axioms

MDPI

Article

Multi-Objective Optimization of the Robustness of Complex Networks Based on the Mixture of Weighted Surrogates

Junfeng Nie, Zhuoran Yu and Junli Li *

School of Computer Science, Sichuan Normal University, Chengdu 610101, China; junf.nie@foxmail.com (J.N.); yuzhuor@foxmail.com (Z.Y.)
* Correspondence: lijunli@sicnu.edu.cn

Abstract: Network robustness is of paramount importance. Although great progress has been achieved in robustness optimization using single measures, such networks may still be vulnerable to many attack scenarios. Consequently, multi-objective network robustness optimization has recently garnered greater attention. A complex network structure plays an important role in both node-based and link-based attacks. In this paper, since multi-objective robustness optimization comes with a high computational cost, a surrogate model is adopted instead of network controllability robustness in the optimization process, and the Dempster–Shafer theory is used for selecting and mixing the surrogate models. The method has been validated on four types of synthetic networks, and the results show that the two selected surrogate models can effectively assist the multi-objective evolutionary algorithm in finding network structures with improved controllability robustness. The adaptive updating of surrogate models during the optimization process leads to better results than the selection of two surrogate models, albeit at the cost of longer processing times. Furthermore, the method demonstrated in this paper achieved better performance than existing methods, resulting in a marked increase in computational efficiency.

Keywords: multi-objective optimization; controllability robustness; surrogate model; Dempster–Shafer theory; complex network

Citation: Nie, J.; Yu, Z.; Li, J. Multi-Objective Optimization of the Robustness of Complex Networks Based on the Mixture of Weighted Surrogates. *Axioms* **2023**, *12*, 404. https://doi.org/10.3390/axioms12040404

Academic Editor: J. Alberto Conejero

Received: 9 March 2023
Revised: 12 April 2023
Accepted: 19 April 2023
Published: 21 April 2023

1. Introduction

The Internet, transportation networks, and interpersonal connections [1] are all examples of networked systems, which are pervasive in both biological and societal contexts. Due to their complex structures, it can be challenging to analyze their dynamics. To address this issue, researchers have developed sophisticated network models to effectively represent these systems. Common simulation network models are the Erdős–Rényi (ER) network [2], small-word (SW) network [3], scale-free (SF) network [4], random triangle (RT) network [5], random rectangular (RR) network [6] and q-snapback (QS) network [7]. Most networks are presently reconfigured by rewiring strategies to form a larger network structure [8].

Recently, network controllability robustness has become a focal topic in complex network studies. The concept of controllability refers to the capacity of a network to move from any initial state to any target state with an admissible control input within a finite duration of time, while network controllability robustness measures the ability of a network to maintain its controllability under attack or failure [9].

The operating environment of such networks includes uncertainty and disturbances, which adds to the complexity of network systems. Complex networks are expected to be robust in the face of uncertainty, disruptions, and attacks, i.e., networks must be able to properly function even in the presence of attacks and errors. Network robustness under random or malicious attack serves as a benchmark for judging its normal operation under interference. Random attacks aim to randomly target nodes or links, while malicious

attacks seek to identify and compromise the most critical components of the network [10]. Single-objective optimization can be used to define complex network structures, and single-objective evolutionary algorithms can be employed to enhance the network robustness against node-based or link-based attacks. Previous studies utilized network robustness as the optimization objective, and a surrogate model was adopted to assist single-objective evolutionary algorithms in optimizing network robustness [11].

Currently, numerous approaches are being employed to create networks with enhanced robustness. For example, the SP_RV_MOEA$_{\mathrm{Net}}$ method was used to optimize the complex network structure [12]. The simulated annealing approach is employed to address the optimization problem of enhancing the network robustness [13]. A memetic algorithm is employed to address the network structure seeking problem, incorporating evolutionary operators such as selection, crossover, and mutation, along with a local search process based on structural information, to obtain better optimization results [14,15]. In the study [10], network robustness is enhanced by rewiring the topology.

Multi-objective evolutionary algorithms have been used to solve a variety of engineering and material science problems. In multi-objective controllability robustness optimization, robustness under node attack is negatively correlated with the robustness under link attack [16]. In practical applications, computing the network robustness can be a time-consuming process. According to the literature [17], it takes a long time to search for network architectures with better robustness when using multi-objective evolutionary algorithms to solve network robustness optimization problems. Surrogate models have been used by some researchers in recent years to replace the evaluation of network robustness in the optimization process.

Surrogate models can be either non-interpolating or interpolating. Non-interpolating models include, for example, polynomial regression [18] models and support vector machines [19]. Radial basis function (RBF) networks, least squares (LS) method, the inverse distance weighting (IDW) interpolation method [11], and Kriging interpolation method (Kriging) [20] are interpolating surrogate models. There is no clear criterion for determining how to select a good surrogate model for a complex network in order to evaluate its controllability robustness.

The Dempster–Shafer theory (D-S theory) [21] is used to combine surrogate models. This theory was first introduced by Arthur P. Dempster [22] in the context of statistical inference, and it has the ability to handle uncertain information. Subsequently, Glenn Shafer [23] further quantitatively extended the theory to Bayesian inference methods, utilizing Bayesian conditional probabilities derived from probability theory and experimentally a priori known probabilities. The theory is a mathematical theory of evidence that allows for the mixture of information from various sources in order to construct a degree of belief. The theory permits the mixture of imprecise and uncertain information, which may even be contradictory. So-called basic probability assignments (BPAs) contain information about specific hypotheses (focal elements) and are combined to calculate the credibility of a given hypothesis. Three functions are usually associated with BPAs, namely the belief (Bel), plausibility (Pl) and pignistic probability (BetP) function.

In terms of surrogate model, the BPAs can be, for example, model characteristics such as correlation coefficients (CCs), root mean squared errors (RMSEs), and maximal absolute errors (MAEs). It is possible that one surrogate model has conflicting characteristics, i.e., good (e.g., high correlation coefficients), bad (e.g., high RMSE) characteristics, and bad (e.g., high MAE) characteristics. This conflict must be considered when calculating the belief that one has in the given model. Several conflict redistribution rules have been developed in the literature. Dempster's combination rule redistributes the conflict among all focal elements, regardless of which elements cause the conflict.

The remainder of this paper is structured as follows: Section 2 introduces the work relevant to this article. Section 3 introduces the algorithm framework in detail. Section 4 discusses the surrogate model selection and evaluation method. Experimental results are

reported in Section 5. Section 6 discusses the advantages and limitations of this method. Section 7 provides the conclusion and outlines future work.

2. Related Work

2.1. Network Controllability Robustness

For a network topology graph, N nodes and M links are recorded, and the adjacency matrix between networks is saved (with the link recorded as 1 and nodes as 0). When considering a network of many LTI systems, the node system with control input is called a driver node. The network controllability is quantitatively expressed by the density of driver nodes [24] (cn_D) in the network, which is calculated as follows:

$$cn_D = \frac{N_D}{N}, \tag{1}$$

where N_D is the number of driver nodes required in the network and N is the total number of nodes in the network, $cn_D \in [\frac{1}{N}, 1]$, when $cn_D = \frac{1}{N}$ means that the current network of N nodes requires only one driver node, and the network has the best controllability; $cn_D = 1$ means that all nodes in the current network are isolated, so each node requires a controller, and the network has the worst controllability.

The most common methods for calculating controllability are structural controllability [25] and exact controllability [26], where the network structure targeted by structural controllability is a directed graph, which is time-consuming and even impossible for large networks. The exact controllability adopted in this paper is applicable to all sparse networks, undirected graphs, and directed graphs. The required drive nodes are calculated as follows:

$$N_D = max\{1, N - rank(A)\}, \tag{2}$$

where A is the network adjacency matrix, rank is used to calculate the rank of the matrix. If matrix A is in full rank, then the $cn_D = 1$ driver node is required; otherwise, the $N - rank(A)$ driver nodes are required. Network controllability under node-based attack is defined as follows:

$$cn_D^N(i) = \frac{N_D(i)}{N - i}, i = 0, 1, \ldots, N - 1, \tag{3}$$

where $N_D(i)$ is the number of driver nodes required when i nodes are attacked; and $N - i$ is the number of remaining nodes in the network after i nodes are attacked, which decreases one by one with each attack. Similarly, network controllability under link-based attack is defined as follows:

$$cn_D^E(j) = \frac{N_D(j)}{N}, j = 0, 1, \ldots, M, \tag{4}$$

where $N_D(j)$ the number of driver nodes required when the link of j is attacked; N and M denote the number of network nodes and links, respectively. Under link attack, the number of nodes remains constant while the number of links decreases one by one in a continuous link-based attack. Equations (3) and (4) define the dynamic process of controllability change under attack. The overall controllability robustness can be obtained by averaging controllability, and the equation is as follows:

$$R = \frac{1}{N} \sum_{i=0}^{N-1} cn_D^N(i), \tag{5}$$

$$RL = \frac{1}{M+1} \sum_{j=0}^{M} cn_D^E(j), \tag{6}$$

a lower R or RL indicates better network controllability robustness [27–29] under node-based or link-based attack. A similar approach used the connectivity robustness [30,31] metric.

2.2. Surrogate Models

If no previous experiment was conducted, determining the optimal surrogate model for various complex networks may require a large number of experiments. To simplify the process, multiple surrogate models can be used to optimize the selection. Different surrogate models can be selected to suit different complex networks, and it must be ensured that the results obtained using these models are close to the true values. With the development of surrogate modeling techniques, they have been widely applied to optimize single-objective and multi-objective problems.

When analyzing graphs, computation resources must be consumed. Methods that can efficiently parse the network have been emphasized, and a successful one is the use of representation learning, i.e., the structural deep network embedding (SDNE) [32,33]. SDNE focuses on obtaining the system's latent representation while preserving its structural information. In general, the graph with the $N * N$ connection matrix is transformed into $N * d$ vectors (d tends to be smaller than N here); By reducing the dimensionality while still retaining sufficient structural information, these vectors can accelerate the resolution of tasks involving networks.

In addition, heterogeneous input can foster diversity in integration, leading to the extraction of various features from the original data. The extracted features and the original data can be combined to form the training data of the surrogate model. To achieve this, principal component analysis (PCA) can be employed as it can effectively extract the major components with minimal computational cost. The second step is to cluster the embedded information using K-means clustering with the number of clusters set to twenty percent of N. As demonstrated in [34], the clustered embedded data, which contains essential connectivity information, is essential for interpreting information from networks. The original data obtained after applying SDNE for dimensionality reduction, as well as those obtained through PCA dimensionality reduction and those obtained after clustering are all used to train the surrogate model.

Given the network data, the robustness measures (R and RL) of the networks are evaluated first, and networks are recoded by SDNE to obtain their 2D representations. Subsequently, two feature extraction techniques, along with the original embedded graph, are used as the input for the surrogate. The surrogate is able to estimate the robustness values and the uncertainty of the estimates, as shown in Figure 1. The surrogate can later be used to facilitate network robustness optimization.

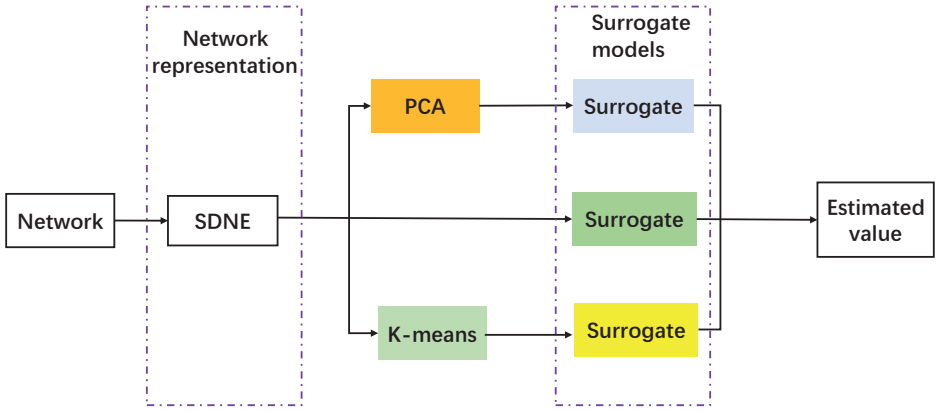

Figure 1. The structure of the surrogate ensemble for estimating network robustness.

The network robustness measure (R, RL) is first calculated from the given network structure, and then the network is dimensionally reduced using SDNE to obtain the two-dimensional data (a_i, b_i) of the network, where i is the i-th network nodes. Following that, the original data as well as two feature extraction techniques are used as heterogeneous

inputs for training the surrogate models. The following section outlines the surrogate models employed in this article.

2.2.1. Radial Basis Function (RBF) Networks

The use of radial basis functions (RBFs) is critical for this network, as it enables the creation of a symmetric radial basis function for every interpolation point in the function. These basis functions can then be combined linearly, providing an accurate approximation of the true objective value. The surrogate model can be represented as follows:

$$\hat{y}(x) = \sum_{i=1}^{n} \lambda_i \phi(\|x_i - c\|_2), \tag{7}$$

where $\|.\|$ denotes the Euclidean norm, $\lambda_1, \lambda_2, \ldots\ldots, \lambda_n$ is the weight coefficient, $\phi(\|x_i - c\|_2) = exp(-\|x_i - c\|/\sigma^2)$ is the radial basis function, the distance from the prediction point x_i to the training point c is the key in the basis function, and a coefficient will be trained before making predictions. The RBF can be taken as a global polynomial to fit the general trends in training data.

2.2.2. Least Square (LS) Method

Linear approximation can also be used to evaluate function values, We consider a linear model as follows:

$$\widehat{y} = Ax + v, \tag{8}$$

where $x \in R^n$ is the vector of unknowns, $y \in R^m$ is the vector of observations, $v \in R^m$ is the noise, and $A \in R^{m*n}$ is the data matrix.

When $m \geq n$ and the columns of A are linearly independent, $\|Ax - \widehat{y}\|_2^2$ is the minimum multiplier loss function.

2.2.3. Inverse Distance Weighting (IDW) Interpolation Method

The inverse distance weighting interpolation method judges the similarity between each other by Euclidean distance, and the similarity is higher if the distance is closer. Therefore, the weight is judged according to the distance prediction value, and a small prediction value is instead heavily weighted, and vice versa. A surrogate model can be represented as follows:

$$\hat{y} = \sum_{i=1}^{n} \lambda_i Z(x_i, y_i), \tag{9}$$

where x_i, y_i are the values of the i-th known point, n is the number of known data points, $\lambda_i = \frac{1/d_i}{\sum_{i=1}^{n} 1/d_i}$ is the weight corresponding to the known point, d_i is the Euclidean distance between known and predicted points, and the weight is a function of the inverse of the distance. $Z(x_i, y_i)$ is the attribute value corresponding to the i-th known point.

2.2.4. Kriging Interpolation Method

Kriging is a method of spatially interpolating stochastic processes using a covariance function. It uses variables to create a function that provides an accurate, optimal estimate of unknown data. Kriging models consist of two components. The first part is a basic model that describes the data's trend, while the second is stochastic and measures the discrepancy between the simple model and the actual function. This difference is modeled using the covariance function. A model $\hat{y}$ is built as a realization of a regression model and a random function z in order to express the deterministic response y for the input, and Kriging model can be represented as follows:

$$\hat{y} = \sum_{i=1}^{n} \lambda_i f(x_i, y_i) + z(x), \tag{10}$$

where $\hat{y}$ is the approximation function, $f(x_i, y_i)$ denotes the regression function, which is a polynomial in the independent variable (x_i, y_i), λ_i is the weight coefficient, and n is the number of known points. The random process z is assumed to have zero mean and a covariance that depends on an underlying correlation model, including parameters that must be optimized. Commonly used correlation functions are exponential, generalized exponential, Gaussian, spherical, or spline, and regression models can be selected as constant, linear, or quadratic.

2.3. New Contributions of the Proposed Algorithm

Focusing on the multi-objective network robustness optimization problem, this work addresses the issue of the high cost of evaluating controllability robustness, thereby improving the overall efficiency of the optimization process. Simulations on four types of synthetic networks, namely SF, ER, SW, and RR, were used to validate the effectiveness of the approach used in this paper. Specifically, the work and contributions of this paper are summarized as follows:

1. The D-S theory was used to calculate the weights of the four surrogate models, and then two or three of them with their corresponding weights were chosen for the optimization process. Compared to selecting three surrogate models, selecting two surrogate models resulted in improved controllability robustness in a shorter amount of time.

2. As the number of iterations increased, an adaptive updating surrogate model selection approach is necessary to attain an optimal solution. This approach offers superior results compared to previous methods, however, it necessitates enhanced optimization time. To achieve the best outcome without compromising on optimization time, an adaptive updating surrogate model selection approach is recommended.

3. Algorithm Framework

This paper focuses on the two-objective problem, and therefore the function of the two-objective problem is shown here. A multi-objective optimization problem (MOP) can be stated as follows: the formula of multi-objective optimization is shown in Equation (11).

$$\begin{cases} min \quad F(r) = (R(r), RL(r))^T, \\ subject \quad to : r \in \Omega, \end{cases} \tag{11}$$

where Ω is the network space and r is a network structure. $F(r) : \Omega \to \mathbb{R}^2$ is a two-dimensional objective vector. $R(r)$ and $RL(r)$ are the network controllability robustness under node attacks and link attacks.

Very often, since the objectives in (11) contradict each other, no point in Ω maximizes all the objectives simultaneously. One has simply to balance them. The best trade-offs among the objectives can be defined in terms of Pareto optimality.

Let $\mu, v \in R^2$, and μ is said to dominate v if and only if $\mu_i \geq v_i$ for every $i \in \{1, 2\}$ and $\mu_j > v_j$ for at least one index $j \in \{1, 2\}$. A point $r^* \in \Omega$ is Pareto optimal to (11) if there is no point $r \in \Omega$ such that $F(r)$ dominates $F(r^*)$. $F(r^*)$ is then called a Pareto optimal (objective) vector. In other words, any improvement in a Pareto optimal point in one objective must lead to deterioration in at least one other objective. The set of all Pareto optimal points is called the Pareto set (PS) and the set of all the Pareto optimal objective vectors is the Pareto front (PF) [35].

The optimization steps of the algorithm are as follows: (1) Initialize the network structure; (2) Individuals' controllability robustness is computed in terms of the population; (3) The network in the population and the associated robustness value is used as training data to build the surrogate model; (4) During the subsequent crossover, mutation, and selection evolution, the trained surrogate model is used to estimate the robustness measure; (5) Update the surrogate model; and (6) Update the non-dominate solutions. The pseudocode of the multi-objective optimization Algorithm 1:

Algorithm 1: Multi-objective evolutionary algorithm optimization.

Input:
Initialize network population P;
Iterations $t = 0$;
Max iterations MaxGen;
output:
The non-dominated solution set and network structure;
Calculate the network controllability robustness and initialize the surrogate model;
While $t <$ MaxGen:
Conduct the crossover operator on P_t to generate Q_t;
Conduct the mutation operator on Q_t;
Select better individuals from P_t and Q_t to P_{t+1};
Conduct the local search operator on P_{t+1};
Update the surrogate model;
Update EP with P_{t+1};
$t = t + 1$;
end while

The crossover operator selects two networks from the population, and then randomly selects an edge from each network structure, and exchanges them with each other.

In the mutation operation, the topological rewiring operator seeks to modify the network's connections without changing its degree of distribution. That is, two edges are randomly selected, namely e_{ij} and e_{kl} in the network, and e_{il} and e_{kj} are added first, then e_{ij} and e_{kl} are deleted (i, j, k, and l represent four nodes, and e represents the connection between nodes).

At the end of a generation, the selection operator is used to select the best candidates from the population of parent candidates and update them into the child population with the initial population of candidates. The best candidate is then saved as the first one in the child population.

The local search operator seeks to improve the quality of the obtained solutions without relying on surrogates, a time-consuming but essential process for enhancing the R or RL of the individuals.

4. Selection and Mixture of Surrogate Models

4.1. Dempster–Shafer Theory Weighting Method

Rather than calculating the real controllability robustness, this paper focuses on selecting appropriate surrogate models for a specific complex network to assist the evolutionary algorithm in optimizing the structure of the complex network. As shown in Figure 2, the steps of the weight evaluation of surrogate models under the D-S theory are illustrated.

In Figure 2, the input data are the data of the network structure after SDNE dimensionality reduction. The middle part is used to calculate the correlation coefficient (CC), root mean square error (RMSE), and maximal absolute error (MAE) between the predicted value obtained from the surrogate model and the true values; then the weights of the four surrogate models are calculated using the D-S theory. Next, the weights are brought into the surrogate model evaluation process (select one, two, or three surrogate models according to the weights). As the input data are in various forms, as shown in Figure 1, the weighted average value obtained is taken as the prediction value. In the later experiments, two or three surrogate models with the highest weights are selected. The goal of this paper is to apply the D-S theory to select the most suitable surrogate model for a given complex network among different surrogate models according to the weight, so that the predicted value of the selected surrogate model is close to the true value.

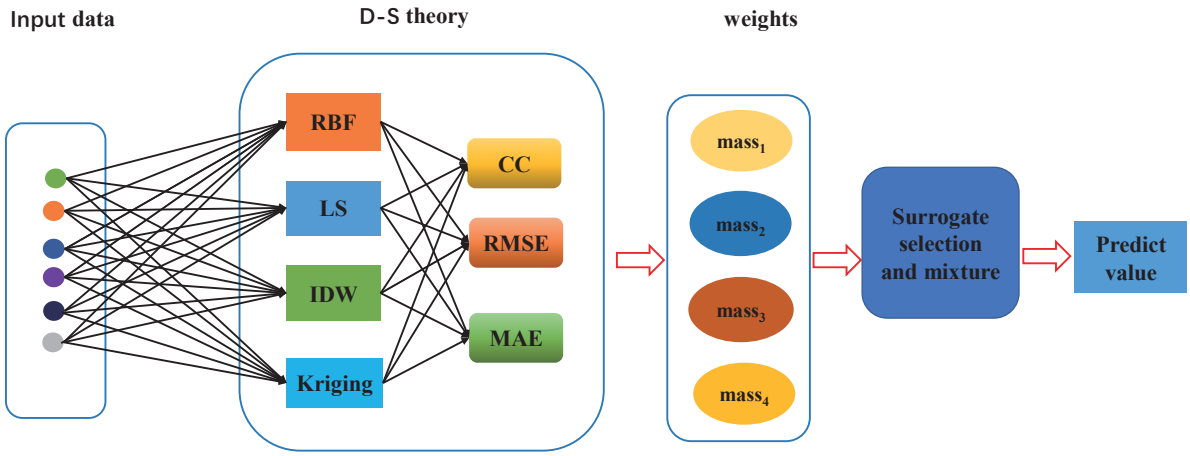

Figure 2. D-S theory weighted evaluation of surrogate model steps.

4.2. Single Selection of Surrogate Models

Surrogate models can replace expensive evaluation functions. If it is unknown which surrogate model is best suited to the problem at hand, different models must be tried to find the most effective one. Next, the challenge becomes determining the best surrogate model.

In this section, we identify appropriate surrogate models for various complex networks, then integrate them into a multi-objective evolutionary algorithm to investigate a network structure that can withstand both node-based and link-based attacks. This paper employs the Dempster–Shafer (D-S) theory to select the best surrogate model based on the weights of these four surrogate models. Algorithm 2 provides the pseudocode for calculating weights using the D-S theory.

Algorithm 2: The process of calculating the weights of D-S theory.

Input:
The number of surrogate: N;
The number of complex networks: G;
output:
Weights of different surrogate models in complex networks;
Randomly initialize the complex network and set iterator $t = 0$;
Calculate the true network controllability robustness;
While $t < N$:
Training surrogate models with controllability robustness at population;
The trained surrogate model is used to evaluate the controllability robustness of the network;
Calculate the correlation coefficient between the true value and the predicted value m_t^{CC};
Root mean square error m_t^{RMSE} and maximal absolute errors m_t^{MAE};
Several feature attribute values have been calculated in the previous step, and D-S theory
is used to calculate the weights under these feature attribute values;
$t = t + 1$;
end while
Output the weight of each surrogate model;

The weight of each surrogate model is calculated using the predicted values obtained from the surrogate models and the corresponding true values based on the D-S theory weights calculated in Table 1. In this paper, D-S theory is utilized to select the optimal single surrogate model for this complex network, and the surrogate model is utilized to guide the multi-objective evolutionary algorithm to the best solution.

Table 1. Model characteristics of the four types of networks.

Surrogates		**SF**			**ER**	
	CC	RMSE	MAE	CC	RMSE	MAE
RBF	0.25	0.252	0.252	0.25	0.279	0.285
LS	0.25	0.245	0.241	0.25	0.212	0.202
IDW	0.25	0.252	0.253	0.25	0.267	0.265
Kriging	0.25	0.251	0.253	0.25	0.242	0.248
Surrogates		**SW**			**RR**	
	CC	RMSE	MAE	CC	RMSE	MAE
RBF	0.25	0.250	0.248	0.25	0.251	0.247
LS	0.25	0.250	0.247	0.25	0.249	0.250
IDW	0.25	0.251	0.255	0.25	0.250	0.252
Kriging	0.25	0.249	0.246	0.25	0.250	0.251

After the model characteristics have been calculated, BPAs are computed for every model. For this purpose, the model characteristics are scaled so that the sum over all models for each BPA equals one and the non-negativity conditions of BPAs are fulfilled. The Dempster–Shafer theory is applied in order to find the pignistic probabilities for each model. Based on these values, it can be decided which of all considered models is the best, or in case mixture models are considered, which weight should be assigned to each contributing model.

When considering mixture models CC, $RMSE$ and MAE must also be calculated for each mixture in order to find the best one. The goodness values are normalized to obtain the BPAs for each mixture as follows

$$\begin{cases} m_i^{CC} = \frac{CC_i}{\sum_{j\epsilon\Omega} CC_j}, \\ m_i^{RMSE} = \frac{\frac{1}{RMSE_i}}{\sum_{j\epsilon\Omega} \frac{1}{RMSE_j}}, \\ m_i^{MAE} = \frac{\frac{1}{MAE_i}}{\sum_{j\epsilon\Omega} \frac{1}{MAE_j}}, \end{cases} \tag{12}$$

where i is the current surrogate model, Ω is the set of models contributing to the combination, and CC, $RMSE$ and MAE, respectively, correspond to three performance indicators: namely the correlation coefficient, root mean square error, and maximal absolute errors.

These three performance measures—the correlation coefficient between evaluated and true values, the root mean square error, and the maximal absolute errors—were used. Table 2 displays the weights.

Table 2. Weights of each single surrogate model.

Surrogates	**SF**	**ER**	**SW**	**RR**
RBF	0.26	0.31	0.26	0.14
LS	0.23	0.19	0.24	0.13
IDW	0.27	0.27	0.27	0.37
Kriging	0.24	0.23	0.23	0.36

The uncertainty of the above networks with different weights is evaluated using D-S theory. For each network, the surrogate model with the highest ranking weight is chosen(Select the surrogate model according to the bold data in the table.) by referring to the weight data in Table 2.

4.3. Mixture of Weighted Surrogate Models

As shown, for example, by [36], one surrogate model does not fit all types of problems. For instance, one surrogate model may perform very well on one type of problem but

poorly on another. When evaluating a set of surrogate models, the weight rankings of each model can be used to make a decision. After applying D-S theory, the top few surrogate models are chosen based on the ranking of the four weights needed, and the weights are recalculated in accordance with Equation (13).

In the selection of two surrogate models, the weight is calculated as follows:

$$\begin{cases} w_K = \frac{BetP(K)}{BetP(K)+BetP(R)}, \\ w_R = \frac{BetP(R)}{BetP(K)+BetP(R)}, \end{cases} \tag{13}$$

where K and R are the two surrogate models chosen based on their ranking. $BetP(K)$ and $BetP(R)$ are calculated under D-S theory. The w_K and w_R are recalculated by Equation (14). The final evaluation value is defined as follows:

$$\hat{y}(x) = w_K \hat{y}_K(x) + w_R \hat{y}_R(x), \tag{14}$$

where $\hat{y}_K(x), \hat{y}_R(x)$ are the evaluated values of the two models at point x, respectively. The choice of surrogate models is shown in Table 3.

Table 3. Two mixture surrogate models.

Surrogates	SF	ER	SW	RR
RBF	0.51	0.47	0.51	0.59
LS	0.49	0.53	0.49	-
IDW	-	-	-	-
Kriging	-	-	-	0.41

For the selection of the three surrogate models, the same method is used as for the selection of the two surrogate models, as follows:

$$\begin{cases} w_K = \frac{BetP(K)}{BetP(K)+BetP(R)+BetP(Q)}, \\ w_R = \frac{BetP(R)}{BetP(K)+BetP(R)+BetP(Q)}, \\ w_Q = \frac{BetP(Q)}{BetP(K)+BetP(R)+BetP(Q)}, \end{cases} \tag{15}$$

where $BetP(K)$, $BetP(R)$, and $BetP(Q)$ are the weights of the single surrogate model. w_K, w_R, and w_Q are recalculated from the top three surrogate models selected according to Equation (16). The recalculated weights are shown in Table 4, and the evaluation value of the network is as follows:

$$\hat{y}(x) = w_k \hat{y}_k(x) + w_R \hat{y}_R(x) + w_Q \hat{y}_Q(x), \tag{16}$$

Table 4. The weights assignment of the surrogate model for the three mixtures.

Surrogates	SF	ER	SW	RR
RBF	0.35	0.33	0.35	0.43
LS	0.34	0.38	0.24	0.16
IDW	-	-	0.31	-
Kriging	0.31	0.31	-	0.41

4.4. Adaptively Updating Surrogate Models

The D-S-theory-based surrogate model in the preceding section does not change the type of the model during the search process, but only updates the parameter values. When IDW and RBF models are chosen in the SF network, they are always used with an update operation to modify the model training, but no newly appropriate surrogate model is selected for the updated complex network. Therefore, this section employs D-S theory

after each iteration to re-select the most suitable surrogate models for the complex network. The weights of the four surrogate models used in this paper are recalculated through D-S theory, and the number of adaptive surrogate models is determined according to the optimal solution obtained under the single surrogate model or the mixture model. The pseudocode is shown in Algorithm 3.

Algorithm 3: Updated pseudo-code for the surrogate model.

Input:
    The number of surrogates: N;
    Maximum number of iterations: MaxGen;
    Surrogate model update probability: update_rate;
output:
    Output the two surrogate models with the largest weights;
Randomly initialize the complex network and set iterator $t = 0$;
Using the obtained controllability robustness at the population to train the surrogate model;
Evaluate the network controllability robustness on a trained surrogate model
using the evaluated complex network structure;
Calculate CC, RMSE, and MAE between the true value and the evaluated value;
The D-S theory is used to calculate the weights under three feature attribute values;
While $t <$ MaxGen:
    if random.random() < update_rate:
        D-S theory computational process is used to assign weights to the
surrogate model;
        Select the top N surrogate models based on weights;
        Retrain the surrogate model using the obtained non-dominated solution;
    $t = t + 1$;
    Output the surrogate model with the highest weights and save the weights;
end while

5. Experimental Results

In this paper, the network structure is constructed through a multi-objective evolutionary algorithm under node attack and link attack. Selecting appropriate surrogate models under D-S theory to replace the calculation of network controllability robustness in the optimization process can not only reduce the optimization cost, but also obtain a network structure with better controllability robustness. In the experiments, the nodes of SF, ER, SW, and RR are all 200, whilst the average degree is 4; the population size is 20, the number of iterations is 100, the crossover rate is 0.6, the mutation rate is 0.5, the number of topological rewiring is 50, the update rate of the surrogate model is 0.4, local_search rate is 0.7, and the output dimension of SDNE as two. It is important to note that, if the update_rate is too large, it will be time-consuming to update the surrogate model; however, if the update_rate is too small, it will not achieve the update effect [12].

5.1. Experimental Results of Single Selection and Mixture Weighted Surrogate Models

In this paper, we choose the same multi-objective evolutionary algorithm framework and compare different surrogate model mixtures. One surrogate model (MOEA_One), two surrogate models (MOEA_Two), and three surrogate models (MOEA_Three) are compared with two existing algorithms SP_RV_MOEA$_{\mathrm{Net}}$ and MOEA$_0$ in [12] on four synthetic networks. For the different methods, the non-dominated solutions obtained by multi-objective optimization are shown in Figure 3, and the running time is shown in Table 5.

Table 5. Running time (hours) and HV values of the four types of network with different methods.

Networks	Method	HV	Run_Time
SF	$MOEA_0$	0.1430	209.34
	$SP\_RV\_MOEA_{Net}$	0.1690	95.62
	MOEA_One	0.1660	57.26
	MOEA_Two	0.1929	58.65
	MOEA_Three	0.1672	107.09
ER	$MOEA_0$	0.1404	257.95
	$SP\_RV\_MOEA_{Net}$	0.1698	104.21
	MOEA_One	0.1559	70.85
	MOEA_Two	0.1721	72.34
	MOEA_Three	0.1646	124.36
SW	$MOEA_0$	0.2149	198.63
	$SP\_RV\_MOEA_{Net}$	0.2345	101.36
	MOEA_One	0.2362	30.10
	MOEA_Two	0.1721	36.54
	MOEA_Three	0.2342	102.32
RR	$MOEA_0$	0.1030	234.36
	$SP\_RV\_MOEA_{Net}$	0.1149	126.31
	MOEA_One	0.1390	42.13
	MOEA_Two	0.1671	45.57
	MOEA_Three	0.1200	102.46

Figure 3 shows that the results obtained by $MOEA_0$ are less satisfactory on four types of networks. The performance of the existing algorithm $SP\_RV\_MOEA_{Net}$ is clearly much better than $MOEA_0$. Under the methods designed in this paper, MOEA_One, MOEA_Two, and MOEA_Three also have a better performance than $MOEA_0$. Preliminarily, this shows that the non-dominated solutions obtained after adding the surrogate model in the optimization process under the same algorithmic framework are better than those of $MOEA_0$. The performance of $SP\_RV\_MOEA_{Net}$ is second only to MOEA_Two in SF, ER, and SW networks, but in RR, the performance of $SP\_RV\_MOEA_{Net}$ is only better than $MOEA_0$. However, in terms of computational cost, $MOEA_0$ is too time-consuming, and several other methods consume much less computational time than $MOEA_0$, as shown in Table 5. The MOEA_Two method under D-S theory outperforms several other tested methods and achieves a good balance between search and diversity. It can be seen from Table 5 that the more surrogate models there are, the more time-consuming the method is.

Among the five methods used in this paper, the MOEA_Two method is better than the other four methods. On the one hand, the surrogate model can be used as a low-cost fitness function evaluator to guide the optimization process, and more candidate solutions can be generated. On the other hand, the uncertainty information obtained by the surrogate model provides additional criteria for selecting individuals. The results in both Figure 3 and Table 5 validate the significant performance of MOEA_Two in designing robustness networks against multiple attacks, and the method can handle different types of synthetic networks. The bold data in Table 5 are the optimal values obtained under the method.

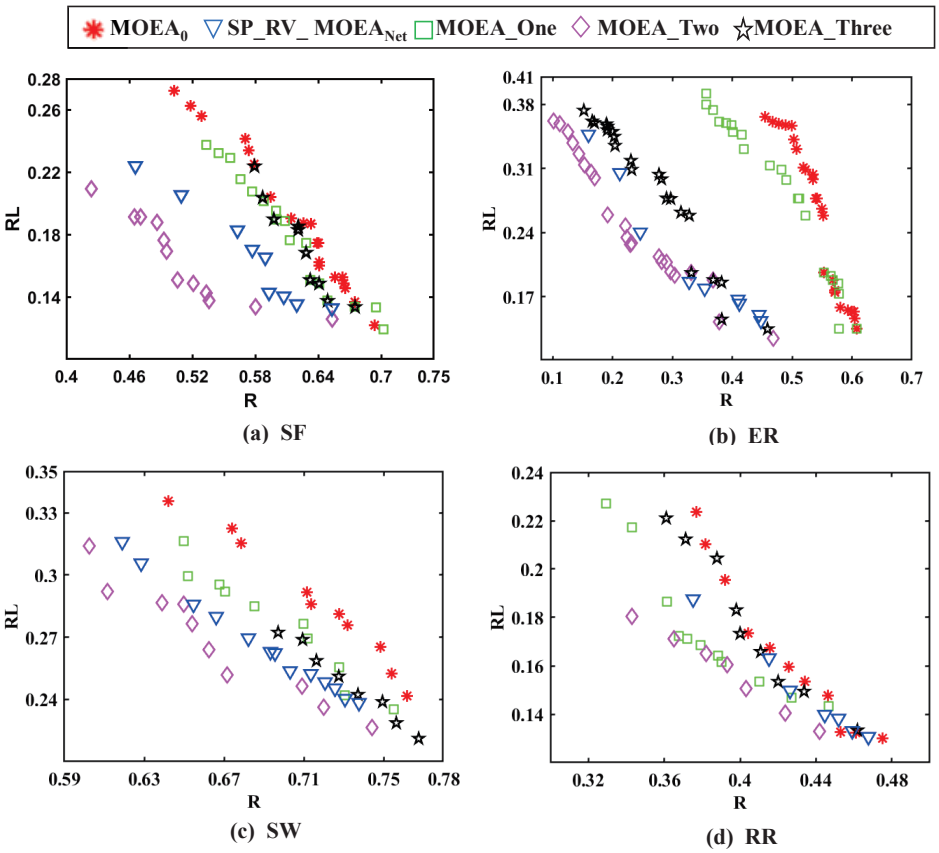

Figure 3. Non-dominated solutions of different methods for four types of complex networks.

The non-dominated solutions obtained by several methods in Figure 3 are better when evaluating the controllability robustness, which means that the smaller the non-dominated solutions, the better it is.

Hypervolume (HV) values are used to evaluate the performances of various approaches [12,37]. This estimates the volume of the region in the target space contained by the set of produced non-dominated solutions and the reference points. The higher the HV values are, the better the comprehensive performance of the algorithm is. The HV values are calculated as shown in Equation (17) as follows:

$$HV = \delta\left(\bigcup_{i=1}^{S} V_i\right), \tag{17}$$

among δ, it is a Lebesgue measure that is used to calculate the volume. V_i represents the hypervolume formed by the reference point and the ith non-dominated solution in the solution set, and S represents the number of non-dominated solutions.

The variation in HV values with the number of iterations for the five compared methods is given in Figure 4. For MOEA$_0$, the HV values of the obtained results are significantly smaller than those of the other methods. The HV values curve of MOEA_Two is above the other curves, and its result is the best. Table 6 depicts the average HV values for different methods for each complex network.

According to the experimental results, with the increase in surrogate models, the training time of surrogate models also increases. The shortest running time and highest average HV values were observed when IDW and RBF were used as a mixture of surrogate models for three complex networks. This is likely due to the different network architectures and the different weights allocated to each network.

In summary, for SF, ER, SW, and RR networks with an average degree of four at 200 nodes, IDW and RBF are selected as mixture surrogate models to obtain the best

optimization results. For the RR network specifically, IDW and Kriging are selected as the mixture surrogate models, yielding better optimization results than the other networks.

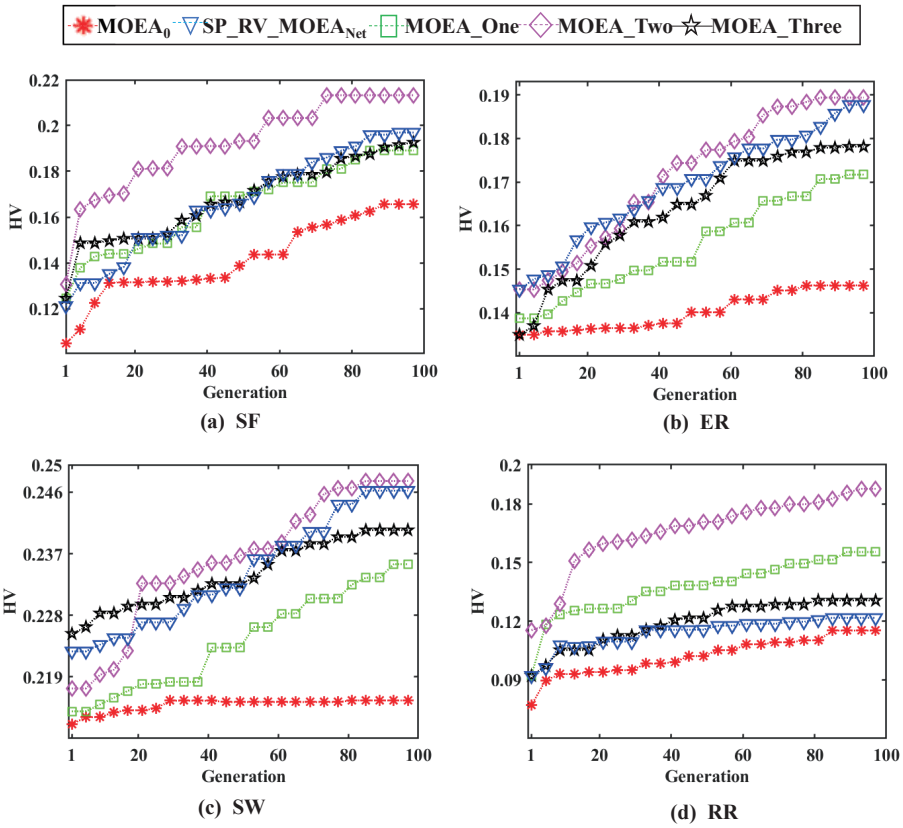

Figure 4. HV values of different methods under four types of complex networks.

The HV values is used to determine the performance of multi-objective optimization, and the larger the HV values, the better the performance of the method. From Figure 4, it can be seen that the performance of MOEA_Two is better, as the HV curve of MOEA_Two ultimately remains above other methods.

5.2. *Experimental Results of Adaptively Updating Surrogate Models*

The experimental results in the preceding section show that the network controllability robustness obtained by *MOEA_Two* assisted by the multi-objective evolutionary algorithm on SF, SW, ER, and RR are relatively better. Therefore, during the adaptive updating process, the number of surrogate models is set to two. The experiments in this section compare the controllability robustness of MOEA_Two with that of MOEA_Two_Adapt. Figure 5 depicts the non-dominated solutions obtained on four types of complex networks, and Table 4 shows the time consumption and average HV values of the two methods.

MOEA_Two in Figure 5 represents the network controllability robustness sought by two surrogate models assisted by the multi-objective evolutionary algorithm, whereas MOEA_Two_Adapt represents the network controllability robustness sought by the adaptively updating surrogate model assisted by the multi-objective evolutionary algorithm. As shown in Figure 5, the MOEA_Two_Adapt can assist the evolutionary algorithm in finding the network structure with the best controllability robustness.

The minimum time consumption and maximum HV values in Table 6 are shown in bolded to compare the time under the two approaches in the four types of networks, and reveals that the MOEA_Two_Adapt takes three times longer than the *MOEA_Two*, whilst the former has a better average HV values. If time-consuming situation is ignored, the MOEA_Two_Adapt can help the evolutionary algorithm find a network structure with improved controllability robustness. HV values curves for MOEA_Two and

MOEA_Two_Adapt are shown in Figure 6. HV values curves are better at the top of the curve, and Figure 6 clearly shows that the MOEA_Two_Adapt assists the evolutionary algorithm in its drive to achieve greater network controllability robustness.

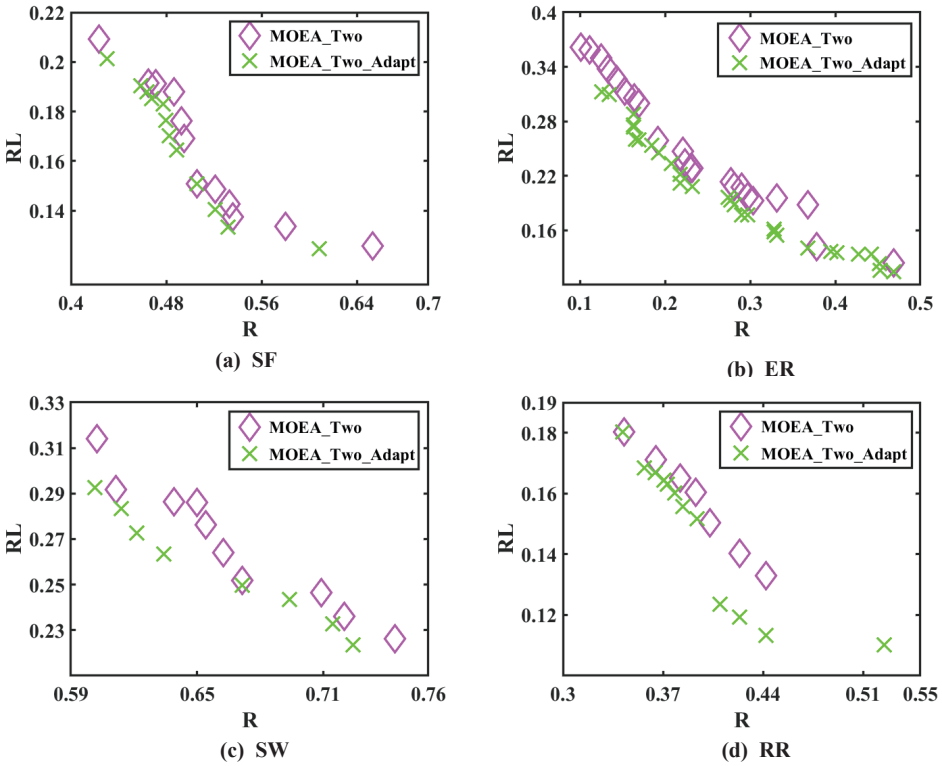

Figure 5. MOEA_Two and MOEA_Two_Adapt under four types of complex network.

Here, the optimal non-dominated solutions obtained from the previous MOEA_Two is compared with adaptively updating. Similarly, when evaluating the controllability robustness, the smaller the non-dominated solutions, the better it is.

Table 6. Running time (hours) and HV values for MOEA_Two_Adapt and MOEA_Two for four types of complex network.

Networks	Method	HV	Run_time
SF	MOEA_Two	0.1929	58.65
	MOEA_Two_Adapt	0.1939	178.59
ER	MOEA_Two	0.1721	72.34
	MOEA_Two_Adapt	0.1766	180.77
SW	MOEA_Two	0.2362	36.54
	MOEA_Two_Adapt	0.2439	176.78
RR	MOEA_Two	0.1671	45.57
	MOEA_Two_Adapt	0.1765	137.52

In comparison with the two surrogate models, the adaptive update surrogate model can achieve superior non-dominated solutions while consuming only about three times the computing power.

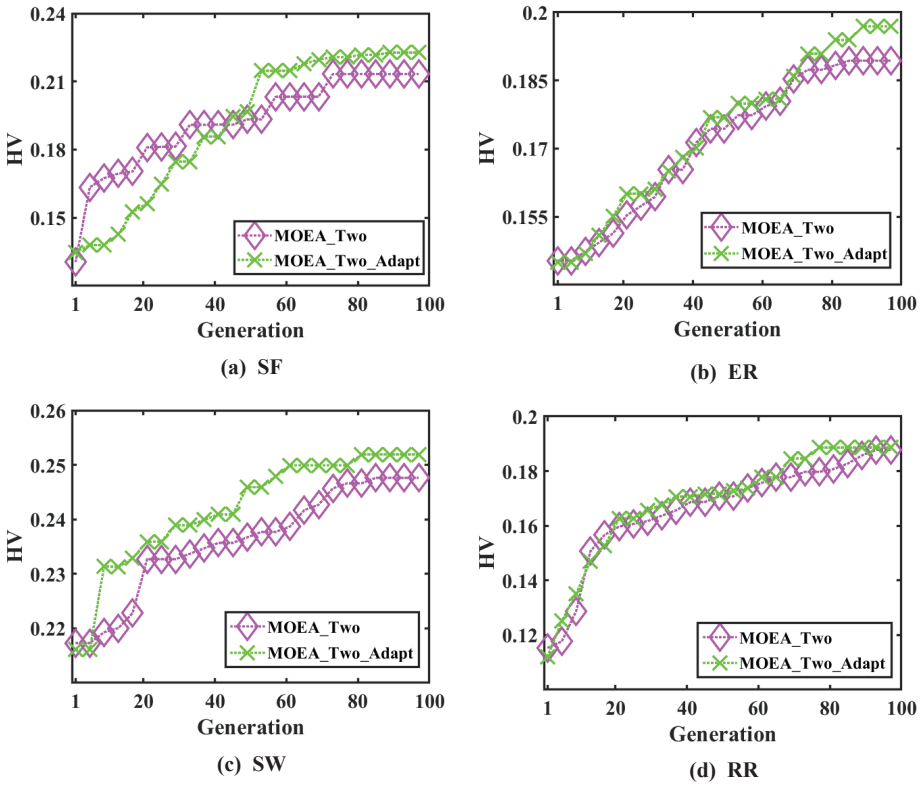

Figure 6. HV values of MOEA_Two and MOEA_Two_Adapt in four types of network.

Here, we compare the previously obtained MOEA_Two method's HV curve with the adaptively updating method. Similarly, when evaluating the performance of the method, the higher the HV value, the better it is. From the Figure 6, it can be seen that the adaptively updating method ultimately remains above the other HV curve.

6. Discussions

The method used in this paper is verified on SF, ER, SW, and RR networks. The results of the MOEA_Two under D-S theory, such as non-dominated solutions, HV values, and optimization time, are better than those obtained by the existing method [12]. The MOEA_Two_Adapt is better than the MOEA_Two, but the optimization time is threefold better. Under degree-based node and betweenness-based link attacks, the method in this paper can optimize the network structure with stronger controllability robustness.

In order to further reduce the computational cost and enable the application of multi-objective controllability robustness to large-scale networks, this paper introduces a surrogate model to approximate the controllability robustness in complex networks. Through extensive experiments on large-scale networks with up to 200 nodes and an average degree of four, the accuracy and efficiency of the proposed surrogate model were demonstrated. Next, the MOEA_Two can expand to optimize the network structure of the controllability robustness of large-scale complex networks.

The obtained network structure can be applied to the producer community. At present, blockchain technology is widely utilized to manage the energy exchange between consumers. The continuous practices of blockchain and distributed ledger technology (DLT) are studied in order to optimize a blockchain network for the purpose of system and construction design in the field of continuous energy delivery. By doing so, it can improve the efficiency of energy transaction and reduce the cost of energy delivery [38]. A framework was developed to facilitate both practitioners and researchers creating blockchain networks that are efficiently designed, reproducible, and dependable [39]. If the multi-objective optimization method used in this paper is applied to the design of the blockchain network, this network can enhance the controllability robustness of the network and normally

complete its own work when externally damaged. In addition to the above environment, the method of network structure constructed in this paper can also be applied to the aviation network. When a base station is damaged in the aviation network [12], the whole line will stop. If a network structure that can simultaneously resist multiple attacks is designed, the probability of the network structure being damaged causing paralysis is small. In addition, an important evolution was discovered in the article, as, besides the human network, the network of things is becoming increasingly common. Therefore, there is increasing research on the Internet of Things and various IoT scenarios. This article is the first attempt to investigate abnormal situations in multiple Internet of Things (MIoT) scenarios. In the IoT context, especially in areas such as anomaly detection and attack recognition, network robustness is also taken into account [40].

Convolutional neural network (CNN), a powerful deep learning model for tasks such as image and speech processing, has also been applied to complex network analysis in recent years, particularly for predicting the robustness of complex networks [30,31]. In general, predicting the robustness of complex networks requires the consideration of the following factors: (1) the network topology structure is an important factor that affects the robustness of complex networks. CNNs can predict the robustness of networks by learning the topological structure characteristics of the network. (2) Another important factor that affects the robustness of complex networks is node attributes. CNN can predict the robustness of networks by learning the node attribute characteristics.

7. Conclusions and Future Work

This paper proposes a network design approach focusing on better robustness, allowing the network to remain operational even after nodes and links are attacked simultaneously. As the computational cost of multi-objective robustness optimization is excessively high, surrogate models are used to replace the robustness calculation in the optimization process. This paper further explores the selection of surrogate models, including the mixture between a surrogate model based on the D-S theory and adaptive surrogate models.

In the experiment, five methods were compared in the network. From the MOEA_Two experimental results, on the one hand, the HV value of SF was 0.1929, taking 58.65 h; the HV value of ER was 0.1721, taking 72.34 h; the HV value of SW was 0.1721, taking 36.54 h; and the HV value of RR was 0.1671, taking 45.57 h. MOEA_Two methods from HV value, non-dominated solutions, and time consuming situation can help the evolutionary algorithm obtain a network structure with better controllability robustness. In SF, ER, and SW networks, the selected surrogate models are RBF and IDW. In RR networks, the selected surrogate models are IDW and Kriging.

On the other hand, the non-dominated solutions obtained under the MOEA_Two are compared with the solution set obtained under the MOEA_Two_Adapt during the optimization process. Under the MOEA_Two_Adapt, the HV value of SF was 0.1939, the HV value of ER was 0.1766, the HV value of SW was 0.2439, and the HV value of RR was 0.1765. These HV values and non-dominated solutions are better than those of MOEA_Two. As such, a network structure with better controllability robustness can be obtained under MOEA_Two_Adapt. However, from the perspective of time consumption, the MOEA_Two_Adapt optimization time is approximately three-fold that of MOEA_Two.

This paper optimizes the network structure under the multi-objective controllability robustness of small-scale complex networks. In future work, this method can be used to optimize large-scale networks and connectivity robustness. This paper used different methods under the same algorithm framework for comparison. Next, we will study the comparison between different metaheuristics algorithm frameworks.

Author Contributions: J.N., Conceptualization, Formal analysis, Investigation, Methodology, Writing—original draft preparation; Z.Y., Formal analysis, Visualization; J.L., Investigation, Project administration, Supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the National Natural Science Foundation of China (No. 62002249).

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to acknowledge the tutor and students for their valuable feedback on the draft which improved the flow and quality of the work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, D.; Shao, Q.; Liu, Z.; Yu, W.; Chen, C.P. Ridesourcing behavior analysis and prediction: A network perspective. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 1274–1283. [CrossRef]
2. Erdős, P.; Rényi, A. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci.* **1960**, *5*, 17–60.
3. Newman, M.E.; Watts, D.J. Renormalization group analysis of the small-world network model. *Phys. Lett. A* **1999**, *263*, 341–346. [CrossRef]
4. Goh, K.I.; Kahng, B.; Kim, D. Universal behavior of load distribution in scale-free networks. *Phys. Rev. Lett.* **2001**, *87*, 278701. [CrossRef] [PubMed]
5. Lou, Y.; Yang, D.; Wang, L.; Tang, C.B.; Chen, G.R. Controllability Robustness of Henneberg-growth Complex Networks. *IEEE Access* **2022**, *10*, 5103–5114. [CrossRef]
6. Chen, G.R.; Lou, Y.; Wang, L. A comparative study on controllability robustness of complex net-works. *IEEE Trans. Circuits Syst. II Express Briefs* **2019**, *66*, 828–832. [CrossRef]
7. Lou, Y.; Wang, L.; Chen, G.R. Toward stronger robustness of network controllability: A snapback network model. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2018**, *65*, 2983–2991. [CrossRef]
8. Lou, Y.; Xie, S.; Chen, G. Searching better rewiring strategies and objective functions for stronger controllability robustness. *IEEE Trans. Circuits Syst. II Express Briefs* **2020**, *68*, 2112–2116. [CrossRef]
9. Lou, Y.; Li, J.L.; Li, S.; Deng, H. Research Progress on controllability and robustness of complex networks. *Acta Autom. Sin.* **2021**, 2374–2391. [CrossRef]
10. Wang, S.; Liu, J. Community robustness and its enhancement in interdependent networks. *Appl. Soft Comput.* **2019**, *77*, 665–677. [CrossRef]
11. Wang, S.; Liu, J.; Jin, Y. Surrogate-assisted robust optimization of large-scale networks based on graph embedding. *IEEE Trans. Evol. Comput.* **2019**, *24*, 735–749. [CrossRef]
12. Wang, S.; Liu, J.; Jin, Y. A Computationally Efficient Evolutionary Algorithm for Multiobjective Network Robustness Optimization. *IEEE Trans. Evol. Comput.* **2021**, *25*, 419–432. [CrossRef]
13. Buesser, P.; Daolio, F.; Tomassini, M. Optimizing the robustness of scale-free networks with simulated annealing. In *Adaptive and Natural Computing Algorithms*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 167–176. [CrossRef]
14. Zhou, M.; Liu, J. A memetic algorithm for enhancing the robustness of scale-free networks against malicious attacks. *Physica A* **2014**, *410*, 131–143. [CrossRef]
15. Tang, X.; Liu, J.; Zhou, M. Enhancing network robustness against targeted and random attacks us-ing a memetic algorithm. *Europhys. Lett.* **2015**, *111*, 38005. [CrossRef]
16. Wang, S.; Liu, J. A Multi-Objective Evolutionary Algorithm for Promoting the Emergence of Co-operation and Controllable Robustness on Directed Networks. *IEEE Trans. Netw. Sci. Eng.* **2018**, *5*, 92–100. [CrossRef]
17. Sun, C.L.; Li, Z.; Jin, Y.W. Model-assisted computationally time-consuming evolutionary high-dimensional multi-objective optimization. *Acta Autom. Sin.* **2022**, *48*, 1119–1128. [CrossRef]
18. Hallabia, H.; Hamam, H. An Enhanced Pansharpening Approach Based on Second-Order Poly-nomial Regression. In Proceedings of the 2021 International Wireless Communications and Mo-bile Computing (IWCMC), Harbin City, China, 28 June–2 July 2021; pp. 1489–1493. [CrossRef]
19. Guo, D.; Jin, Y.; Ding, J.; Chai, T. Heterogeneous Ensemble-Based Infill Criterion for Evolution-ary Multiobjective Optimization of Expensive Problems. *IEEE Trans. Cybern.* **2018**, *49*, 1012–1025. [CrossRef]
20. Chi, M. A Comparative Study Of Improved Kriging Furthermore, Distance Power Inverse Surface Interpo-lation. In Proceedings of the 2020 13th International Conference on Intelligent Computation Tech-nology and Automation (ICICTA), Xi'an, China, 24–25 October 2020; pp. 1–3. [CrossRef]
21. Müller, J.; Piché, R. Mixture surrogate models based on Dempster-Shafer theory for global opti-mization problems. *J. Glob. Optim.* **2011**, *51*, 79–104. [CrossRef]
22. Dempster, A.P. Upper and lower probabilities induced by a multivalued mapping. *Ann. Math. Stat.* **2008**, *38*, 325–339. [CrossRef]
23. Shafer, G. *A Mathematical Theory of Evidence*; Princeton University Press: New York, NY, USA, 1976; ISBN 0-608-0250.
24. Lou, Y.; Wang, L.; Chen, G.R. Structural Robustness of Complex Networks: A Survey of A Poste-riori Measures. *IEEE Circuits Syst. Mag.* **2023**, *4*, 12–35. [CrossRef]
25. Liu, Y.Y.; Slotine, J.J.; Barabasi, A.L. Controllability of complex networks. *Nature* **2011**, *473*, 167–173. [CrossRef]

26. Yuan, Z.; Zhao, C.; Di, Z.; Wang, W.X.; Lai, Y.C. Exact controllability of complex networks. *Nat. Commun.* **2013**, *4*, 2447. [CrossRef] [PubMed]
27. Lou, Y.; Wu, R.Z.; Li, J.L.; Wang, L.; Tang, C.B.; Chen, G. A Learning Convolutional Neural Network Approach for Complex Network Robustness Prediction. *IEEE Trans. Cybern.* **2022**, *4*, 1–4. [CrossRef] [PubMed]
28. Lou, Y.; He, Y.D.; Wang, L.; Li, X.; Chen, G. Predicting Network Controllability Robustness: A Convolutional Neural Network Approach. *IEEE Trans. Cybern.* **2020**, *52*, 4052–4063. [CrossRef] [PubMed]
29. Lou, Y.; He, Y.D.; Wang, L.; Li, X.; Chen, G. Knowledge-Based Prediction of Network Controllability Robustness. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 5739–5750. [CrossRef]
30. Lou, Y.; Wu, R.Z.; Li, J.L.; Wang, L.; Wang, L.; Tang, C.B.; Chen, G. A Convolutional Neural Network Approach to Predicting Network Connectedness Robustness. *IEEE Trans. Netw. Sci. Eng.* **2021**, *8*, 3209–3219. [CrossRef]
31. Lou, Y.; Wu, R.; Li, J.L.; Wang, L.; Tang, C.B.; Chen, G. Classification-Based Prediction of Network Connectivity Robustness. *Neural Netw.* **2023**, *157*, 136–146. [CrossRef]
32. Wang, D.; Cui, P.; Zhu, W. Structural deep network embedding. In Proceedings of the 22nd August ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1225–1234. [CrossRef]
33. Grover, A.; Leskovec, J. node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 855–864. [CrossRef]
34. Zhang, Q.; Li, H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731. [CrossRef]
35. Zhou, M.; Liu, J. A two-phase multi-objective evolutionary algorithm for enhancing the robustness of scale-free networks against multiple malicious attacks. *IEEE Trans. Cybern.* **2017**, *47*, 539–552. [CrossRef]
36. Xia, B.; Ren, Z.; Koh, C.S. Selecting proper Kriging surrogate model for optimal design of elec-tromagnetic problem. In Proceedings of the 9th IET International Conference on Computation in Electromagnetics, London, UK, 31 March–1 April 2014; pp. 1–2. [CrossRef]
37. Wang, S.; Liu, J.; Jin, Y. Robust Structural Balance in Signed Net-works Using a Multiobjective Evolutionary Algorithm. *IEEE Comput. Intell. Mag.* **2020**, *15*, 24–35. [CrossRef]
38. Górski, T. Continuous Delivery of Blockchain Distributed Applications. *Sensors* **2021**, *22*, 128. [CrossRef]
39. Tran, N.K.; Babar, M.A.; Walters, A. A Framework for Automating Deployment and Evaluation of Blockchain Network. *arXiv* **2022**, arXiv:2203.10647. [CrossRef]
40. Cauteruccio, F.; Cinelli, L.; Corradini, E.; Terracina, G.; Ursino, D.; Virgili, L.; Savaglio, C.; Liotta, A.; Fortino, G. A framework for anomaly detection and classification in Multiple IoT scenarios. *Future Gener. Comput. Syst.* **2021**, *114*, 322–335. [CrossRef]