SEE Fault Sensitivity Analysis and Security Reinforcement Design for FPGA Circuits Based on Complex Network

MENG-RU WANG, TAO ZHANG, JIN-BO WANG, SHAN ZHOU, AND LU KONG
Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences, Beijing 100094, China
Corresponding author: Meng-Ru Wang (wangmengru@csu.ac.cn)

ABSTRACT There may exist many high-energy particles in spacecraft, so the FPGA circuits design needs corresponding sensitivity analysis and security reinforcement of anti-SEE (Single-event effects). However, it may be impractical to perform such measures to all modules of FPGA circuits due to limited resources. To identify the key modules which have a vital impact on the design and operation of FPGA circuits in spacecraft, this paper presents a novel scheme based on complex network for modeling the modules considering both the circuit functional structure and signal interaction relationship between modules. First, complex networks like MSN (Module Structure Network) and SFN (Signal Flow Network) are established to identify modules by treating each module as a node, and indicators including degree centrality (DC), betweenness centrality (BC), clustering coefficient (CC), etc., are calculated. Then, an entropy-weight method (EWM) is utilized to calculate the indicators comprehensively for identifying key modules. Next, network efficiency and sensitivity analysis are performed for failure modes. Finally, a case study is carried out, demonstrating the effectiveness of the proposed scheme for the key module identification. This work provides useful technical support for engineers in spacecraft FPGA circuits design and performance enhancement.

INDEX TERMS Complex network, FPGA circuits, SEE sensitivity analysis, entropy-weight method.

I. INTRODUCTION
Considering the environmental complexity and high cost of space applications, the devices we adopt in the spacecraft should be reconfigurable. FPGA has attracted more and more attention and has been applied to many space programs [1]–[4]. However, there exist many high energy particles in the space environment, and the interaction of high energy particles with electronic components will cause SEEs (Single Event Effects). The SEE happens when the collected fraction of the charge liberated by the high energy particle might be larger than the electric charge stored on a sensitive node [5]. There are many manifestations of SEEs, such as SEU, MEU, SET, SEFI, etc., which will cause damage to the FPGA circuits to different degrees [4]–[6], [6]–[9].

Besides, as the development of semiconductor manufacturing technology, the characteristic size of chips has reduced from micrometer level to nanometer level, so the SEEs on the FPGA circuits are more obvious [10]. According to statistics from the relevant institutions on the anomalous statistics of 39 synchronous satellites launched between 1971 and 1986, there were 1589 faults in total, among which 1129 were caused by irradiation, and the SEE was induced by irradiation. There were 621 SEE faults, accounting for 39% of the total faults, and 55% of the irradiation faults [11]. Therefore, it is of great significance to apply anti-SEE security reinforcement design on FPGA circuits, such as the read back scrubbing, timed scrubbing, TMR (three-mode redundancy) method, etc. Among which TMR is the most widely used. However, the FPGA device resources are limited, so it is impossible to apply TMR on every function module of the FPGA circuits. It is of great significance to develop an effective way to evaluate the FPGA circuit design, so as to seek the most vital module, i.e., the key module in the system. Here the key module means the module closely coupled with other modules functionally. That is, a SEE error happened to this module may be propagated to other modules fast and easily, so that the efficiency of the entire network is greatly reduced. Once the
key modules are identified, the security reinforcement design can be applied to these modules to improve the SEE resistance of the whole FPGA circuit efficiently.

Some scholars have adopted a certain amount of methods on the failure sensitivity analysis and safety design of circuits, such as Cellular Automata (CA), Petri Net, fault tree analysis (FTA) method, etc. Wu constructed the CA model of the circuit by introducing the mathematical tool of cellular automata, analyzed the propagation mechanism of the circuit fault, and assessed the fault-sensitive module in the circuit based on the coupling relationship between modules. However, the traditional fault sensitivity judgment on modules lacks quantitative standards [12]. Alexander applied a modified Petri Net simulation algorithm in digital integrated circuit safety design and solved the issue of the priority choice problem [13]. Georgilakis exploited Petri Nets to simulate the transformer fault diagnosis process with actions to reinforce the design, reduced the time needed for transformer fault diagnosis [14]. Shu used the FTA method on the failure analysis problem of printed circuit board assembly (PCBA) to generate the fault-tree and corresponding nodes, as well as circuit reliability computation [15]. Note that the actual engineering system has high complexity and density which make the FTA and the Petri Net model abstract and analysis difficult.

Complex networks have been developed rapidly based on graph theories. A network can be viewed as a graph consisting of nodes connected by edges according to a certain rule or form, so the nodes and edges do not necessarily have physical meanings in the graphs. Complex network analysis has been applied to many fields for key node identification and efficiency analysis, such as wireless communication system [16]–[18], aviation transportation system [19], [20], community network [21], underwater acoustic sensor network [22], urban transportation system [23], [24], power transmission system [25], [26], etc. Complex network analysis models have been proposed for different systems, and structural characteristics are studied based on these models, such as node degree, betweenness centrality, and clustering coefficient. Then, the whole network vulnerability can be analyzed in different node failure modes.

In terms of the integrated circuit field, the complex network analysis method has been applied widely. Zhou proposed the optimization strategies of circuits design by using the complex network to analyze IMB-PLACE 2.0 benchmark circuits, and simulation data proved that large-scale circuits have small-world characteristics and scale-free characteristics [27]. Fan applied the complex network analysis method to build a cascading failure model based on the flow redistribution, analyzed the system vulnerability and robustness, and thus provided theoretical support for circuit design optimization [28]. Tan used the complex network for physical designs of circuit global similarity tests [29]. Liu established a network model of power electronic circuits and evaluated the impact of each component failure on the reliability of power electronic circuits [30]. For FPGA circuit security analysis and design, Nie analyzed the complex network characteristics of the FPGA IP core circuit network, extracted the circuit network information from the FPGA IP core, and provided a method for security design of the FPGA IP [31].

However, most existing models focus on the FPGA circuit at the netlist layer [31]–[33], which is not directly related to the functional performance of the design. FPGA designers cannot directly seek the possible functional failure of the design through the analysis results. To address this problem, this paper formulates the complex network model in the RTL layer, which is directly related to the design function, so the analysis result can intuitively reflect the FPGA hidden functional dangers.

Second, for modeling the complex circuit network, most researchers focus only on the physical topology [34]–[36]. In fact, FPGA circuits are dynamic systems, which need the interaction of signals between modules. This requires a more proper network model that can reflect the actual operating states of the FPGA circuit.

Third, there have been some related studies concerning the ranking of the node importance [37], [38], but only one indicator is used for ranking. This is insufficient because different indicators reflect different network characteristics. Some studies consider various indicators comprehensively, but the effectiveness needs to be improved. In [39], the average weighted method (AWM) has been used to integrate different indicators, but it is not preferable because the influence of different indicators on the importance of nodes is obviously different in most cases. Besides, the analytic hierarchy process has been used such as the Delphi method [40], but this method depends heavily on the designer experience. Therefore, a more rigorous mathematical method is needed to utilize the factors comprehensively.

Motivated by the above observations, this paper presents an improved approach for FPGA Circuits SEE sensitivity analysis based on the complex network. Thus, the major contributions can be summarized as follows.

1) Modeling and analysis of the FPGA circuits at the RTL layer. Compared to the netlist layer, the RTL layer reflects the functions of FPGA circuits intuitively. With the proposed method, it is easier for designers to reason about possible functional anomalies through the SEE sensitivity module failures, so that more targeted security reinforcement measures can be taken.

2) We consider both the physical topology and the multiplicity or directionality of the signal interaction between modules. A weighted directed complex network modeling method for the FPGA circuits is then proposed which can be applied to all FPGA circuits, including the FPGA IP.

3) We present an entropy weight method (EWM) to consider all the network indicators comprehensively. It determines the weight of each factor based on the data information. This method achieves better module SEE fault sensitivity ranking results than the typical average weighting method and stronger applicability than the Delphi method.
The rest of this paper is organized as follows. Section II describes the network efficiency analysis process for FPGA circuits, as well as the modeling of Module Structure Network (MSN) and Signal Flow Network (SFN). Section III adopts the indicators for key module identification and corresponding calculation. In Section IV, a novel network efficiency analysis method is developed. Comparative case studies are given in Section V. Finally, the conclusions and potential issues are provided in Section VI.

II. MODELING FPGA CIRCUITS USING MSN AND SFN

In this section, we model the FPGA circuits at the RTL layer using the complex network. Both the module structure and the signal flow are considered, and two complex network models are developed: Module Structure Network (MSN) and Signal Flow Network (SFN). By blending the two networks, a novel method is formulated to calculate the network efficiency. The proposed method for the FPGA circuit network efficiency analysis is shown in Figure 1. The overall scheme is composed of four parts: FPGA circuits complex network model formulation, identification of key module (the failure of the key module has a greater impact on the whole network efficiency), module importance ranking, and network efficiency analysis.

First, MSN is developed based on the functional coupling of FPGA circuit modules. Then, SFN is formulated...
by considering the signal interaction frequency and direction between modules. Next, the key module identification indicators (DC, SC, BC, CC) are calculated. To evaluate the importance of modules in SFN more accurately, an entropy-weight method is introduced to determine the weight of each indicator, so that the modules in the SFN can be ranked. Finally, with the ranked key modules, the network efficiency is analyzed under the Selective module failure mode and Random module failure mode, respectively.

A. MODULE STRUCTURE NETWORK (MSN)

In the RTL layer, the target FPGA circuit consists of multiple modules. A module represents a logical entity in a hardware circuit, so each module implements a specific function. All modules are connected to complete the entire FPGA circuit design. Each module can be considered as a node. If there exists a functional interaction between two modules, then we connect two modules with one edge in the MSN [41]. Figure 2 shows a typical MSN. This is based on a simple FPGA circuit design. The main functions of the circuit are 1) receive and execute the instruction from the external system and return the engineering data; 2) output the OC pulse according to the external instructions; 3) control AD chip to collect and send analog quantity periodically. This design contains 8 modules: clock management module, reset module, instruction receiving and distributing module, 1553B instruction processing module, 422 instruction processing module, AD acquisition control module, OC pulse generation module, and data return module. Each module is treated as a node, and if there exists a functional interaction between two nodes (modules), we consider that there is a connection between them. Then, an edge is added between them. The functional topology relationship for the modules is shown in Figure 2. The red nodes (S7, S8) represent the output nodes, which are directly related to the functional appearance of the FPGA circuit. The MSN can be thus expressed as $G_m(V_m, E_m)$, where $V_m$ is the module set in the network, and $E_m$ is a set of connections between modules which is determined by functional interactions. The MSN reflects the basic FPGA circuit function logic and can be used to analyze the SFN further.

B. SIGNAL FLOW NETWORK (SFN)

The SFN is established considering both the signal interaction frequency and direction which can be obtained by the module’s outputs and inputs. For example, node S3 has 36 signals. It receives 12 input signals from node S1, S2, S5, S6, S7 and outputs 24 signals to other nodes S4, S5, S6, S7. As shown in Figure 3, the number of signals that pass from S3 to S4 determines the weight of the edge between S3 and S4.

C. MSN AND SFN CONSIDERATIONS

As described before, the MSN can be improved by using the signal interactions to produce the SFN. The signal interaction relationships are listed in Table 1, and the SFN is established as shown in Figure 4. It is seen that the edge between nodes S3 and S5 is created by R15 and R20. The weight of the edge is just the number of signals passing through node S3 to node S5, i.e., 5, and 2 from S5 to S3. Other edges and weights can be produced similarly. In this way, the Signal Flow Network can be established by considering the module structure, signal direction, and signal frequency.

III. INDICATORS CALCULATION FOR KEY MODULE IDENTIFICATION ANALYSIS USING THE COMPLEX NETWORKS

As demonstrated in II, based on MSN, the SFN can be formulated by considering the signal interactions. Then we can use the complex network principle to calculate the key module identification indicators [41], including the degree centrality (DC), strength centrality (SC), betweenness centrality (BC), clustering coefficient (CC), comprehensive factor (CF).
TABLE 1. Signal interaction description.

| Interaction relationship | Starting node | Ending node | Signal number |
|------------------------|---------------|------------|--------------|
| R1                     | S1            | S2         | 2            |
| R2                     | S1            | S3         | 2            |
| R3                     | S1            | S4         | 2            |
| R4                     | S1            | S5         | 1            |
| R5                     | S1            | S6         | 2            |
| R6                     | S1            | S7         | 1            |
| R7                     | S1            | S8         | 1            |
| R8                     | S2            | S3         | 2            |
| R9                     | S2            | S4         | 2            |
| R10                    | S2            | S5         | 1            |
| R11                    | S2            | S6         | 2            |
| R12                    | S2            | S7         | 1            |
| R13                    | S2            | S8         | 1            |
| R14                    | S3            | S4         | 8            |
| R15                    | S3            | S5         | 5            |
| R16                    | S3            | S6         | 7            |
| R17                    | S3            | S7         | 4            |
| R18                    | S4            | S7         | 12           |
| R19                    | S4            | S8         | 6            |
| R20                    | S5            | S3         | 2            |
| R21                    | S5            | S8         | 7            |
| R22                    | S6            | S3         | 3            |
| R23                    | S6            | S7         | 5            |
| R24                    | S6            | S8         | 6            |
| R25                    | S7            | S3         | 3            |
| R26                    | S7            | S8         | 2            |

These indicators can be used to evaluate the impact of module failures on the FPGA design networks. The details are given below.

A. DEGREE CENTRALITY (DC)

We assume that node $v_i$ denotes the $i$-th node within the SFN, the DC value of node $v_i$ equals to the number of the connections between $v_i$ and other nodes in the MSN. This value describes the function interaction tightness of a node with its functional neighbor nodes. For example, the DC value of node S1 is 7, as shown in Figure 4, which means that 7 nodes i.e., S2, S3, S4, S5, S6, S7, S8, are connected to Node S1 directly. We define the DC value $k_i$ of a node $v_i$ in the MSN as:

$$k_i = \sum_{j=1}^{N} n_{i,j}$$  \hspace{1cm} (1)

where $N$ is the node number within the MSN and $n_{i,j}$ is a variable of 0 or 1, and the rules are as follows:

$$n_{i,j} = \begin{cases} 
1, & \text{there exists signal interaction between node } i \text{ and node } j \\
0, & \text{no signal interaction between node } i \text{ and node } j 
\end{cases}$$

If a node in the MSN has more edges, it will be given a larger DC value. Thus, in the FPGA circuit, the reachability of a module can be evaluated by its DC value.

B. STRENGTH CENTRALITY (SC)

Within the SFN, the edge weight determines the importance of the edge in the SEE propagation network. That is, a bigger edge weight corresponds to the more frequent use of the edge for signal interaction. A SEE fault can be easily propagated to other nodes through the edge. In this paper, the signal interaction frequency of a module, together with its neighbor, can be represented by its SC value. It can be calculated as:

$$S_i = \sum_{j=1}^{N} w_{i,j}$$ \hspace{1cm} (2)

where $S_i$ is the SC value of the $i$-th node $v_i$; $w_{i,j}$ is the weight of the edge between $i$-th node and $j$-th node. An edge’s weight $w_{i,j}$ between node $v_i$ and $v_j$ in the SFN is the number of signals that interact between nodes $v_i$ and $v_j$. $N$ is the number of nodes that have connections with node $v_i$ in the SFN. Taking node S4 as an example, the SC value is 30 ($S_4 = 2+2+8+12+6$), as shown in Figure 4. On the other hand, strengthening the safety protection design for modules with higher SC values is beneficial to reduce the probability of fault propagation and enhance the reliability of the whole design.

C. BETWEENNESS CENTRALITY (BC)

The node-betweenness of node $v_i$ is defined as:

$$b_i = \sum_{j \neq i \neq l} \frac{L_{j,l}(i)}{L_{j,l}}$$ \hspace{1cm} (3)

where $L_{j,l}$ is the number of all existing shortest paths from node $v_j$ to node $v_l$. These paths have the minimum number of edges, and $L_{j,l}(i)$ the ones that actually pass through node $v_i$ within $L_{j,l}$ [41]. For example, there are 21 shortest paths between any two nodes in the network in Figure 5(a): S5-S4-S3-S2-S1, S5-S4-S3-S2-S6, S5-S4-S3-S2-S7, S5-S4-S3-S2-S4-S3-S2-S1, S4-S3-S2-S6, S4-S3-S2-S7, S4-S3-S2-S4-S3-S2-S1, S3-S2-S6, S3-S2-S7, S3-S2-S3-S2-S1, S3-S2-S4-S3-S2-S6, S3-S2-S4-S3-S2-S7, S3-S2-S3-S2-S1, S3-S2-S4-S3-S2-S6, S3-S2-S4-S3-S2-S7, S3-S2-S3-S2-S1, S2-S6, S2-S7, S6-S2-S7, S6-S2-S1, S7-S2-S1, while there are 10 paths that do not pass through node S4. The BC value of node S4 is $(21-10)/21 = 0.52$. Thus, the influence of a node in the network can be evaluated by its BC value [41]. In this study, the BC is related to the shortest path that the SEE fault propagates from one node to the other one, i.e., from one module to another module. If the SEE failures happen in the SFN modules with high BC values, the severer effect on the whole design will occur. That is, the failure may be propagated to more modules with higher possibility. Thus, compared with indicators DC and SC, the BC evaluates the network in the perspectives of connectivity.
D. CLUSTERING COEFFICIENT (CC)

In the network, two neighbors of a node may or may not be connected themselves. This is characterized by the concept of clustering [41]. Let \( v_i \) be a node with \( k_i \) edges connected to other \( k_i \) nodes, called neighbors of node \( v_i \). \( k_i \) is the DC value of node \( v_i \), as stated in III.A. It is easy to verify that there are at most \( k_i(k_i + 1)/2 \) edges between these \( k_i \) neighbors and the node \( v_i \). Let \( E_i \) be the number of the actual edges existing between these \( k_i \) nodes. Then the ratio between the actual and the possible numbers of edges in the cluster of these \( k_i \) nodes can be defined as the clustering coefficient of node \( v_i \), as:

\[
c_i = \frac{2E_i}{k_i(k_i + 1)}
\]  

(4)

As shown in Figure 5(b), where node S2 is connected with nodes S1, S3, S5, S6, the maximum number of edges between each pair is \( 4(4+1)/2 = 10 \). However, there actually exist 6 connections: S2-S1, S2-S3, S2-S5, S2-S6, S5-S6, S3-S6. The other 4 connections do not exist but can be created with the actual connections. For example, the direct connection between S1 and S6 does not exist, but can be reached by the route: S1-S2-S6. So, with equation (4) the CC of S4 is \( 6/10 = 0.6 \). Therefore, CC represents the influence of modules in the network from the perspective of clustering, and the average CC value of all nodes represents the clustering of the entire network. However, if nodes have the same connections in the network, their CC values are equal. Thus, sometimes only using the CC to rank the importance of the modules is not sufficient.

E. COMPREHENSIVE FACTOR (CF)

As stated before, the DC, SC, BC, CC are indicators used to quantize the node influence on the complex network. However, there are significant limitations using a single indicator to evaluate the importance of a node. For example, in terms of degree centrality, not all-important nodes have a large number of connected edges. The nodes with the same number of connected edges in the network may not have the same importance. For example, a bridge node has a small degree value but plays a vital role in the network. Thus, we need to find a method to evaluate the node importance in the network considering various indicators comprehensively. The weighting method is a commonly used one. The typical way to choose weights is the average weighting method which allocates all indicators the same weight. Actually, in specific areas, we can use the analytic hierarchy process such as the Delphi method in selecting weights of different indicators. In this section, we propose the entropy-weight method (EWM) to calculate the weight for each indicator, and then integrate the four indicators for identification analysis on the key modules of the network.

The concept of Entropy was first introduced by Shannon to information theory [42] and has been widely used. The basic idea of EWM is to determine weight according to the size of the indicator variability.

In general, if the information entropy of an indicator is smaller, it indicates that this indicator deserves more variation, and provides more information to play a greater role in the comprehensive evaluation, so the weight assigned to this indicator will be greater. Conversely, a greater indicator information entropy corresponds to the less indicator variation and provides less information. The less information provided by the indicator corresponds to the smaller the role it plays on the comprehensive evaluation, with smaller weight assigned to this indicator.

The key steps of the EWM are as follows:

First, we normalize the indicators as:

\[
Z_{i,DC} = \frac{Z_{i,DC}^{DC} - Z_{\text{min}}^{DC}}{Z_{\text{max}}^{DC} - Z_{\text{min}}^{DC}}
\]  

(5)

where \( Z_{i,DC}^{DC} \) is the calculated DC value of node \( v_i \); \( Z_{\text{min}}^{DC} \) is the minimum of DC values, \( Z_{\text{max}}^{DC} \) is the maximum of DC values, \( Z_{i,DC}^{DC} \) is the normalized value of node \( v_i \). Similarly, we can obtain the normalized value of the other three indicators (SC, BC, CC) as:

\[
Z_{i,SC} = \frac{Z_{i,SC}^{SC} - Z_{\text{min}}^{SC}}{Z_{\text{max}}^{SC} - Z_{\text{min}}^{SC}}
\]

\[
Z_{i,BC} = \frac{Z_{i,BC}^{BC} - Z_{\text{min}}^{BC}}{Z_{\text{max}}^{BC} - Z_{\text{min}}^{BC}}
\]

\[
Z_{i,CC} = \frac{Z_{i,CC}^{CC} - Z_{\text{min}}^{CC}}{Z_{\text{max}}^{CC} - Z_{\text{min}}^{CC}}
\]  

(6)

Second, we formulate the information entropy of each indicator.

According to the definition of information entropy theory, the information entropy of DC can be calculated as:

\[
E_{DC} = - \ln(n) - \sum_{i=1}^{n} p_{i,DC} \ln p_{i,DC}
\]  

(7)

where \( p_{i,DC} = \frac{Z_{i,DC}^{DC}}{\sum_{i=1}^{n} Z_{i,DC}^{DC}} \). Note that if \( p_{i,DC} = 0 \), we have \( \lim_{p_{i,DC} \to 0} p_{i,DC} \cdot \ln(p_{i,DC}) = 0 \).
Finally, we determine the weights of the four indicators as:

\[ W_{DC} = (1 - E_{DC})/[4 - (E_{DC} + E_{SC} + E_{BC} + E_{CC})] \]
\[ W_{SC} = (1 - E_{SC})/[4 - (E_{DC} + E_{SC} + E_{BC} + E_{CC})] \]
\[ W_{BC} = (1 - E_{BC})/[4 - (E_{DC} + E_{SC} + E_{BC} + E_{CC})] \]
\[ W_{CC} = (1 - E_{CC})/[4 - (E_{DC} + E_{SC} + E_{BC} + E_{CC})] \]  

(8)

The CF of node \( v_i \) can be calculated as:

\[ c_f = W_{DC}Z_i^{DC} + W_{SC}Z_i^{SC} + W_{BC}Z_i^{BC} + W_{CC}Z_i^{CC} \]  

(9)

IV. EFFICIENCY ANALYSIS ON THE FPGA DESIGN NETWORK

FPGA design reliability can be obtained by analyzing the network performance under different failures. In this paper, we choose the selective module failure mode and random module failure for case studies. For the selective module failure mode, designers usually choose modules to impose fault according to certain rules or their experience to analyze the design reliability. Herein, we impose faults to modules according to the sequences obtained with (9). For the random module failure mode, we select modules randomly and assess the network efficiency. The network efficiency \( R \) calculation methods can be described as:

\[ E = \frac{2}{M(M - 1)} \sum_{i<j}^{M} \frac{1}{d_{ij}} \]  

(10)

\[ R = \frac{E}{E_0} \]  

(11)

where \( M \) is the number of remaining modules in the network that operates normally after other modules failure. \( d_{ij} \) denotes the number of edges in the shortest path between node \( v_i \) and node \( v_j \).

V. CASE STUDY AND ANALYSIS

A. BACKGROUND

To validate the proposed methodology for SEE failure sensitivity module identification, we perform case studies on network efficiency analysis for an FPGA circuit in TianGong-2 Project. The object includes 113 modules, and there are various signal interactions between modules. We model each module as a node and establish the MSN with 113 nodes, as shown in Figure 6. Based on the signal interaction (frequency and direction), the SFN can be established with the same number of nodes as the MSN. The number of edges is created according to the method described in Section II.B. Thus, the indicators used in key module identification in Section III can be calculated.

B. DEGREE CENTRALITY (DC)

As described in III.A, the node DC value can be calculated by (1). Here we decompose DC into DC-out and DC-in. The DC-out of node \( v_i \) indicates the number of nodes connected to the output of node \( v_i \) and the DC-in indicates the number of nodes connected to the input of node \( v_i \).

C. STRENGTH CENTRALITY (SC)

Figure 8(a) demonstrates the SC values of nodes calculated by (2). It is shown in Figure 8(b) that only a few nodes have high SC values. Besides, about 7.07% of the total modules have SC values larger than 25, and 7 modules (about 6.1% of the total modules) have DC-in values larger than 20. The modules are totally different.

D. BETWEENNESS CENTRALITY (BC)

The BC values calculated by (3) are shown in Figure 9. It is revealed that most BC values are small. Only 4 modules have relatively large BC values (24, 61, 98, 112) which account for 3.54% of the total modules. In other words, these 4 modules
are SEE fault-sensitive and contribute more significantly to the efficiency of the SFN.

**E. CLUSTERING COEFFICIENT (CC)**
The CC value of nodes can be calculated by (4). Figure 10 shows all the modules CC values. We also can get that the SFN average CC value is 0.2663, demonstrating relatively low aggregation characteristics of the SFN. This means that the modules within the FPGA circuit network are not connected very closely. Besides, from the calculation results, we can figure out that some nodes have the same CC value, so it is inappropriate to merely use CC value for the node importance ranking.

**F. COMPREHENSIVE FACTOR (CF)**
The CF value of node $v_i$ can be calculated with the method proposed in III.E, that is, using (9). We normalize the five factors of the nodes and then use the Entropy-weight method to calculate $W_{DC_0}$, $W_{DC_1}$, $W_{SC}$, $W_{BC}$, $W_{CC}$. The results are listed in Table 2.

Table 3 shows the importance rankings of the top 10 modules based on their CF values. It also shows the rankings in terms of DC, SC, BC, and CC, respectively.

It is obvious that the ranking results are different for different indicators. The reason is that different indicators correspond to different characters of the network. However, the CF combines the functions of DC, SC, BC, and CC comprehensively. In other words, the CF not only considers the connection topology but also reflects the signal interaction, direction, frequency, original modules, and destination modules. Therefore, using the CF to assess the module importance as well as SEE fault sensitivity, is a more preferable and reliable way.

**G. NETWORK EFFICIENCY ANALYSIS**
Two failure models are applied on the network, respectively: the selective failure mode and the random one. In case of

### Table 2. Indicators’ weight values.

| Parameter | Value          |
|-----------|----------------|
| $W_{DC_0}$| 0.227913407   |
| $W_{DC_1}$| 0.103209247   |
| $W_{SC}$  | 0.172535531   |
| $W_{BC}$  | 0.417395054   |
| $W_{CC}$  | 0.078946761   |

### Table 3. Ranking of top 10 modules.

| Ranking | Modules (DC0 value) | Modules (DC1 value) | Modules (SC value) | Modules (BC value) | Modules (CC value) | Modules (CF value based on EWM) | Modules (CF value based on AWM) |
|---------|---------------------|---------------------|--------------------|--------------------|--------------------|---------------------------------|---------------------------------|
| 1       | 112(1)              | 28(1)               | 113(1)             | 112(1)             | 37(1)              | 112(0.7765778)                 | 112(0.5658178)                 |
| 2       | 61(0.8571429)       | 113(1)              | 112(0.6721311)     | 24(0.22791428)     | 88(0.654088)       | 24(0.4110488)                  | 113(0.4496297)                 |
| 3       | 24(0.8367347)       | 65(0.92)            | 28(0.483606557)    | 98(0.13707561)     | 44(0.654088)       | 113(0.3710792)                 | 28(0.4042581)                  |
| 4       | 98(0.8367347)       | 102(0.92)           | 102(0.483606557)   | 61(0.12908864)     | 81(0.654088)       | 61(0.3528352)                  | 24(0.3982744)                  |
| 5       | 62(0.5510264)       | 32(0.88)            | 50(0.483606557)    | 32(0.06946563)     | 40(0.555256)       | 98(0.34338755)                 | 102(0.3926887)                 |
| 6       | 25(0.5306122)       | 69(0.8)             | 24(0.483606557)    | 106(0.0668298)     | 70(0.518194)       | 28(0.2699841)                  | 65(0.3875050)                  |
| 7       | 91(0.5306122)       | 106(0.8)            | 65(0.450819672)    | 89(0.06082293)     | 42(0.433962)       | 102(0.2661668)                 | 32(0.3729332)                  |
| 8       | 101(0.5306122)      | 50(0.68)            | 61(0.450819672)    | 12(0.05881484)     | 65(0.418412)       | 32(0.26381176)                 | 106(0.362194)                  |
| 9       | 17(0.48979592)      | 39(0.68)            | 42(0.43442623)     | 39(0.054327869)    | 79(0.415094)       | 106(0.25597227)                | 69(0.3526858)                  |
| 10      | 27(0.48979591)      | 33(0.64)            | 79(0.43442623)     | 52(0.0507314)      | 77(0.407068)       | 25(0.24498168)                 | 61(0.3478977)                  |
selective failure, modules fail in descending order of CF value (calculated by EWM), DC-in value, DC-out value, SC value, CC value, BC value, respectively. In case of random failure, modules fail in random order generating with a random function. The relative network efficiency can be calculated with (11). Figure 11 shows the calculation result under two failure modes. The green line denotes the selective failure mode (CF), and the blue one denotes the random failure mode. It can be seen that in the selective mode, the relative network efficiency R drops below 0.5 when 20% of modules fail, and almost zero when 60% of modules fail. In the random mode, almost all modules fail, the network efficiency drops to 0.

Besides, comparing with R based on CF value (calculated by EWM), R under the individual five indicators (DC-in, DC-out, SC, CC, BC) drops much slower, which indicates that using a comprehensive factor to rank the importance of nodes does achieve better results than using a single factor.

Further, we simulate the changes of R for failing the first 10 nodes, seconds 10 nodes..., eleventh 10 nodes, as shown in Figure 12. We can see that the change of the first 10 nodes does have a greater impact on the whole network efficiency. The following 9 groups of node failures (11-20, 21-30, ...,100-110.) have less effect on R with a very slight difference.

Finally, we compare the average-weight method (AWM) with the proposed entropy-weight method (EWM). For the AWM, we assign every indicator equal weight value as 0.2. Then the CF of all nodes can be calculated, and the module
SEE fault-sensitivities can be ranked. Then we apply the fault to modules in ascending order of comprehensive factor calculated by AWM. The results are shown in Figure 13, demonstrating that EWM achieves a more accurate module SEE fault sensitivities ranking than AWM.

VI. CONCLUSION

The paper presents a novel scheme based on the complex network for modeling the FPGA circuits at the RTL layer. Two network models of MSN and SFN are established. Both circuit topology and dynamic signal interactions are considered. Once the module importance is ranked, the FPGA circuits network efficiency can be analyzed under selective failure modes and random failure modes. We apply the proposed method on an FPGA circuit in TianGong-2 Project for validation. The key modules can be identified using the CF with considering the reachability, connectivity, and local influence comprehensively. Simulation results show that the proposed method performs better in the network efficiency analysis. It can quickly identify the most important modules in the design. Moreover, the FPGA circuits designers can apply the security reinforcement design on circuits with limited logic resources. The FPGA circuits verifiers can select SEE fault-injection objects more conveniently which makes the fault-injection experiment more effective and efficient.

REFERENCES

[1] J. J. Rodríguez-Andina, M. D. Valdés-Peña, and M. J. Moure, “Advanced features and industrial applications of FPGAs—A review,” IEEE Trans. Ind. Informat., vol. 11, no. 4, pp. 853–864, Aug. 2015, doi: 10.1109/TII.2015.2431223.

[2] N. Battezzati, L. Sterpone, and M. Violante, Reconfigurable Field Programmable Gate Arrays for Mission-Critical Applications. London, U.K.: Springer, 2011.

[3] I. Villalta, U. Bidarte, J. Gomez-Cornejo, J. Lazaro, and C. Cuadrado, “Dependability in FPGAs, an overview,” in Proc. Conf. Design Circuits Integr. Syst. (DCIS), Nov. 2015, pp. 1–6.

[4] S. Tamimi, Z. Ebrahimi, B. Khaleghi, and H. Asadi, “An efficient SRAM-based reconfigurable architecture for embedded processors,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 38, no. 3, pp. 466–479, Mar. 2019.

[5] W. He, Y. Wang, K. Xing, and J. Yang, “Modeling and experimental study of soft error propagation based on cellular automaton,” Math. Problems Eng., vol. 1, pp. 1–11, Jul. 2016.

[6] E. Petersen, Single Event Effects in Aerospace. Piscataway, NJ, USA: IEEE Press, 2011.

[7] N. Rezzak, J.-J. Wang, D. Dsila, and N. Jat, “TID and SEE characterization of Micronsemi’s 4th generation radiation tolerant RTG4 flash-based FPGA,” in Proc. IEEE Radiat. Effects Data Workshop (REDW), Jul. 2015, pp. 211–216.

[8] M. Nicolaidis, Soft Error in Modern Electronic Systems. Boston, MA, USA: Springer, 2011.

[9] T. Li, H. Liu, and H. Yang, “Design and characterization of SEU hardened circuits for SRAM-based FPGA,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 27, no. 6, pp. 1276–1283, Jun. 2019.

[10] H. Guo, W. Wang, and Y. Luo, “Challenges of single-particle effect research in new microelectronic technology,” Nucl. Technol., vol. 7, no. 33, pp. 538–542, 2010.

[11] C. Wang, “The influence of single-particle effect on the reliability of satellite space operation,” Semicond. Inf., vol. 35, no. 1, pp. 1–8, 1998.

[12] W. Jimei, “Modeling and application of circuit fault propagation based on cellular automata,” M.S. thesis, Dept. Inf. Sci. Technol., Donghua Univ., Shanghai, China, 2008.
H. L. Wang and H. Shan, “Analysis of a small analog electronic circuits based on the complex network,” J. Harbin Univ. Sci. Technol., vol. 238, no. 9, pp. 2256–2260, 2006.

R. F. I. Cancho, C. Janssen, and R. V. Solé, “Topology of technology graphs: Small world patterns in electronic circuits,” Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 64, no. 4, pp. 322–333, Sep. 2001.

H. Dingding, “Complex network property and reliability analysis of wireless receiver circuit,” Electron. Meas. Technol., vol. 31, no. 9, pp. 1–5, 2008.

D. Q. Wei, X. S. Luo, and B. Zhang, “Analysis of cascading failure in complex power networks under the load local preferential redistribution rule,” Phys. A, Stat. Mech. Appl., vol. 465, pp. 13–24, Jan. 2017.

J. Harbin Univ. Sci. Technol., vol. 238, no. 9, pp. 2256–2260, 2006.

MENG-RU WANG received the B.S. and M.S. degrees in control theory and control engineering from Beijing Jiaotong University, Beijing, China, in 2010 and 2013, respectively. She is currently pursuing the Ph.D. degree with the University of Chinese Academy of Sciences. She is currently an Engineer of the Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences. She works on high-reliability software testing, space FPGA circuit security design, and technologies for fault injection.

TAO ZHANG received the B.S. and Ph.D. degrees in power engineering and control from Tsinghua University, Beijing, China, in 1995 and 2000, respectively. He is the Chief Engineer of the Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences, Beijing. His research interests include testing and verification technology for high-reliability software, analysis and design method for high-reliability electronic information systems, the simulation of complex systems, the overall design of space application systems, and virtual reality techniques.

JIN-BO WANG received the B.S. and Ph.D. degrees in computer application technology from the University of Chinese Academy of sciences. He is a Researcher and the Director of the Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences, Beijing. His research interests include testing and verification technology for high-reliability software, analysis and design method for high-reliability testing systems, and information security.

SHAN ZHOU received the Ph.D. degree. She is a Senior Engineer of the Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences. She was engaged in the research and practice of aerospace key FPGA/SOPC verification methods and the research and applications of intelligent test platforms. At the same time, she is also responsible for many large-scale test systems such as the flexible integration test system of the space station. Her research interests include high-reliability software testing and flexible intelligent simulation testing technology.

LU KONG received the M.S. degree in microelectronics and solid-state electronics from the Beijing University of Technology, Beijing, in 2010. Since 2010, she has been an Engineer of the Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences. Her research interests include high-reliability software testing, flexible intelligent simulation testing technology, the practice of aerospace key FPGA/SOPC verification methods, and the research and applications of intelligent test platforms.

***