming but critical process to enhance the quality of the obtained solutions. Therefore, this operator contains two separate sets of individuals selected from the current population to enhance *R* (***L***-*R*) and *Rl* (***L***-*Rl*). The selection follows the roulette wheel strategy (fitness-proportional) based on the angle values, i.e., the angle between an individual and the associated reference vector.

To enhance *R*, two strategies can be used. The first strategy is to enhance *R* of each individual in the set, and later evaluate their *Rl* values using *Parallel*-*Eva*. The second strategy utilizes *Parallel*-*Sea*&*Eva* iteratively to evaluate *Rl* and improve *R* simultaneously. To enhance *Rl*, two strategies are considered as well. Since the computational cost for directly evaluating *Rl* is extremely high, in the first strategy, the operator decreases *R* of the individuals to approximately improve *Rl*, then *Rl* values are evaluated using *Parallel*-*Eva*; the second strategy focuses on the individual with the largest *Rl* value in the set. In this way, both *R* and *Rl* are improved to accelerate the convergence of the search process. The pseudocode of this local search operator is given in Algorithm 5.

|  |
| --- |
| **Algorithm 5: Local search operator** |
| **Input**:  ***P****t*: The current population;  *LocNum*: The number of individuals to conduct the operator;  *MaxLocalSea*: The maximum number of searches;  *pl*: The possibility of conducting local search operations;  *ps*: The possibility of selecting strategies; |
| ***L***-*R*, ***L***-*Rl* ← ∅;  Select *LocNum* individuals into ***L***-*R* and ***L***-*Rl* separately;  ***if*** *rand*(0, 1) < *pl* and *rand*(0, 1) < *ps*:  Use *Topological rewiring* (*g*, *MaxLocalSea*) on each *g* in ***L***-*R* to optimize *R*;  Evaluate *Rl* of individuals in ***L***-*R* using *Parallel*-*Eva*;  ***else if*** *rand*(0, 1) < *pl* but *rand*(0, 1) < *ps*:  Use *Parallel*-*Sea*&*Eva* on ***L***-*R*;  ***end if***;  ***if*** *rand*(0, 1) < *pl* and *rand*(0, 1) < *ps*:  Use *Topological rewiring* (*g*, *MaxLocalSea*) on each *g* in ***L***-*Rl* to decrease *R*;  Evaluate *Rl* of individuals in ***L***-*Rl* using *Parallel*-*Eva*;  ***else if*** *rand*(0, 1) < *pl* but *rand*(0, 1) < *ps*:  Find *g* with the largest value of *Rl* in ***L***-*Rl*, use *Topological rewiring* (*g*, *MaxLocalSea*) to optimize *Rl*, and get its *R*;  ***end if***;  Update the corresponding individuals in ***P****t* with ***L***-*R* and ***L***-*Rl*; |

# Experimental Results

This section verifies the performance of SP-RV-MOEANet on several synthetic and real-world networks. In the experiments, *Initsize* is set as 20, *Extrasize* as 15, *MaxGen* as 200, *LocNum* as 5, *MaxLocalSea* as 50, *pc* as 0.6, *pm* as 0.5, *pl* as 0.7, *ps* as 0.8, and *psu* as 0.7. The existing method to enhance the network robustness against multiple attacks, MOEA-RSFMMA [15], is implemented for comparisons. The parameters of MOEA-RSFMMA follow the setting in [15].

## Experiments on synthetic networks

Several different kinds of synthetic networks have been generated to test the performance of algorithms, including scale-free (SF) networks generated by the BA model [5], small-world (SW) networks defined in [41], and random networks based on Erdős-Rényi (ER) model [42]. In addition to MOEA-RSFMMA, the proposed algorithm, SP-RV-MOEANet, can be degenerated into certain variants for comparison. The basic variant without using the surrogate, parallel strategy, reference vectors, and population initialization strategy is denoted as MOEA0; and the variant that involves with population initialization strategy and parallel strategy is denoted as P-MOEANet; and the variant with reference vectors is denoted as P-RV-MOEANet. The obtained non-dominated solutions of these methods on three kinds of synthetic networks with *N* = 300 and averaged degree 〈*k*〉 = 4 are shown in Fig. 2. Results in terms of the hyper volume (HV) values of the obtained non-dominated set and their corresponding runtime are given in Table I. Fig. 3 plots the non-dominated solutions obtained by the algorithms under comparison, from which we can see that the performance of the algorithms varies a lot. Here, the basic algorithm MOEA0 serves as the baseline for comparison.

Fig. 2 indicates that the results obtained by MOEA0 are unsatisfactory. The state-of-the-art algorithm MOEA-RSFMMA performs similarly to P-MOEANet, which indicates that the parallel evaluation strategy might not contribute substantially to the search capability. But in terms of the computational cost, P-MOEANet consumes a considerably smaller computation time than MOEA-RSMMMA, as listed in Table I. The results demonstrate that the designed parallel strategy can mitigate the side effect of non-uniform latencies between *R* and *Rl*. Furthermore, P-RV-MOEANet improves the diversity of Pareto fronts to a certain extent using reference vectors, which validates the effectiveness of reference-vector-guided search techniques [22, 31] in solving network optimization problems. Finally, the proposed SP-RV-MOEANet outperforms all other tested methods, and competitive results are attained with a good balance between exploration and exploitation. Compared with P-RV-MOEANet, SP-RV-MOEANet requires a larger runtime, primarily because of the construction of the surrogate model.

TABLE I

Experimental results on synthetic networks with 300 nodes. The Wilcoxon rank sum tests with a significance level of *p* = 0.05 are adopted to analyze the difference of these results compared with SP-RV-MOEANet. “-” means the algorithm is inferior to SP-RV-MOEANet, and “+” means the algorithm outperforms SP-RV-MEOANet. The results are averaged over 10 independent runs are given together the standard deviations are provided.

|  |  |  |  |
| --- | --- | --- | --- |
| Network | Algorithm | Averaged HV | Running time (hours) |
| SF | MOEA-RSFMMA | 0.0792±0.0021(-) | 121.1 |
| MOEA0 | 0.0580±0.0019(-) | 111.7 |
| P-MOEANet | 0.0790±0.0025(-) | 67.5 |
| P-RV-MOEANet | 0.0799±0.0018(-) | 68.7 |
| SP-RV-MOEANet | **0.0836**±0.0020 | 69.2 |
| ER | MOEA-RSFMMA | 0.0887±0.0022(-) | 115.6 |
| MOEA0 | 0.0523±0.0025(-) | 110.8 |
| P-MOEANet | 0.0886±0.0017(-) | 63.3 |
| P-RV-MOEANet | 0.0887±0.0023(-) | 64.2 |
| SP-RV-MOEANet | **0.0903**±0.0019 | 65.5 |
| SW | MOEA-RSFMMA | 0.1042±0.0042(-) | 112.8 |
| MOEA0 | 0.0931±0.0041(-) | 106.1 |
| P-MOEANet | 0.1044±0.0045(-) | 58.1 |
| P-RV-MOEANet | 0.1045±0.0039(-) | 59.4 |
| SP-RV-MOEANet | **0.1083**±0.0043 | 60.4 |

Nevertheless, compared with the current state-of-the-art method, SP-RV-MOEANet exhibits better computational efficiency. The cost of training surrogates is worthy for solving the problem. On the one hand, these surrogates work as the low-cost fitness estimator to guide part of the search process, and more potential candidates can be generated. On the other hand, the uncertainty information maintained by the surrogate ensemble provides additional criteria to select individuals, which is lack in methods without surrogates. The results in Fig. 2 and Table I validate the marked performance of SP-RV-MOEANet in designing robust networks against multiple attacks, and the algorithm can deal with different kinds of synthetic networks.

Changes of the HV values over the generations of five compared algorithms are given in Fig. 3. For MOEA0, the HV value of obtained results increases only slightly, and the initial HV value is considerably smaller than that of other algorithms. These results demonstrate the effectiveness of the proposed network-related operators in solving the multi-objective network robustness optimization problem. The two algorithms using reference vectors for selection, namely, SP-RV-MOEANet and P-RV-MOEANet, exhibit a comparable performance as that of other algorithms (except for MOEA0) in the early search stage; however, as the search process proceeds, the two algorithms gradually outperform the others. Guided by the reference vectors, the exploitation is emphasized in the early stage, and the exploration takes the priority afterwards. Such a mechanism impacts the selection of individuals to conduct genetic operators, and may result in a relatively inferior initial HV performance. The HV profile of MOEA-RSFMMA is similar to that of P-MOEANet, as shown in Fig. 3. A few additional analyses on the random removal betweenness-based nodal attacks are described in Supplementary Materials II.

|  |  |  |
| --- | --- | --- |
|  | | |
|  |  |  |
| (a) SF | (b) ER | (c) SW |

Fig. 2. The non-dominated solutions obtained by the algorithms on three synthetic networks with 300 nodes.



|  |  |  |
| --- | --- | --- |
|  |  |  |
| (a) SF | (b) ER | (c) SW |

Fig. 3. The change of HV values over generations. The results are averaged over five independent runs.

It can also be seen that SF networks correspond to the most notable improvement among the three kinds of synthetic networks. In the evaluation of *R* and *Rl*, malicious attacks have been applied, which delete the most important component (node or link) in the current network. With a power-law degree distribution, only few nodes in SF networks possess a large number of connections between neighbors, and this scenario is commonly observed in many realistic networked systems [4]. This unique structure makes SF networks “robust yet fragile” [43]; that is to say, SF networks tend to have a good tolerance against random attacks but can be extremely vulnerable to malicious attacks. The results in Figs. 2 and 3 reveal that the fragility of SF networks can be considerably mitigated through proper rewiring operations. Considering the generality of SF networks, the proposed algorithm can solve optimization problems pertaining to real networked systems, as described in the following subsection. The previous study in [15] did not touch the optimization on ER and SW networks. The performed experiments demonstrate the feasibility of existing optimization algorithms in enhancing the performance on these networks. The performance analysis of representative solutions can be found in Supplementary Materials III.

The changes of HV in the evaluation of real robustness measures are examined, and an example of SF networks is given in Fig. 4. The results of the method adopting the original parallel evaluation strategy in [21] are labelled as “P0-MOEANet”. The non-uniform latency between *R* and *Rl* is presented through different evaluation costs. The evaluation budget on *R* and *Rl* is considered as 1 and 5, respectively. A maximal budget of 10000 is implemented in the experiment. First, in terms of the parallel strategy, P0-MOEANet is inferior to P-MOEANet, which demonstrates the efficacy of the designed parallel strategy in Algorithm 2. The computational budget for *R* is less than that of *Rl*, but *R* remains a high-cost measure as shown in [30]. The strategy in [21] takes *R* directly as the guidance while evaluating *Rl*, and only a few searches to improve *R* can be completed during the process. Such strategy may not guarantee a good performance. In contrast, the designed strategy takes a low-cost network property parameter *r* as the search guidance while evaluating *Rl*. Consequently, more thorough searches can be performed on *R* and better results can be obtained. Meanwhile, as shown in the figure, SP-RV-MOEANet and P-RV-MOEANet are more efficient in utilizing computational resources over other compared algorithms, and better non-dominated solutions can be attained under a given budget. These results also demonstrate the competitive performance of the proposed algorithm.

|  |
| --- |
|  |
|  |

Fig. 4 Changes of the HV values over the number of evaluations on SF networks. The results are averaged over five independent realizations.

Furthermore, experiments on SF networks with 500 and 1000 nodes and 〈*k*〉 = 4 are conducted to examine the scalability of SP-RV-MOEANet on large-scale networks. MOEA-RSFMMA is also included in the comparative analysis. The HV values of the obtained non-dominated solution sets and the required runtime are listed in Table II. The results indicate that SP-RV-MOEANet remains effective on large-scale networks and achieves considerably better results in a given period of time. MOEA-RSFMMA, which is the state-of-the-art, is inferior to the proposed algorithm, and consumes larger amount of computational resources. In [15], MOEA-RSFMMA was tested on networks with up to 500 nodes, and its prohibitive computational cost prevents it from being applied to the optimization of larger networks. Even months of running time is necessary to optimize networks with 1000 nodes mainly due to the frequent invoking of high-cost robustness measures. In this work, several strategies have been designed to improve the computational efficiency of the optimization process, and the proposed SP-RV-MOEANet algorithm can handle networks with 1000 nodes with a dramatically reduced computational budget compared with the existing method.

TABLE II

Experiments on SF networks with 500 and 1000 nodes. The results of statistical tests are also given. Results of SF-500 are averaged over 5 realizations, and those of SF-1000 are over 3 realizations.

|  |  |  |  |
| --- | --- | --- | --- |
| Network | Algorithm | Averaged HV | Running time (hours) |
| SF-500 | MOEA-RSFMMA | 0.0747±0.0025(-) | 182.8 |
| SP-RV-MOEANet | 0.0781±0.0024 | 95.2 |
| SF-1000 | MOEA-RSFMMA | 0.0638±0.0019(-) | 1047.5 |
| SP-RV-MOEANet | 0.0689±0.0023 | 552.1 |

## Experiments on real-world networks

The effectiveness of SP-RV-MOEANet is validated on two real-world networks, namely the USA airline network (UAN) [44] consisting of 332 nodes and 2126 links and the India power grid (IPG) [45] composed of 572 nodes and 871 links. Experimental results on these two networks are presented in Fig. 5. The HV values and runtime are given in Table III.

As shown in Fig. 5, both *R* and *Rl* of the two networks can be enhanced, and a dramatic improvement is attained in the IPG case. The effectiveness of topological rewiring process is verified. The UAN network has a large link density compared with that of other tested networks; therefore, the evaluation of its robustness *Rl* is extremely time-consuming, leading to a costly optimization processes as shown in Table III. Also, thanks to the extraordinary link density, the initialization operator can generate considerable candidates for the algorithm. All the tested algorithms achieve satisfactory results on UAN, except for MOEA0 which does not maintain the initial operation. Nevertheless, SP-RV-MOEANet slightly outperforms other algorithms at a reduced computation, including the state-of-the-art method reported in [15]. The IPG network has a relatively smaller link density, and SP-RV-MOEANet remarkably outperforms other algorithms. In terms of the obtained non-dominated solutions, both diversity and convergence can be significantly improved at a reduced cost. These results indicate that SP-RV-MOEANet is effective in solving multi-objective robustness optimization on real networks as well.

|  |  |
| --- | --- |
|  |  |
| (a) UAN | (b) IPG |

Fig. 5. The obtained non-dominated solutions of different methods on two real-world networks. Based on the required cost defined in Eqn. (5), the possible preferred solutions for each network are marked in green.

Furthermore, the structural cost required by the optimization process is analyzed through changes in the distance information between nodes in the networks. The distance between nodes (*D*) in a network is defined as,

 (5)

where *d*(*m*, *n*) is the shortest path from *m* to *n*, and **V** is the set of all nodes in this network. The *D* values of UAN and IPG are evaluated first as the baseline, and the corresponding values of the optimized networks are also obtained. A smaller distance variance indicates a milder change on the network structure, which requires a smaller cost [2]. In this manner, numerical results on the required cost for each non-dominated solution in Fig. 5 can be evaluated; based on which, low-cost solutions can be detected. A qualitative analysis on the cost is given in Fig. S4 of Supplementary Materials. Solutions that require the minimum cost may be more valuable for decision makers, and the corresponding solutions are marked in Fig. 5. As it can be seen, solutions located in the middle of Pareto front with a balanced performance between *R* and *Rl* are likely to be chosen as the final output of the optimization process.

TABLE III

Experimental results on two real-world networks. The results are averaged over 10 independent realizations.

|  |  |  |  |
| --- | --- | --- | --- |
| Network | Algorithm | Averaged HV | Running time (hours) |
| UAN | MOEA-RSFMMA | 0.0790±0.0019(≈) | 290.5 |
| MOEA0 | 0.0567±0.0023(-) | 277.8 |
| P-MOEANet | 0.0791±0.0015(≈) | 175.9 |
| P-RV-MOEANet | 0.0789±0.0021(-) | 177.1 |
| SP-RV-MOEANet | **0.0794**±0.0028 | 178.4 |
| IPG | MOEA-RSFMMA | 0.04636±0.0015(-) | 196.1 |
| MOEA0 | 0.03700±0.0009(-) | 181.9 |
| P-MOEANet | 0.04690±0.0018(-) | 101.2 |
| P-RV-MOEANet | 0.05018±0.0019(-) | 102.5 |
| SP-RV-MOEANet | **0.05199**±0.0019 | 103.1 |

Some intuitive analyses on the structure of optimized networks are presented in Supplementary Materials III. The design of robust infrastructures is of significance for networked systems. A number of attacks or errors may threaten normal functioning of systems, including intentional attacks, natural disasters, and systematic failures. In the context of aforementioned networks, damages may aim at nodes in the system, i.e., airports in UAN and power stations in IPG. Meanwhile, links are also vulnerable, and air routes and transmission lines are at risk. Related studies [7, 11, 15] indicated that optimization methods considering multiple damage scenarios can help enhance the overall performance of network systems. Diversified solutions obtained using an MOEA can be presented to the user to handle different circumstances, and such techniques are more effective than single-objective optimization methods like in [2, 9, 10]. With a notably reduced running time and an improved performance, the proposed algorithm is more promising to solve optimization dilemmas in real systems over existing methods.

# Conclusions

In this paper, we focus on designing robust networks against multiple attacks. Several challenges critical for the performance of the optimization have been addressed. In this context, we propose a new optimization algorithm that adopts the parallel fitness evaluation strategy and a set of reference vectors to optimize complex networks. Furthermore, a graph embedding based surrogate model is constructed to serve as a low-cost robustness estimator and provides uncertainty information for selection. These strategies are integrated into a multi-objective optimization framework named SP-RV-MOEANet. The proposed algorithm is validated on both synthetic and real-life complex networks and shown to outperform the state-of-the-art algorithm in [15]. Another advantage of the proposed algorithm is its low computation complexity compared to that of existing algorithms, which make it practical to be applied to solve large-scale network optimization problems.

Our contributions can be summarized into two parts. First, in the complex network field, the network performance enhancement has been a key research topic in recent studies, and noted to be a significant aspect in a wide range of applications. However, the existing solutions for the network robustness optimization problems, especially those with multi-objectives, always require prohibitive computational resources, and the quality of obtained results tends to be unsatisfactory. In this work, a computationally efficient algorithm is developed considering certain ingenious strategies for other optimization problems. Corresponding modifications or improvements are implemented to cater to the unique features of network data. The empirical results indicate the outstanding efficiency of the proposed algorithm when designing robust network against multiple attacks. This algorithm can work as a basic framework to solve other optimization problems related to network data.

Moreover, in the computational intelligence field, strategies like parallel evaluation and surrogates have been intensively emphasized. The experimental results validate their surprising effect in solving optimization tasks. Nonetheless, most of the existing studies concentrated on benchmark-like problems with numerical decision variables. The application on problems with other data shapes is still lacked. In this work, we focus on complex networks which are not formed by numerical variables, but by highly discrete data and with strict structural constraints. Mechanisms of reference-vector-guided search, surrogates, and parallel evaluation are introduced into the robustness optimization process of networks, and encouraging improvements can be noticed. This tentative study reveals the significance of computational intelligent methods on network-based optimization problems.

This work provides implications to solving network-related and other high-cost discrete optimization problems. For example, a combination of the performance estimator and the true performance measure can significantly improve the computational efficiency of the algorithm and further boost the optimization process. The proposed algorithm shows a uniform framework of this method, and it is not hard for other researchers to extend the algorithm from other optimization problems such as structural fragility [45], community robustness [46] and controllability [47]. A wide range of real-world applications can possibly benefit from this work, especially for complex optimization tasks such as the design of infrastructures (e.g. the water supply system [16] and the power grid [17]), the organization of manpower resources [48], and the maintenance of cooperative behaviors [49]. In addition to networks, the solution of other discrete optimization problems may get implications from this work too. Techniques, like surrogates and reference vector guided multi-objective search, that can successfully fulfil non-convex numerical optimization tasks [13, 14, 22, 24] are also valuable and considerable when designing algorithms. Proper modifications and applications of such techniques can promote the search ability and achieve competitive results. Furthermore, several pilot studies [20, 21] indicated the necessity of considering the latency difference between optimization objectives, but the application is mostly limited to benchmark test problems. This work presents a case study on the multi-objective network robustness optimization, in which one objective (*Rl*) requires considerably higher evaluation cost than the other one (*R*). Parallel evaluation strategies are developed according to the features of this problem. The experimental results validate that these strategies can greatly improve the performance of the algorithm, thereby revealing the significance of theories and methods in [20, 21].

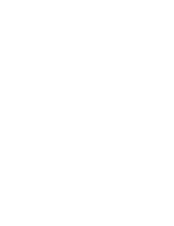
The robustness optimization of networked systems remains a challenging topic, particularly when the size of the networks becomes even much larger. Several knotty problems remain to be investigated. First, a relatively high cost is always inevitable when assessing the performance of networks as indicated in the experimental results. This issue causes difficulty in evaluating and optimizing networked systems. Therefore, methods that can evaluate networks more efficiently are in high demand. The network structural parameters [39] and the graph embedding [27] may become potential alternative approaches. For example, the performance of a specific network can be roughly estimated via several structural features or the numerical presentation of its topology. Note that this type of performance estimation is problem specific, and the structural parameter and embedding methods should be selected considering the functionality or characteristics of the network. More validations and analyses are to be achieved in the future. Also, the majority of existing studies focused on static networks, but a large number of real systems are dynamic. The structure and the functionality of a network may change over time. Considering this complex feature, how to utilize information maintained in the time series data is a key but open question. Based on the proposed work, we can predict the future developing trend of a dynamic network and evaluate its overall performance. In addition to the performance enhancement of networks with only a single layer, the enhancement of systems with multiple functional layers [8, 46] is also required. In general, different layers are relatively independent but correlated with each other, and all layers are crucial to the normal functioning of the whole systems. This feature incurs complexity in the corresponding evaluation and optimization process, and effective solutions are needed to be addressed and worthy of further study.

References

1. R. Albert, H. Jeong, and A. L. Barabási, “Error and attack tolerance of complex networks,” *Nature*,vol. 406, pp. 378-382, 2000.
2. C. M. Schneider, A. A. Moreira, J. S. Andrade, S. Havlin, and H. J. Herrmann, “Mitigation of malicious attacks on networks,” *PNAS*, vol. 108, no. 10, pp. 3838–3841, 2011.
3. M. Girvan and M. E. J. Newman, “Community structure in social and biological networks,” *PNAS*, vol. 99, no. 12, pp. 7821-7826, 2002.
4. M. E. J. Newman, Networks: An Introduction, *Oxford University Press*, Oxford, 2010.
5. A. L. Barabási and R. Albert, “Emergence of scaling in random networks,” *Science*, vol. 286, no.5439, pp.509-512, 1999.
6. M. E. J. Newman, “Assortative mixing in networks,” *Physical Review Letters*, vol. 89, no. 20, p. 208701, 2002.
7. A. Zeng and W. Liu, “Enhancing network robustness against malicious attacks,” *Physical Review E*, vol. 85, p. 066130, 2012.
8. S. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin, “Catastrophic cascade of failures in interdependent networks,” *Nature*, vol. 464, pp. 1025-1028, 2010.
9. M. Zhou and J. Liu, “A memetic algorithm for enhancing the robustness of scale-free networks against malicious attacks,” *Physica A*, vol. 410, pp. 131-143, 2014.
10. X. Tang, J. Liu, and X. Hao, “Mitigate cascading failures on networks using a memetic algorithm,” *Scientific Reports*, vol. 6, p. 38713, 2016.
11. J. Liu, M. Zhou, S. Wang, and P. Liu, “A comparative study of network robustness measures,” *Frontiers of Computer Science*, vol. 11, no. 4, pp. 568-584, 2017.
12. S. Wang and J. Liu, “Designing comprehensively robust networks against intentional attacks and cascading failures,” *Information Sciences*, vol. 478, pp. 125-140, 2019.
13. Q. Zhang and H. Li, “MOEA/D: A multiobjective evolutionary algorithm based on decomposition,” *IEEE Trans. on Evolutionary Computation*, vol. 11, no. 6, pp. 712-731, 2007.
14. K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Trans. on Evolutionary Computation*, vol. 6, no. 2pp. 182-197, 2002.
15. M. Zhou and J. Liu, “A two-phase multi-objective evolutionary algorithm for enhancing the robustness of scale-free networks against multiple malicious attacks,” *IEEE Trans. on Cybernetics*, vol. 47, no. 2, pp. 539-552, 2017.
16. H. S. Jeong, J. Qiao, D. M. Abraham, M. Lawley, J. Richard, and Y. Yih, “Minimizing the consequences of intentional attack on water infrastructure,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 21, pp. 79-92, 2006.
17. W. Zhang, W. Pei, and T. Guo, “An efficient method of robustness analysis for power grid under cascading failure,” *Safety Science*, vol. 64, pp. 121-126, 2014.
18. S. Wang and J. Liu, “A multi-objective evolutionary algorithm for promoting the emergence of cooperation and controllable robustness on directed networks,” *IEEE Trans. on Network Science and Engineering*, vol. 5, no. 2, pp. 92-100, 2018.
19. S. Wang and J. Liu, “Constructing robust cooperative networks using a multi-objective evolutionary algorithm,” *Scientific Reports*, vol. 7, p. 41600, 2017.
20. R. Allmendinger, J. Handl, and J. Knowles, “Multiobjective optimization: when objectives exhibit non-uniform latencies,” *European Journal of Operational Research*, vol. 243, pp. 497-513, 2015.
21. T. Chugh, R. Allmendinger, and V. Ojalehto, and K. Miettinen, “Surrogate-assisted evolutionary biobjective optimization for objective with non-uniform latencies,” In *GECCO* 18, pp. 15–19, Japan, 2018.
22. R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, “A reference vector guided evolutionary algorithm for many-objective optimization,” *IEEE Trans. on Evolutionary Computation*, vol. 20, no. 5, pp. 773-791 ,2016.
23. H. Wang, Y. Jin, and J. Doherty, “Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems,” *IEEE Trans. on Cybernetics*, vol. 47, no. 9, pp. 2664-2677, 2017.
24. C. Sun, Y. Jin, R. Cheng, J. Ding, and J. Zeng, “Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems,” *IEEE Trans. on Evolutionary Computation*, vol. 21, no. 4, pp. 644-660, 2017.
25. D. Guo, Y. Jin, J. Ding, and T. Chai, “Heterogeneous ensemble-based infill criterion for evolutionary multiobjective optimization of expensive problems,” *IEEE Trans. on Cybernetics*, vol. 49, no. 3, pp. 1012-1025, 2019.
26. S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, vol. 290, no. 5500, pp. 2323-2326, 2000.
27. A. Grover and J. Leskovec, “Node2vec: scalable feature learning for networks,” In: *KDD*’16, pp. 855–864, San Francisco, USA, 2016.
28. D. Wang, P. Cui, and W. Zhu, “Structural deep network embedding,” In: *KDD*’ 16, pp. 1225-1234, San Francisco, USA, 2016.
29. X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang, “Community preserving network embedding,” In: *AAAI*-17, pp. 203-209, San Francisco, USA, 2017.
30. S. Wang, J. Liu, and Y. Jin, “Surrogate-assisted robust optimization of large-scale networks based on graph embedding”, *IEEE Trans. on Evolutionary Computation*, vol. 24, no. 4, pp. 735-749, 2020.
31. T. Chugh, Y. Jin, K. Miettinen, J. Hakanen, and K. Sindhya, “A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization,” *IEEE Trans. on Evolutionary Computation*, vol. 22, no. 1, pp. 129-142, 2018.
32. Y. Jin, “Surrogate-assisted evolutionary computation: recent advances and future challenges,” *Swarm and Evolutionary Computation*, vol. 1, pp. 61-70, 2011.
33. H. Wang, Y. Jin, and J. Jansen, “Data-driven surrogate-assisted multiobjective evolutionary optimization of a trauma system,” *IEEE Trans. on Evolutionary Computation*, vol. 20, no. 6, pp. 939-952, 2016.
34. D. Buche, N. Schraudolph, and P. Koumoutsakos, “Accelerating evolutionary algorithms with Gaussian process fitness function models,” *IEEE Trans. on Systems, Man, and Cybernetics, Part C*, vol. 35, no. 2, pp. 183-193, 2005.
35. R. L. Hardy, “Multiquadric equations of topography and other irregular surfaces,” *Journal of Geophysical Research*, vol. 76, no. 8, pp. 1905-1915, 1971.
36. Z. Zhou, Y. S. Ong, M. H. Nguyen, and D. Lim, “A study on polynomial regression and Gaussian process global surrogate model in hierarchical surrogate-assisted evolutionary algorithm,” in *Proceedings of IEEE Congress on Evolutionary Computation*, vol. 3, pp. 2832-2839, Edinburgh, U.K., 2005.
37. D. Shepard, “A two-dimensional interpolation function for irregularly-spaced data,” *Proceedings of the 1968 23rd ACM National Conference*, pp. 517–524, 1968.
38. J. Qiu, Y. Dong, H. Ma, J. Li, K. Wang, and J. Tang, “Network embedding as matrix factorization: unifying deepwalk, line, pte and node2vec,” In *WSDM*’ 18, pp. 459-467, Los Angeles, USA, 2018.
39. S. Wang and J. Liu, “Robustness of single and interdependent scale-free interaction networks with various parameters,” *Physica A*, vol. 460, pp. 139-151, 2016.
40. B. Anthony and S. Terry, “The independent components of natural scenes are edge filters,” *Vision Research*, vol. 37, no. 23, pp. 3327-3338, 1997.
41. D. J. Watts and S. H. Strogatz, “Collective dynamics of small-world networks,” *Nature*, vol. 393, pp. 440-442, 1998.
42. P. Erdős and A. Rényi, “On the evolution of random graphs,” *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, vol. 5, pp. 17-61, 1960.
43. J. Doyle, D. Alderson, L. Li, S. Low, M. Rouqhan, S. Shalunov, R. Tanaka, and W. Willinger, “The "robust yet fragile" nature of the Internet,” *PNAS*, vol. 102, no. 41, pp. 14497-14502, 2005.
44. V. Batagelj and A. Mrvar. *Pajek Datasets*. <http://vlado.fmf.uni-lj.si/pub/networks/data>.
45. J. Wu, S. Tan, Z. Liu, Y. Tan, and X. Lu, “Enhancing structural robustness of scale-free networks by information disturbance,” *Scientific Reports*, vol. 7, p. 7559, 2017.
46. S. Wang and J. Liu, “Constructing robust community structure against edge-based attacks,” *IEEE Systems Journal*, vol. 13, no. 1, pp. 582-592, 2019.
47. B. Wang, L. Gao, and Y. Deng, “Maintain the structural controllability under malicious attacks on directed networks,” *Europhysics Letters*, vol. 101, no. 58003, 2013.
48. D. Shizuka and D. Farine, “Measuring the robustness of network community structure using assortativity,” Animal Behavior, vol. 112, pp. 237-246, 2016.
49. M. Prec, “Evolution of cooperation on scale-free networks subject to error and attack,” New Journal of Physics, vol. 11, no. 033027, 2009.
50. S. Wang, J. Liu, and Y. Jin, “Finding influential nodes in multiplex networks using a memetic algorithm,” *IEEE Trans. on Cybernetics*, in press, 2019.

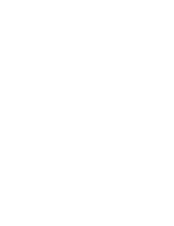


**Shuai Wang** received the B.S. degree in intelligent science and technology from Xidian University, Xi'an, China in 2015. Now, he is pursuing the Ph.D. degree in circuits and systems from the School of Artificial Intelligence, Xidian University. His research interests include complex networks and evolutionary algorithms.



**Jing Liu** (SM’15) received the B.S. degree in computer science and technology and the Ph.D. degree in circuits and systems from Xidian University in 2000 and 2004, respectively. In 2005, she joined Xidian University as a lecture, and was promoted to a full professor in 2009. From Apr. 2007 to Apr. 2008, she worked at The University of Queensland, Australia as a postdoctoral research fellow, and from Jul. 2009 to Jul. 2011, she worked at The University of New South Wales at the Australian Defence Force Academy as a research associate.

Now, she is a full professor in School of Artificial Intelligence, Xidian University. Her research interests include evolutionary computation, complex networks, fuzzy cognitive maps, multiagent systems, and data mining. She is the associate editor of IEEE Trans. Evolutionary Computation. She had been the chair of Emerging Technologies Technical Committee (ETTC) of IEEE Computational Intelligence Society from 2017-2018. Please see her homepage (https://faculty.xidian.edu.cn/LJ22/zh\_CN/index.htm) for more information.

**Yaochu Jin** (M’98–SM’02–F’16) received the B.Sc., M.Sc., and Ph.D. degrees from Zhejiang University, Hangzhou, China, in 1988, 1991, and 1996, respectively, and the Dr.-Ing. degree from Ruhr University Bochum, Bochum, Germany, in 2001.

He is a Distinguished Chair Professor in Computational Intelligence with the Department of Computer Science, University of Surrey, Guildford, U.K., where he heads the Nature Inspired Computing and Engineering Group. He was also a Finland Distinguished Professor funded by the Finnish Funding Agency for Innovation (Tekes) and a Changjiang Distinguished Visiting Professor appointed by the Ministry of Education, China. He has (co)-authored over 300 peer-reviewed journal and conference papers and been granted eight patents on evolutionary optimization. He has delivered 30 invited keynote speeches at international conferences. His current research interests include computational intelligence, computational neuroscience, computational systems biology, and nature-inspired and realworld driven problem-solving.

Dr. Jin is the recipient of the Best Paper Award of the 2010 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, the 2015, 2017 and 2019 IEEE Computational Intelligence Magazine Outstanding Paper Award, and the 2018 and 2021 IEEE Transactions on Evolutionary Computation Outstanding Paper Award. He is the Editor-in-Chief of the IEEE Transactions on Cognitive and Developmental Systems and the Editor-in-Chief of Complex & Intelligent Systems. He is an IEEE Distinguished Lecturer for the period 2013–2015 and 2017–2019, and past Vice President for the Technical Activities of the IEEE Computational Intelligence Society from 2014 to 2015. He is a Fellow of IEEE.

1. This work was supported in part by the General Program of National Natural Science Foundation of China (NSFC) under Grant 61773300, and in part by the Royal Society under Grant IEC\NSFC\170279. *Corresponding authors: Jing Liu and Yaochu Jin*

   S. Wang is with School of Artificial Intelligence, Xidian University, Xi'an 710071, China. E-mail: shuai\_wang\_xd@163.com.

   J. Liu is with School of Artificial Intelligence, Xidian University, Xi'an 710071, China. E-mail: neouma@mail.xidian.edu.cn, neouma@163.com.

   Y. Jin is with the Department of Computer Science, University of Surrey, Guildford, GU2 7XH, UK. Email: yaochu.jin@surrey.ac.uk. [↑](#footnote-ref-1)