Research Article

Matrix Approach to Formulate and Search k -ESS of Graphs Using the STP Theory

Jumei Yue ¹, Yongyi Yan ², and He Deng ²

¹College of Agricultural Equipment Engineering, Henan University of Science and Technology, Luoyang, Henan, China

²College of Information Engineering, Henan University of Science and Technology, Luoyang, Henan, China

Correspondence should be addressed to Jumei Yue; yjm@mail.nankai.edu.cn

Received 16 April 2021; Accepted 26 June 2021; Published 6 July 2021

Academic Editor: Xian-Ming Gu

Copyright © 2021 Jumei Yue et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, the structure of graphs in terms of k -externally stable set (k -ESS) is investigated by a matrix method based on a new matrix product, called semitensor product of matrices. By defining an eigenvector and an eigenvalue of the node subset of a graph, three necessary and sufficient conditions of k -ESS, minimum k -ESS, and k -kernels of graphs are proposed in a matrix form, respectively. Using these conditions, the concepts of k -ESS matrix, minimum k -ESS matrix, and k -kernel matrix are introduced. These matrices provide complete information of the corresponding structures of a graph. Further, three algorithms are designed, respectively, to find all these three structures of a graph by conducting a series of matrix operation. Finally, the correctness and effectiveness of the results are checked by studying an example. The proposed method and results may offer a new way to investigate the problems related to graph structures in the field of network systems.

1. Introduction

A graph is a figure consisting of nodes and edges; nodes represent things; an edge linking two nodes represents the relationship between the two things. A graph usually describes a specific binary relation between things. And, graph theory is a delightful playground for the exploration of proof techniques in discrete mathematics, and its results have applications in many areas of computing, social, and natural sciences [1]. Interest in graphs and their applications continues to grow rapidly, largely due to the usefulness of graphs as models for computation and optimization [2].

An externally stable set (ESS) of a graph is a set of nodes that any node outside of the set is linked with a node in the set. A k -externally stable set (k -ESS) of a graph is an advanced version of an ESS that any node outside of the set is linked with a node in the set by a path of length k . A kernel of a graph is a set of nodes, that is, both an ESS and internally stable set (ISS). An internally stable set of a graph is a set of nodes where any two nodes are not linked with each other. Similarly, a k -kernel of a graph is an advanced form of a kernel, that is, a k -kernel of a graph is both a k -ESS and a

k -internally stable set (k -ISS), where a k -ISS is a set of nodes where any two nodes are not linked with each other by a path of length k . ISS, ESS, kernel, k -ISS, k -ESS, and k -kernels are important modes of describing the structure of graphs.

The structure of graphs is a lasting hot topic in graph theory and has been widely discussed in various areas. In the field of novelty detection, Mohammadi et al. [3] used kernel dependence technique to characterize the statistical dependency of the random variable. Previous results mainly use the density assessment method to determine the dependency of random variables and are unable to address high dimension data. This approach uses a kernel of the graph to characterize the dependency and is robust and suitable to problems of arbitrary high dimension data. Bernshtejn and Bozhenyuk [4] proposed the concepts of fuzzy internally stable and fuzzy externally stable sets of a fuzzy directed graph and defined their degrees. By generalizing the approach to discriminate maximal internally stable sets and minimal externally stable sets of an ordinary graph, they developed a method of determining fuzzy internally stable and fuzzy externally stable sets with maximum stability degree. In the analysis of logical networks, Liu et al. [5] applied the graph structure to detect some special states such as

observable and unobservable states of Boolean networks. Compared with the methods based on computer algorithms, the advantage of such graph structure-based method is that the observable and unobservable states of Boolean networks can be determined directly. Besides, the structures of graphs are used to many areas such as fault detection [6], network sentiment analysis [7], and decision theory [8].

Matrix can be applied in almost every field of natural science. Although traditional matrix multiplication is widely used, it has limitations. In terms of application scope, it can only be used when two matrices meet the equal dimension condition, and it does not meet the commutative law. In terms of the types of problem-solving, it is mainly used to solve linear or bilinear problems. For multilinear and nonlinear problems, it is almost powerless. With the development of computer and nonlinear sciences, a new method able to deal with high-dimensional array is expected.

A new matrix multiplication, called semitensor product of matrices (STP), has been developed in recent years [9]. STP extends the traditional multiplication of matrices to any two matrices, which overcome the dimension limitation and keep almost all the important properties of the traditional multiplication. Furthermore, due to the introduction of the swap matrix, STP has some exchangeable properties, called pseudoexchangeability, which overcomes the weakness of commutativity of traditional matrix multiplication to some extent. STP is now well established and has been successfully applied in many fields [10–17].

Based on STP, a matrix method of studying the graph structure has been developed. Wang et al. [18] investigated the maximum stable sets and node coloring problems of graphs by using the STP. Several new results and algorithms are presented. The results are useful to consider the network control problems such as group consensus of multiagent systems. Since then, the matrix method has been used to investigate many problems of graphs. Xu et al. [19] continued to consider the coloring of robust graphs by transforming the problem into a type of optimal problem in the framework of the STP and used the matrix method to design an algorithm that can find all the colorings of robust graphs. Further, the timetabling problem is solved by the presented results. In [20, 21], Xu et al. pushed the matrix method to a new level and studied the coloring of fuzzy graphs and the conflict-free coloring problem. Using the matrix method, Zhong et al. [22] investigated the minimum stable sets and cores of graphs; several new theoretic results and algorithms are proposed. These new results provide a way to further consider the optimization of minimum stable sets of graphs and control problems of Boolean control networks such as pinning control. In the hypergraph area, Meng et al. [23] applied the matrix method to investigate the hypergraph coloring. A necessary and sufficient condition of the algebraic inequality form is given, and an algorithm of finding all the coloring schemes of a hypergraph with specific numbers of colors is designed. Yue et al. [24] considered the k -track scheduling problem by investigating the k -ISS of graphs. Two necessary and sufficient conditions and schedulable conditions are developed by using the matrix method.

The merits of this matrix method in addressing graph problems are concise mathematical description and clear results. Generally speaking, the problems related to the graph structure can be formulated by matrices, usually called structure matrices. A structure matrix contains all the structure information of a particular class of the graph structure. The embodying of complete structure information provides us a chance to design algorithms to find all the structures of a graph.

As far as we know, there has been no report on using the matrix method to investigate the k -ESS and k -kernel structures of graphs. Generally speaking, there are three kinds of ways to consider the graph structures: based on graph theory, mathematical programming and computer algorithm. Most results obtained by these methods, especially by the computer-based algorithms, are inconvenient to explain the structures mathematically and cannot theoretically guarantee to find all the k -ESSes and k -kernels of a graph.

Motivated by the above defects of the existing results and the advantages of the matrix method of the investigating graph structure, the aims of this paper are to use the matrix method to consider several more complicated structures of graphs, including the k -ESS, minimum k -ESS, and k -kernel, and to design algebraic algorithms to find all these structures of a given graph.

The main contributions of this paper contain two parts, theoretic results and algorithms. The theoretic part includes algebraic formulations of the k -ESS, minimum k -ESS, and k -kernels, which are presented as several necessary and sufficient conditions. The algorithm part consists of three algorithms of finding k -ESS, minimum k -ESS, and k -kernels, respectively. These results can find all these structures of a graph and may provide a new way to consider the structure problems of network systems such as reductions of Boolean networks, dynamic evolution game systems, and finite state machine networks. A disadvantage of the presented algorithms is the memory burden when solved by computers for graphs with more than 20 nodes.

The remainder of this paper is organized as follows. Section 2 gives some preliminaries on the mathematical tool (STP) and graphs. In Section 3, the main results of this paper are presented, including theoretic and algorithm parts. In Section 4, we analyze the computation complexity of the presented algorithms and discuss the differences between the presented results and existing ones. Section 5 is devoted to examine the correctness of the results. Finally, a concluding remark is given in Section 6.

The following are the notations used in this paper.

$1_n/0_n$: the $n \times 1$ vector of all element being 1/0

$\text{col}_i(A)$: the i th column of matrix A

$\text{col}(A)$: the set of all the columns of matrix A

δ_n^i : the i th column of the identity matrix I_n

Δ_n : the set $\{\delta_n^1, \dots, \delta_n^n\}$

$\delta_n[i_1, \dots, i_k]$: the compactness form of matrix $[\delta_n^{i_1}, \dots, \delta_n^{i_k}]$

$|S|$: the cardinality of set S

2. Preliminaries

This section gives some necessary preliminaries on the theoretic basis and research object, namely, STP and k -ESS of graphs, respectively.

Definition 1 (see [9]).

- (1) Let X be a row vector of dimension np and Y be a column vector with dimension p . Split X into p equal-size blocks as $X_1, \dots, X_p$, which are $1 \times n$ row vectors. The semitensor product of X and Y , $X \ltimes Y$, is defined as

$$\begin{cases} X \ltimes Y = \sum_{i=1}^p X_i y_i \in R^n, Y^T \ltimes X^T = \sum_{i=1}^p y_i (X_i)^T \in R^n, \end{cases} \quad (1)$$

where $y_i \in R$ is the i th component of Y .

- (2) For any matrices $A \in R_{m \times n}$ and $B \in R_{p \times q}$, if either n is a factor of p , say, $nt = p$, or p is a factor of n , say, $n = pt$, then the semitensor product of A and B , denoted by $C = \{C^{ij}\} = A \ltimes B$, is defined as follows: C consists of $m \times q$ blocks, and each block is defined as

$$C_{ij} = A^i \ltimes B_j, \quad i = 1, 2, \dots, m \text{ and } j = 1, 2, \dots, q, \quad (2)$$

where A^i is the i th row of A and B_j is the j th column of B .

Remark 1. Definition 1 is a special case of STP; the generalized definition is defined as $A \ltimes B := (A \otimes I_{s/n}) (B \otimes I_{s/p})$, where s is the least common multiple of n and p and $\otimes$ is the Kronecker product. The STP used in this paper is the form of Definition 1.

Remark 2. If $n = p$, Definition 1 reduces to the traditional product of matrices. Therefore, STP is a generalization of the traditional multiplication of matrices. In this paper, the matrix multiplication is defaulted to the STP, and the symbol $\ltimes$ is omitted except for special emphasis needed.

Definition 2 (see [9]). A swap matrix $W_{[m,n]}$ is a matrix of size $mn \times mn$; its rows and columns are labelled by index (i, j) . The rows and columns are organized by the order of multiindex $\text{Id}[j, i; n, m]$ and $\text{Id}[i, j; m, n]$, respectively. The element at position $[(I, J), (i, j)]$ is

$$W_{((I,J),(i,j))} = \delta_{i,j}^{I,j} = \begin{cases} 1, & I = i \text{ and } J = j, \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

Proposition 1 (see [9]). Let $X \in R^m$ and $Y \in R^n$; then,

$$\begin{aligned} W_{[m,n]} \ltimes X \ltimes Y &= Y \ltimes X, \\ W_{[n,m]} \ltimes Y \ltimes X &= X \ltimes Y. \end{aligned} \quad (4)$$

Definition 3 (see [25]). The following matrices $E_{d_latter}(m, n)$ and $E_{d_former}(m, n)$ are called dummy operators:

$$\begin{aligned} E_{d_latter}(m, n) &= \begin{bmatrix} I_n & \dots & I_n \\ & & \end{bmatrix}_m, \\ E_{d_former}(m, n) &= \begin{bmatrix} I_m & \dots & I_m \\ & & \end{bmatrix}_n. \end{aligned} \quad (5)$$

Epecially, $E_{d_latter}(2, 2) := E_{d_latter}$ and $E_{d_former}(2, 2) := E_{d_former}$.

Proposition 2 (see [25]). Let $u \in \Delta_m$ and $v \in \Delta_n$ be two logical vectors; then,

$$\begin{aligned} E_{d_latter}(m, n) \ltimes u \ltimes v &= v, \\ E_{d_former}(m, n) \ltimes W_{[n,m]} \ltimes u \ltimes v &= u. \end{aligned} \quad (6)$$

Proposition 3 (see [18]). Let x be a 2-valued logical variable; then,

$$M_n x = \bar{x}, \quad (7)$$

where $M_n = \delta_2[2, 1]$.

Definition 4 (see [26]). A graph $G = (V, E)$ is a mathematical structure composed of two finite sets V and E . The elements of V are called nodes, and the elements of E are called edges. Each edge has a set of one or two nodes associated to it, which are called its endpoints.

An edge is said to join its endpoints. A node joined by an edge to a node v_i is said to be a neighbor of v_i . In this paper, v_i is thought to be a neighbor of itself denoted by $N(v_i) = \{v_j \in V \mid (v_j, v_i) \in E\}$, the neighbor set of v_i . A graph $P = (V', E')$ is called a subgraph of $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E$.

Definition 5 (see [26]). A subgraph $P = (V', E')$ of $G = (V, E)$ with the form $V' = \{v_{i_1}, v_{i_2}, \dots, v_{i_k}\}$ and $E' = \{(v_{i_1}, v_{i_2}), (v_{i_2}, v_{i_3}), \dots, (v_{i_{k-1}}, v_{i_k})\}$ is called a path of G . The number of the edges of a path from v_i to v_j is called the length of the path, denoted as $d(v_i, v_j)$. $d(v_i, v_i) = 0$ if and only if $v_i = v_j$.

Definition 6 (see [18]). A set S of nodes of G is called a k -externally stable set (k -ESS) of G if for any node $v_i \notin S$, there is a node $v_j \in S$ such that $v_j \in N(v_i)$. A k -ESS S is called a minimum k -ESS if any proper subset of S is not a k -ESS. A minimum k -ESS with the least number of nodes is called an absolutely minimum k -ESS; the number of nodes of an absolutely minimum k -ESS is called the k -externally stable number of G , denoted by $\alpha(G)$.

Definition 7 (see [27]). For two matrices $A = [a_{ij}]_{p \times q}$ and $B = [b_{ij}]_{q \times r}$, their Boolean product $C = A(\times_{\mathcal{B}})B = [c_{ij}]_{p \times r}$ is defined as

$$c_{ij} = \bigvee_{l=1}^q (a_{il} \wedge b_{lj}), \quad (8)$$

where $\vee$ and $\wedge$ are the operations of “logic Or” and “logic And”, respectively. The Boolean power of A is defined as

$$\begin{aligned} A^{0_{\mathcal{B}}} &= A, \\ A^{k_{\mathcal{B}}} &= A^{k-1_{\mathcal{B}}} (\times_{\mathcal{B}}) A, \quad k = 1, 2, \dots \end{aligned} \quad (9)$$

Definition 8 (see [25]). For graph $G = (V, E)$, $V = \{v_1, v_2, \dots, v_n\}$, the matrix $A = [a_{ij}]$ is called its adjacency matrix (AM), where

$$a_{ij} = \begin{cases} 1, & \text{if } (v_i, v_j) \in E, \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

The k -adjacency matrix (k -AM) of G is defined as follows:

$$A^{[k]} = A \vee A^{2_{\mathcal{B}}} \vee \dots \vee A^{k_{\mathcal{B}}}, \quad (11)$$

where

- (i) $A^{i_{\mathcal{B}}}$ denotes the Boolean power of A
- (ii) $A \vee B = [a_{ij} \vee b_{ij}]_{m \times n}$

3. Main Results

In this section, we present the main results of the paper, including theoretic and algorithm results. The theoretic part contains matrix formulations of k -ESS, minimum k -ESS, and k -kernels of graphs. The algorithm part consists of algebraic algorithms of finding all the k -ESSes, minimum k -ESSes, and k -kernels of a graph.

3.1. On k -ESS

Definition 9 (see [18]). Given a node subset $S \subseteq V$ of graph $G = (V, E)$, the eigenvector of S , denoted by $V_S = [x_1, x_2, \dots, x_n]$, is

$$x_i = \begin{cases} 1, & x_i \in S, \\ 0, & x_i \notin S. \end{cases} \quad (12)$$

Using STP, we define the eigenvalue of a node subset of a graph. An eigenvalue and the corresponding node subset are uniquely determined by each other.

Definition 10. Let $V_S = [x_1, x_2, \dots, x_n]$ be the eigenvector of a node subset $S \subseteq V$ of graph $G = (V, E)$, and let us define $y_i = [x_i, 1 - x_i]^T$, $i = 1, 2, \dots, n$. The eigenvalue of S is defined as

$$Y_S = \bowtie_{i=1}^n y_i. \quad (13)$$

Remark 3. Because that y_i ($i = 1, 2, \dots, n$) can be uniquely obtained by calculating $y_i = S_i^n \bowtie Y_S$, where S_i^n is given in (22), and that the y_i ($i = 1, 2, \dots, n$) uniquely determines Y_S , the eigenvalue Y_S of S uniquely determines the eigenvector V_S , and vice versa. Thus, if Y_S (or V_S) is given, then V_S (or Y_S) can be determined. Consequently, the node subset S is obtained.

Proposition 4. Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$, $V = \{v_1, v_2, \dots, v_n\}$. The length of a path from node v_i to v_j is less than or equal to k , i.e., $d(v_i, v_j) \leq k$, if and only if $a_{ij}^{[k]} = 1$.

Proof. Consider $A^{l_{\mathcal{B}}} = [a_{ij}^l]$, $1 \leq l \leq k$. It follows from (4) that

$$a_{ij}^l = \bigvee_{i_1, \dots, i_{l-1}} (a_{ii_1} \wedge a_{i_1 i_2} \wedge \dots \wedge a_{i_{l-1} j}). \quad (14)$$

It is obvious that $a_{ij}^l = 1$ if and only if there are $l-1$ subscripts $i_1, i_2, \dots, i_{l-1}$ such that

$$a_{ii_1} = a_{i_1 i_2} = \dots = a_{i_{l-1} j} = 1. \quad (15)$$

Combining (15) with (5), we get that there is an edge between v_i and v_{i_1} , v_{i_1} and v_{i_2} , $\dots$, and $v_{i_{l-1}}$ and v_j , respectively. Therefore, $a_{ij}^l = 1$ if and only if v_i and v_j are linked by a path of length $d(v_i, v_j) \leq l$. $d(v_i, v_j) = l$ if and only if $i, i_1, i_2, \dots, i_{l-1}, j$ are different from each other.

For $A^{[k]} = [a_{ij}^{[k]}]$, $a_{ij}^{[k]} = a_{ij}^1 \vee a_{ij}^2 \vee \dots \vee a_{ij}^k = 1$ holds if and only if there exists an l ($1 \leq l \leq k$) such that $a_{ij}^l = 1$. And, $a_{ij}^l = 1$ if and only if v_i and v_j are linked by a path of length $d(v_i, v_j) \leq l$; thus, $a_{ij}^{[k]} = 1$ if and only if there is a path of length $d(v_i, v_j) \leq k$ from node v_i to v_j . $\square$

Theorem 1. Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$, $V = \{v_1, v_2, \dots, v_n\}$. G has a k -ESS if and only if there exists a $1 \leq j \leq 2^n$ such that

$$\text{col}_j(M) = 1_n, \quad (16)$$

where

$$M = \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_n \end{bmatrix}, \quad (17)$$

$$M_i = J \bowtie \sum_{j=1}^n (a_{ij}^{[k]} Q_j), \quad i = 1, 2, \dots, n,$$

$$J = [1, 0],$$

$$Q_j = (E_{d_latter})^{n-j} \bowtie W_{[2, 2^{n-j}]} \bowtie (E_{d_latter})^{j-1}.$$

Proof. We first prove the necessity. Suppose S is a k -ESS of G . Let $V_S = [x_1, x_2, \dots, x_n]$ be the eigenvector of S . From Proposition 4 and the definition of k -ESS of a graph, we get that, for v_i not in S , there exists a j ($1 \leq j \leq n$) such that

$$a_{ij}^{[k]} = 1, \quad x_j = 1. \quad (18)$$

Equation (18) implies that

$$x_i \vee \left(\bigvee_{j=1}^n a_{ij}^{[k]} x_j \right) = 1. \quad (19)$$

Using (10) and (19), it can be re-expressed as follows:

$$\bigvee_{j=1}^n a_{ij}^{[k]} x_j = 1. \quad (20)$$

Because a_{ij} and x_j are either 1 or 0, (20) is equivalent to

$$\sum_{j=1}^n a_{ij}^{[k]} x_j = 1. \quad (21)$$

Let $y_j = [x_j, 1 - x_j]^T$, $j = 1, 2, \dots, n$. It follows from (6) that

$$\begin{aligned} y_j &= (E_{d_latter})^{n-j} \bowtie y_{j+1} \bowtie y_{j+2} \bowtie \dots \bowtie y_n \bowtie y_j \\ &= (E_{d_latter})^{n-j} \bowtie W_{[2, 2^{n-j}]} \bowtie y_j \bowtie y_{j+1} \bowtie \dots \bowtie y_n \\ &= (E_{d_latter})^{n-j} \bowtie W_{[2, 2^{n-j}]} \bowtie (E_{d_latter})^{j-1} \bowtie y_1 \bowtie \dots \bowtie y_n := Q_j \bowtie Y_S, \end{aligned} \quad (22)$$

where $Q_j = (E_{d_latter})^{n-j} \bowtie W_{[2, 2^{n-j}]} \bowtie (E_{d_latter})^{j-1}$.

It is evident that

$$\begin{aligned} x_j &= J \bowtie y_j, \\ a_{ij}^{[k]} x_j &= a_{ij}^{[k]} J \bowtie y_j = a_{ij}^{[k]} J \bowtie Q_j \bowtie Y_S. \end{aligned} \quad (23)$$

By (21), we know that

$$\sum_{j=1}^n (a_{ij}^{[k]} x_j) = J \bowtie \sum_{j=1}^n (a_{ij}^{[k]} Q_j) \bowtie Y_S := M_i \bowtie Y_S, \quad (24)$$

where $M_i = J \bowtie \sum_{j=1}^n (a_{ij}^{[k]} Q_j)$.

Therefore, the following statements are equivalent to each other for $i \in \{1, 2, \dots, n\}$:

- (i) $\sum_{j=1}^n a_{ij}^{[k]} x_j = 1$,
- (ii) $M_i \bowtie Y_S = 1$,
- (iii) There exists an l : $\text{col}_l(M_i) = 1_n$,
- (iv) There exists a $1 \leq j \leq 2^n$: $\text{col}_j(M_i) = 1_n$.

Note that (21) is equivalent to the following equations:

$$\left\{ \sum_{j=1}^n a_{1j}^{[k]} x_j = 1, \sum_{j=1}^n a_{2j}^{[k]} x_j = 1, \dots, \sum_{j=1}^n a_{nj}^{[k]} x_j = 1. \right. \quad (26)$$

Equations (26) are solvable if and only if there exists a $1 \leq j \leq 2^n$ such that

$$\text{col}_j(M) = 1_n, \quad (27)$$

where

$$M = \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_n \end{bmatrix}. \quad (28)$$

Obviously, (21) is equivalent to (26); then, the necessity is proved.

We next show the sufficiency. Because the derivation process from equations (18) to (26) is reversible

mathematically, according to Proposition 4 and the definition of k -ESS of a graph, the sufficiency holds. The proof is then completed.

Theorem 1 gives the condition that a graph contains a k -ESS. Based on the condition, we further establish a rule to judge whether a given node subset is a k -ESS or not. $\square$

Theorem 2. Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$. For a given $S \subseteq V$, suppose that the eigenvector and eigenvalue of S are $V_S = [x_1, x_2, \dots, x_n]$ and $Y_S = \delta_{2^n}^k$, respectively. Then, S is a k -ESS of G if and only if

$$\text{col}_k(M) = 1_n, \quad (29)$$

where

$$\begin{aligned} M &= \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_n \end{bmatrix}, \\ M_i &= J \bowtie \sum_{j=1}^n (a_{ij}^{[k]} Q_j), \quad i = 1, 2, \dots, n, \\ J &= [1, 0], \\ Q_j &= (E_{d_latter})^{n-j} \bowtie W_{[2, 2^{n-j}]} \bowtie (E_{d_latter})^{j-1}. \end{aligned} \quad (30)$$

Proof. Comparison of (29) with (16) suggests that it is only needed to prove that

$$\text{col}_k(M) = 1_n, \quad (31)$$

that is,

$$\text{col}_k(M_i) = 1, \quad i = 1, 2, \dots, n. \quad (32)$$

It follows from (25) that $M_i \bowtie Y_S = 1$ is equivalent to there exists a $1 \leq j \leq 2^n$ such that $\text{col}_j(M_i) = 1$. Consequently, we only need to show that k satisfies the condition.

Taking notice of the sizes of M_i and Y_S , they are of dimension 1×2^n and $2^n \times 1$, respectively. In such a situation, STP coincides with the traditional product of matrices. It is clear that $M_i \bowtie Y_S$ is just $\text{col}_k(M_i)$.

Therefore, $M_i \bowtie Y_S = 1$ if and only if $\text{col}_k(M_i) = 1$. This proves (32) and completes the proof of Theorem 2. $\square$

Remark 4. The matrix M of (16) or (29) gives a complete picture for all of the k -ESSes of a graph. We thus propose the following definition.

Definition 11. The matrix M in (16) or (29) is called the k -ESS matrix of graph G .

According to Remark 3 and the proof of Theorem 2, the following Corollary 1 is clearly true.

Corollary 1. Let M be the k -ESS matrix of graph $G = (V, E)$. Then, there is a one-to-one correspondence between the k -ESSes of G and the solutions of the following equation:

$$M \times_{i=1}^n y_i = 1_n, \quad (33)$$

where $y_i = [x_i, 1 - x_i]^T$, $x_i \in D = \{0, 1\}$, is a 2-valued variable assigned to each vertex $v_i \in V$.

Corollary 1 provides a way to obtain all the k -ESSes of a given graph.

Algorithm 1. Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$. Assign each node $v_i \in V$ a 2-valued variable $x_i \in D = \{0, 1\}$ and define $y_i = [x_i, 1 - x_i]^T$. To find all the k -ESSes and minimum k -ESSes of G , one can take the following procedure.

Step 1. Compute the k -ESS matrix M of G .

Step 2. Check if there is a column being 1_n in $\text{col}(M)$. If not, there is no k -ESS in G and then the algorithm ends. Otherwise, construct the set

$$K = \{l | \text{col}_l(M) = 1_n\}. \quad (34)$$

Step 3. For each l in K , compute $y_i = S_i^n \times \delta_{2^n}^l$, $i = 1, 2, \dots, n$, where

$$\begin{cases} S_1^n = \delta_2 \left[\underbrace{1 \dots 1}_{2^{n-1}} \underbrace{2 \dots 2}_{2^{n-1}} \right], \\ S_2^n = \delta_2 \left[\underbrace{1 \dots 1}_{2^{n-2}} \underbrace{2 \dots 2}_{2^{n-2}} \underbrace{1 \dots 1}_{2^{n-2}} \underbrace{2 \dots 2}_{2^{n-2}} \right], \\ \vdots \\ S_n^n = \delta_2 \left[\underbrace{12 \dots 12}_{2^{n-1}} \right]. \end{cases} \quad (35)$$

Step 4. Choose $y_i = \delta_2^1$ and construct a set $S_l = \{v_i | y_i = \delta_2^1\}$. S_l is a k -ESS of G . All the k -ESSes of G are $\{S_l | l \in K\}$.

Step 5. The k -externally stable number of G is $\alpha(G) = \min_{l \in K} |S_l|$. All the minimum k -ESSes of G are $\zeta = \{S_l | |S_l| = \alpha(G)\}$.

Remark 5. Algorithm 1 produces all the k -ESSes of a graph. Naturally, the k -ESSes of some characteristics can also be obtained. For instance, each of the nodes outside a k -ESS is connected with the k -ESS once or exactly k times, etc. In a word, Algorithm 1 provides us a picture of the structure of graphs by producing all the k -ESSes.

3.2. On Minimum k -ESS

Theorem 3. Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$. And, let $V_S = [x_1, x_2, \dots, x_n]$ and $Y_S = \times_{i=1}^n y_i = \delta_{2^n}^k$ be the eigenvector and eigenvalue of a subset $S \subseteq V$, where $y_i = [x_i, 1 - x_i]^T$. Then, S is a minimum k -ESS of G if and only if

$$\text{col}_k(M) = 0_n, \quad (36)$$

where

$$M = JQ_i - n^2 J^{n-1} M_n E_{d_latter} \sum_{j=1}^n (\bar{a}_{ij}^{[k]}), \quad (37)$$

$$J = [1, 0],$$

$$Q_i = (E_{d_latter})^{n-i} W_{[2, 2^{n-i}]} (E_{d_latter})^{i-1},$$

$$M_n \text{ is given in Proposition 3; } \bar{a}_{ij}^{[k]} = 1 - a_{ij}^{[k]}.$$

Proof. We first show the necessity. Let $V_S = [x_1, x_2, \dots, x_n]$ be the eigenvector of the minimum k -ESS S . There are two cases to be discussed.

Case (i): $x_i = 0$. In this situation, it follows from (21) that

$$\sum_{\substack{j=1 \\ j \neq i}}^n a_{ij}^{[k]} x_j = 1. \quad (38)$$

Case (ii): $x_i = 1$. We can prove that

$$\sum_{\substack{j=1 \\ j \neq i}}^n a_{ij}^{[k]} x_j = 0. \quad (39)$$

If (39) is false, i.e.,

$$\sum_{\substack{j=1 \\ j \neq i}}^n a_{ij}^{[k]} x_j = 1, \quad (40)$$

then according to Theorem 1, the node subset corresponding to $V_S = [x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n]$ is a k -ESS too, i.e.,

$$S' = (v_1, \dots, v_{i-1}, v_{i+1}, \dots, v_n), \quad (41)$$

is a k -ESS of G .

This is in contradiction with that $S = (v_1, \dots, v_i, \dots, v_n)$ is a minimum k -ESS of G . Thus, (39) is true. We then get that

$$x_i = \sum_{j=1}^n \left(\bar{a}_{ij}^{[k]} \bigwedge_{\substack{j=1 \\ j \neq i}}^n \bar{x}_j \right), \quad (42)$$

where $\bar{a}_{ij}^{[k]} = 1 - a_{ij}^{[k]}$.

It follows from (22) that

$$x_i = J y_i = J (E_{d_latter})^{n-j} W_{[2, 2^{n-j}]} (E_{d_latter})^{j-1} Y_S = J Q_i Y_S, \quad (43)$$

where $Q_i = (E_{d_latter})^{n-j} W_{[2, 2^{n-j}]} (E_{d_latter})^{j-1}$.

On the contrary, using Propositions 2 and 3, we have

$$\begin{aligned}
\sum_{j=1}^n \left(\bar{a}_{ij}^{[k]} \bigwedge_{\substack{j=1 \\ j \neq i}}^n \bar{y}_j \right) &= \sum_{j=1}^n \left(\bar{a}_{ij}^{[k]} \bigwedge_{\substack{j=1 \\ j \neq i}}^n (M_n y_j) \right) = \sum_{j=1}^n \left(\bar{a}_{ij}^{[k]} M_n \bigwedge_{\substack{j=1 \\ j \neq i}}^n y_j \right) \\
&= n M_n \sum_{j=1}^n \left(\bar{a}_{ij}^{[k]} \bigwedge_{\substack{j=1 \\ j \neq i}}^n y_j \right) = n M_n \sum_{j=1}^n \left(\bar{a}_{ij}^{[k]} E_{d_latter} y_i \bigwedge_{\substack{j=1 \\ j \neq i}}^n y_j \right) = n M_n \sum_{j=1}^n \left(\bar{a}_{ij}^{[k]} E_{d_latter} \bigwedge_{j=1}^n y_j \right) \\
&= n M_n \sum_{j=1}^n (\bar{a}_{ij}^{[k]} E_{d_latter} Y_s) = n^2 M_n E_{d_latter} \sum_{j=1}^n (\bar{a}_{ij}^{[k]}) Y_s.
\end{aligned} \tag{44}$$

Submitting (44), (43), and $x_j = J y_j$ into (42), we get

$$J Q_i - n^2 J^{n-1} M_n E_{d_latter} \sum_{j=1}^n (\bar{a}_{ij}^{[k]}). \tag{45}$$

The necessity is proved.

Sufficiency: suppose (36) is true; then, according to the reversibility of the derivation process from equations (42) to (44), the eigenvector of S , $V_s = [x_1, x_2, \dots, x_n]$, meets the conditions of (38) and (39).

If $x_1 = x_2 = \dots = x_n = 1$, then

$$\sum_{j=1}^n a_{ij}^{[k]} x_j \geq \sum_{j=1}^n a_{ij}^{[k]} = 1, \tag{46}$$

thus S is a k -ESS of G .

If there exists a $x_i = 0$, then

$$\sum_{j=1}^n a_{ij}^{[k]} x_j = \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij}^{[k]} x_j = 1. \tag{47}$$

It follows from (26) that S is a k -ESS of G .

Next, we show that S is a minimum k -ESS. If we delete any vertex, say x_i , from S , denote the new subset by S' ; then, the eigenvector of S' is $V_{S'} = [x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n]$. Because $v_i \in S$ and $x_i = 1$, we get from (39) that

$$\sum_{\substack{j=1 \\ j \neq i}}^n a_{ij}^{[k]} x_j = 0, \tag{48}$$

and then,

$$\text{col}_j(M) = 0_n. \tag{49}$$

This suggests that the subset $V_{S'} = [x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n]$ is unsatisfied with condition (9). Thus, S' is not a k -ESS of G . Consequently, S is a minimum k -ESS of G . This proves the sufficiency, and the proof of Theorem 3 is completed.

Using the minimization of pseudo-Boolean functions (PBFs), the problem of finding all the minimum k -ESSes of a graph can also be solved. The following proposition is needed. $\square$

Proposition 5 (see [28]). For a PBF $f(x_1, x_2, \dots, x_n)$ under the constraints,

$$f_j(x_1, x_2, \dots, x_n) = 0, \quad j = 1, 2, \dots, m, \tag{50}$$

where f_j are nonnegative integer-valued PBFs. Use S^+ and S^- to denote the sum of positive and negative coefficients of f , respectively. Construct the following PBF:

$$\begin{aligned}
F(x_1, x_2, \dots, x_n) &= f(x_1, x_2, \dots, x_n) \\
&\quad + (S^+ - S^- + 1) \sum_{j=1}^m f_j(x_1, x_2, \dots, x_n).
\end{aligned} \tag{51}$$

The following statements hold.

- (i) Suppose $(x_1^*, x_2^*, \dots, x_n^*)$ is a local minimum point of PBF f with constraint (50); then, $(x_1^*, x_2^*, \dots, x_n^*)$ is also a local minimum point of PBF F .
- (ii) Suppose $(x_1^*, x_2^*, \dots, x_n^*)$ is a local minimum point of PBF F and $F(x_1^*, x_2^*, \dots, x_n^*) \leq S^+$; then $(x_1^*, x_2^*, \dots, x_n^*)$ is also a local minimum point of PBF f with constraint (50).

Theorem 4. Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$. For a given subset $S \subseteq V$, suppose its eigenvector and eigenvalue are $V_S = [x_1, x_2, \dots, x_n]^T$ and $Y_S = \kappa_{i=1}^n y_i = \delta_{2^n}^k$, respectively, where $y_i = [x_i, 1 - x_i]^T$. Then, S is a minimum k -ESS of G if and only if the k th component of M is a minimum one in the set of nonnegative components, where

$$M = J \left(\sum_{i=1}^n Q_i + (n+1) \sum_{j=1}^n \sum_{i=1}^n (a_{ij}^{[k]} Q_i) \right), \tag{52}$$

in which $J = [1, 0]$ and $Q_i = (E_{d_latter})^{n-i} W_{[2, 2^{n-i}]} (E_{d_latter})^{i-1}$.

Proof. It follows from the proof of Theorem 1 that the eigenvector of a minimum k -ESS of G is a local minimum point of PBF $h(x_1, x_2, \dots, x_n) = \sum_{i=1}^n x_i$ under constraint (20). Applying Proposition 5, here, $S^+ = n$ and $S^- = 0$, to the minimization problem, we only need to investigate the following minimization problem:

$$f(x_1, x_2, \dots, x_n) = \left(\bigvee_{i=1}^n x_i \right) \vee (n+1) \left(\bigvee_{i=1}^n \bigvee_{j=1}^n (a_{ij}^{[k]} x_j) \right). \quad (53)$$

Because both x_i ($i = 1, 2, \dots, n$) and a_{ij} only take values 0 and 1, (53) amounts to the following PBF:

$$f(x_1, x_2, \dots, x_n) = \sum_{i=1}^n x_i + (n+1) \sum_{j=1}^n \sum_{i=1}^n (a_{ij}^{[k]} x_j). \quad (54)$$

According to (22), (23), (24), and (54) can be represented in the following form:

$$\begin{aligned} f(y_1, y_2, \dots, y_n) &= J \left(\sum_{i=1}^n Q_i \right) Y_S + (n+1) \left(\sum_{j=1}^n \sum_{i=1}^n (a_{ij}^{[k]} Q_i) \right) Y_S \\ &= J \left(\sum_{i=1}^n Q_i + (n+1) \sum_{j=1}^n \sum_{i=1}^n (a_{ij}^{[k]} Q_i) \right) Y_S = MY_S, \end{aligned} \quad (55)$$

where

$$\begin{aligned} M &= J \left(\sum_{i=1}^n Q_i + (n+1) \sum_{j=1}^n \sum_{i=1}^n (a_{ij}^{[k]} Q_i) \right), \\ J &= [1, 0], \\ Q_i &= (E_{d_latter})^{n-i} W_{[2, 2^{n-i}]} (E_{d_latter})^{i-1}, \\ y_i &= [x_i, 1 - x_i]^T, \\ Y_S &= \times_{i=1}^n y_i. \end{aligned} \quad (56)$$

Now, it is easy to see from the eigenvalue of S , $Y_S = \times_{i=1}^n y_i = \delta_{2^n}^k$, that the k -th component of M is the minimum value of PBFs (53) and (54). This completes the proof of Theorem 4.

Similarly to Definition 11, we define the matrix M of (52) as the minimum k -ESS matrix of a graph. $\square$

Definition 12. The matrix M of (52) in Theorem 4 is called the minimum k -ESS matrix of graph G .

Remark 6. The minimum k -ESS matrix of a graph offers us a way to find all the minimum k -ESSes of a graph in a matrix manner. Similarly to Algorithm 1, the following algorithm can produce all the minimum k -ESSes of a given graph.

Algorithm 2. Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of a given graph $G = (V, E)$. Assign each $v_i \in V$ a 2-valued variable $x_i \in D = \{0, 1\}$ and define $y_i = [x_i, 1 - x_i]^T$. To get all the minimum k -ESSes of G , one can take the following steps.

Step 1. Compute the minimum k -ESS matrix of G , denoted by M .

Step 2. Check whether there is a nonnegative component in M ; if not, G has no minimum k -ESS and the algorithm terminates here. Otherwise, set

$$K = \{l | \text{the } l\text{-th component of } M \text{ is a minimum one}\}. \quad (57)$$

Step 3. For each l in K , compute $y_i = S_i^n \times \delta_{2^n}^l$, $i = 1, 2, \dots, n$, where

$$\begin{cases} S_1^n = \delta_2 \left[\frac{1 \dots 1}{2^{n-1}} \frac{2 \dots 2}{2^{n-1}} \right], \\ S_2^n = \delta_2 \left[\frac{1 \dots 1}{2^{n-2}} \frac{2 \dots 2}{2^{n-2}} \frac{1 \dots 1}{2^{n-2}} \frac{2 \dots 2}{2^{n-2}} \right], \\ \vdots \\ S_n^n = \delta_2 \left[\frac{12 \dots 12}{2} \right]. \end{cases} \quad (58)$$

Step 4. Choose $y_i = \delta_2^1$ and construct $S_l = \{v_i | y_i = \delta_2^1\}$. S_l is a minimum k -ESS of G . All the minimum k -ESSes of G are $\{S_l | l \in K, S_l \text{ is produced by Step 3 and Step 4}\}$

3.3. Further Results on k -Kernels. The k -kernels of a graph play a key role in describing the structures of graphs and have a wide application prospect in the network systems [29]. A subset S of nodes of a graph G is called a k -kernel if S is both a k -ESS and a k -internally stable set (k -ISS) of G . Yue and Yan [24] considered the algebraic formulation of k -ISS of graphs in the framework of STP and proposed the matrix form of the necessary and sufficient condition of k -ISS. Integrating the results with those of this paper, we expand the results of the k -ESS and minimum k -ESS to the k -kernels of graphs.

Proposition 6 (see [24]). Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$. There is a k -ISS in G if and only if matrix N contains a column $\text{col}_j(N) = 0_n$, $1 \leq j \leq 2^n$, where

$$N = \begin{bmatrix} N_1 \\ N_2 \\ \vdots \\ N_n \end{bmatrix}, \quad (59)$$

$$N_i = Q \sum_{j=1}^n a_{ij} P_{ij}, \quad i = 1, 2, \dots, n,$$

$$Q = [1, 0, 0, 0],$$

$$P_{ij} = (E_{d_latter})^{n-2} \times W_{[2^j, 2^{n-j}]} \times W_{[2^i, 2^{i-1}]}.$$

Proposition 7 (see [24]). Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$; and, let $V_S = [x_1, x_2, \dots, x_n]$ and $Y_S = \times_{i=1}^n y_i = \delta_{2^n}^k$ be the eigenvector and eigenvalue of a subset $S \subseteq V$, where $y_i = [x_i, x_i]^T$. Then, S is a k -ISS of G if and only if $\text{col}_k(N) = 0_n$, where N is given in Proposition 6.

Based on Propositions 6 and 7 and Theorems 1 and 2, it follows from the definition of the k -kernel that the following results are true.

Corollary 2. Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$. Then, G has a k -kernel if and only if there exists a $1 \leq j \leq 2^n$ such that

$$\begin{cases} \text{col}_j(M) = 1_n, \\ \text{col}_j(N) = 0_n, \end{cases} \quad (60)$$

where M and N are given in Theorem 1 and Proposition 6, respectively.

Corollary 3. Let $A^{[k]} = [a_{ij}^{[k]}]$ be the k -AM of graph $G = (V, E)$. For a given node subset $S \subseteq V$, the eigenvector and eigenvalue are $V_S = [x_1, x_2, \dots, x_n]$ and $Y_S = \otimes_{i=1}^n y_i = \delta_{2^n}^k$, respectively, where $y_i = [x_i, 1 - x_i]^T$. Then, S is a k -kernel of G if and only if

$$\begin{cases} \text{col}_k(M) = 1_n, \\ \text{col}_k(N) = 0_n, \end{cases} \quad (61)$$

where M and N are given in Corollary 2.

Remark 7. It is clear from Corollaries 2 and 3 that the matrices M and N of Corollary 3 provide the complete information of all the k -kernels of a graph. We use them to define the concepts of the k -kernel matrix of a graph.

Definition 13. The k -kernel matrix of the graph G described in Corollary 3 is the following matrix K :

$$K = \begin{pmatrix} M \\ N \end{pmatrix}, \quad (62)$$

in which M and N are given in Corollary 3.

Based on Corollary 3, we can design an algorithm to find all the k -kernels of a graph.

Algorithm 3. For the graph described in Corollary 3, give each $v_i \in V$ a 2-valued variable $x_i \in D = \{0, 1\}$ and define $y_i = [x_i, 1 - x_i]^T$. To find all the k -kernels of the graph, one can take the following steps.

Step 1. Compute the k -kernel matrix K of G .

Step 2. Rewrite K as the form of $K = \begin{pmatrix} M \\ N \end{pmatrix}$ shown in Corollary 3.

Step 3. Check whether there is a 1_n in $\text{col}(M)$. If not, there is no k -kernel in G and the algorithm ends here. Otherwise, go to the next step.

Step 4. Check whether there is a 0_n in $\text{col}(N)$. If not, there is no k -kernel in G and the algorithm ends. Otherwise, set

$$R = \{l | \text{col}_l(M) = 1_n, \text{col}_l(N) = 0_n\}. \quad (63)$$

Step 5. For each l in R , compute $y_i = S_i^n \otimes \delta_{2^n}^l$, $i = 1, 2, \dots, n$, where

$$\begin{cases} S_1^n = \delta_2 \left[\underbrace{1 \dots 1}_{2^{n-1}} \underbrace{2 \dots 2}_{2^{n-1}} \right], \\ S_2^n = \delta_2 \left[\underbrace{1 \dots 1}_{2^{n-2}} \underbrace{2 \dots 2}_{2^{n-2}} \underbrace{1 \dots 1}_{2^{n-2}} \underbrace{2 \dots 2}_{2^{n-2}} \right], \\ \vdots \\ S_n^n = \delta_2 \left[\underbrace{12 \dots 12}_{2^{n-1}} \right]. \end{cases} \quad (64)$$

Step 6. Pick out $y_i = \delta_2^1$, and set $S_l = \{v_i | y_i = \delta_2^1\}$. S_l is a k -kernel of G . All the k -kernels of G are $\{S_l | l \in K\}$.

4. Comparisons with Existing Works and Computational Complexity

Roughly speaking, the approaches to study the k -ESS and k -kernel problems can be classified into three types: graph theory method, linear programming method, and computer algorithm-based method. Most of these results, especially the computer-based algorithms, cannot find all the k -ESSes of a graph.

This paper uses a matrix method to consider the k -ESS and k -kernel problems of graphs. The results presented in this paper have several advantages over existing results. First, we established a matrix formulation of the k -ESS problems, which simply and clearly provide the complete information of the k -ESS of a graph in a matrix manner. Using these information, one can easily develop an algorithm to find all the k -ESSes of a graph. Second, the results of the k -ESS have good expansibility; the main results, including theoretic and algorithm ones, can be transplanted to the case of k -kernels of graphs; after slight modifications, they are applicable to formulate and find the k -kernels of graphs.

Besides, several new concepts are proposed for the k -ESS and k -kernel of graphs, such as k -ESS matrix, minimum k -ESS matrix, and k -kernel matrix. These matrices can be viewed as structure matrices of k -ESS, minimum k -ESS, and k -kernel because they contain the complete information of all the k -ESSes, minimum k -ESSes, and k -kernel sets of graphs.

The time complexity of the proposed algorithms is $O(n)$, including Algorithms 1, 2, and 3. Take Algorithm 2 for example (the complexity analysis of Algorithms 1 and 3 are very similar to that of Algorithm 2). For a given graph G with node set $V = \{v_1, v_2, \dots, v_n\}$, of all the four steps of Algorithm 2, only Step 3 involves iterative operation. Therefore, for graphs of n vertices, the times of iterative operation are no more than $n - 1$ in the worst situation. Thus, the time complexity is $O(n)$. Besides, all the involved computations in Algorithm 2 are STP, which can be easily performed with the STP tool box.

5. Examination of Results

In this section, the correctness of the results are examined by using them to solve a network covering problem, as shown in Figure 1. Because a k -kernel is both a k -ESS and a k -ISS, it needs only to check if that Algorithm 3 can find all the k -kernels of a graph.

The 2-AM of G is

$$A^{[2]} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \quad (65)$$

Steps 1, 2 and 3: with $A^{[2]}$, we can get the matrices M and N in Corollary 3. In M , the columns being 1_n are as follows:

$$\begin{aligned} & \text{col}_{103}(M), \text{col}_{104}(M), \text{col}_{113}(M), \text{col}_{117}(M), \\ & \text{col}_{123}(M), \text{col}_{124}(M), \text{col}_{125}(M), \text{col}_{126}(M), \\ & \text{col}_{127}(M), \text{col}_{128}(M), \text{col}_{171}(M), \text{col}_{172}(M), \\ & \text{col}_{173}(M), \text{col}_{174}(M), \text{col}_{176}(M), \text{col}_{177}(M), \\ & \text{col}_{183}(M), \text{col}_{184}(M), \text{col}_{188}(M), \text{col}_{189}(M), \\ & \text{col}_{190}(M), \text{col}_{192}(M), \text{col}_{193}(M), \text{col}_{216}(M), \\ & \text{col}_{224}(M), \text{col}_{227}(M), \text{col}_{230}(M), \text{col}_{238}(M), \\ & \text{col}_{239}(M), \text{col}_{240}(M), \text{col}_{242}(M), \text{col}_{247}(M), \\ & \text{col}_{248}(M), \text{col}_{249}(M), \text{col}_{251}(M), \text{col}_{252}(M), \\ & \text{col}_{253}(M), \text{col}_{254}(M), \text{col}_{255}(M), \text{col}_{256}(M). \end{aligned} \quad (66)$$

And, the matrix N has the following columns being 0_n :

$$\begin{aligned} & \text{col}_{87}(M), \text{col}_{98}(M), \text{col}_{100}(M), \text{col}_{101}(M), \\ & \text{col}_{102}(M), \text{col}_{118}(M), \text{col}_{119}(M), \text{col}_{120}(M), \\ & \text{col}_{123}(M), \text{col}_{124}(M), \text{col}_{126}(M), \text{col}_{127}(M), \\ & \text{col}_{128}(M), \text{col}_{129}(M), \text{col}_{132}(M), \text{col}_{133}(M), \\ & \text{col}_{174}(M), \text{col}_{176}(M), \text{col}_{178}(M), \text{col}_{179}(M), \\ & \text{col}_{184}(M), \text{col}_{185}(M), \text{col}_{188}(M), \text{col}_{190}(M), \\ & \text{col}_{192}(M), \text{col}_{202}(M), \text{col}_{216}(M), \text{col}_{224}(M), \\ & \text{col}_{225}(M), \text{col}_{228}(M), \text{col}_{238}(M), \text{col}_{240}(M), \\ & \text{col}_{246}(M), \text{col}_{248}(M), \text{col}_{250}(M), \text{col}_{251}(M), \\ & \text{col}_{252}(M), \text{col}_{254}(M), \text{col}_{255}(M). \end{aligned} \quad (67)$$

Step 4: based on Steps 1, 2, and 3, construct the following set:

FIGURE 1: Graph model of the network covering problem.

$$K = \left\{ \begin{array}{l} 123, 124, 126, 127, 128, 174, 176, 184, 190, 192, \\ 216, 224, 238, 240, 248, 251, 252, 254, 255 \end{array} \right\}. \quad (68)$$

According to Corollary 3, since $|K| = 20$, there are total 20 2-kernels in G .

Step 5: for each element $l \in K$, we get y_i by computing $y_i = S_i^8 \times \delta_{256}^l$, $i = 1, 2, \dots, 8$. The set $S_l = \{v_i | y_i = \delta_2^1\}$ corresponds to a 2-kernel. Take $l = 123$ for example, we get

$$\begin{aligned} y_1 &= S_1^8 \times \delta_{256}^{123} = \delta_2^1, \\ y_2 &= S_2^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_3 &= S_3^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_4 &= S_4^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_5 &= S_5^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_6 &= S_6^8 \times \delta_{256}^{123} = \delta_2^1, \\ y_7 &= S_7^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_8 &= S_8^8 \times \delta_{256}^{123} = \delta_2^1, \end{aligned} \quad (69)$$

where $y_1 = \delta_2^1$, $y_6 = \delta_2^1$, and $y_8 = \delta_2^1$. We then get a 2-kernel $S_{123} = \{v_1, v_6, v_8\}$, which is marked by triangles in Figure 1.

For $l = 124$, we calculate $y_i = S_i^8 \times \delta_{256}^{124}$, $i = 1, 2, \dots, 8$, and get

$$\begin{aligned} y_1 &= S_1^8 \times \delta_{256}^{123} = \delta_2^1, \\ y_2 &= S_2^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_3 &= S_3^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_4 &= S_4^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_5 &= S_5^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_6 &= S_6^8 \times \delta_{256}^{123} = \delta_2^1, \\ y_7 &= S_7^8 \times \delta_{256}^{123} = \delta_2^2, \\ y_8 &= S_8^8 \times \delta_{256}^{123} = \delta_2^2, \end{aligned} \quad (70)$$

where $y_1 = \delta_2^1$ and $y_6 = \delta_2^1$. We get another 2-kernel $S_{123} = \{v_1, v_6\}$.

Consider $l = 174$ for another example. Computing $y_i = S_i^8 \times \delta_{256}^{174}$, $i = 1, 2, \dots, 8$, we get

TABLE 1: All the 2-kernels of G .

l	2-kernels
123	$\{v_1, v_6, v_8\}$
124	$\{v_1, v_6\}$
126	$\{v_1, v_7\}$
127	$\{v_1, v_8\}$
128	$\{v_1\}$
174	$\{v_2, v_4\}$
184	$\{v_2, v_5\}$
190	$\{v_2, v_7\}$
192	$\{v_2\}$
216	$\{v_3, v_5\}$
224	$\{v_3\}$
238	$\{v_4, v_7\}$
248	$\{v_4\}$
251	$\{v_5\}$
252	$\{v_6, v_8\}$
254	$\{v_6\}$
255	$\{v_7\}$

$$\begin{aligned}
\gamma_1 &= S_1^8 \times \delta_{256}^{174} = \delta_2^2, \\
\gamma_2 &= S_2^8 \times \delta_{256}^{174} = \delta_2^1, \\
\gamma_3 &= S_3^8 \times \delta_{256}^{174} = \delta_2^2, \\
\gamma_4 &= S_4^8 \times \delta_{256}^{174} = \delta_2^1, \\
\gamma_5 &= S_5^8 \times \delta_{256}^{174} = \delta_2^2, \\
\gamma_6 &= S_6^8 \times \delta_{256}^{174} = \delta_2^2, \\
\gamma_7 &= S_7^8 \times \delta_{256}^{174} = \delta_2^1, \\
\gamma_8 &= S_8^8 \times \delta_{256}^{174} = \delta_2^2,
\end{aligned} \tag{71}$$

where $\gamma_2 = \delta_2^1$, $\gamma_4 = \delta_2^1$, and $\gamma_7 = \delta_2^1$ indicate a 2-kernel $S_{174} = \{v_2, v_4, v_7\}$, which is marked in pentagrams in Figure 1.

Using the similar procedure, all the 2-kernels of G can be obtained, which are collected in Table 1.

For the case of $k = 3$, following the above steps, we can get all the 3-kernels of G :

$$\begin{aligned}
&\{v_1, v_7\}, \\
&\{v_2, v_4\}, \\
&\{v_2, v_5\}, \\
&\{v_3, v_5\}, \\
&\{v_5, v_8\}, \\
&\{v_6, v_8\}.
\end{aligned} \tag{72}$$

For $k = 4$, there are five 4-kernels:

$$\begin{aligned}
&\{2, 4\}, \\
&\{2, 5\}, \\
&\{3, 5\}, \\
&\{5, 8\}, \\
&\{6, 8\}.
\end{aligned} \tag{73}$$

When $k \geq 4$, the k -kernels of G are the same because the k -AM $A^{[k]}$ no longer changes.

Remark 8. According to Zhang et al. [30], the k -kernels of undirected graphs are the same as their maximum k -internally stable sets. The results above coincide with the statement.

6. Conclusion

In graph theory, k -ESS of graphs is an important research topic because many problems in the real world can be described and be solved by such graph model. In this paper, the formulation and search of k -ESS, minimum k -ESS, and k -kernels of a graph are reconsidered by using a new mathematical tool called STP. Several matrix forms of necessary and sufficient conditions of the k -ESS, minimum k -ESS, and k -kernels of graphs are established. Based on the necessary and sufficient conditions, the concepts of the k -ESS matrix, minimum k -ESS matrix, and k -kernel matrix are proposed. Besides, three algorithms of finding all the k -ESSes, minimum k -ESSes, and k -kernels of a graph are designed. The presented results may provide a new way to solve the problems of the network system such as minimization of networks.

Data Availability

No data were used to support the findings of the study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors greatly appreciate Professor Y. G. Hong for his encouragement and valuable suggestions of using STP to investigate the structure problems of graphs. This work was supported in part by National Natural Science Foundation of China (U1804150 and 62073124).

References

- [1] D. West, *Introduction to Graph Theory*, Pearson Education (Singapore) Pte. Ltd., Delhi, India, 2001.
- [2] Y. Yan, J. Yue, and H. Deng, "Matrix formulation of EISs of graphs and its application to WSN covering problems," *Journal of Mathematics*, vol. 2021, Article ID 5526532, 12 pages, 2021.
- [3] R. Mohammadi, R. Welsch, and O. Buyukozturk, "Kernel dependence analysis and graph structure morphing for novelty detection with high-dimensional small size data set," *Mechanical Systems and Signal Processing*, vol. 143, pp. 1265–1277, 2020.
- [4] L. Bernshtejn and A. Bozhenyuk, "Determination of fuzzy internal stable, external stable sets, and kernels of fuzzy oriented graphs, *Izvestiya Akademii Nauk*," *Teoriya I Sistemy Upravleniya*, vol. 38, pp. 161–166, 1999.
- [5] H. Liu, Y. Liu, Y. Li, Z. Wang, and F. E. Alsaadi, "Observability of Boolean networks via STP and graph methods," *IET Control Theory & Applications*, vol. 13, no. 7, pp. 1031–1037, 2019.

- [6] R. An, C. Yang, and Y. Pan, "Graph-based method for fault detection in the iron-making process," *IEEE Access*, vol. 8, pp. 40171–40179, 2020.
- [7] M. Wang and G. Hu, "A novel method for twitter sentiment analysis based on attentional-graph neural network," *Information (Switzerland)*, vol. 11, pp. 236–244, 2020.
- [8] S. Yuan and M. Huifang, "Method based on directed graph and binary decision diagram to break logic loops," *Kern-technik*, vol. 84, no. 6, pp. 488–499, 2019.
- [9] D. Cheng, H. Qi, and Y. Zhao, *An Introduction to Semi-tensor Product of Matrices and its Applications*, World Scientific Publishing Co. Pte. Ltd., Singapore, 2012.
- [10] X. Han, P. Wang, and Z. Chen, "Matrix approach to verification and enforcement of nonblockingness for modular discrete-event systems," *Science China-Information Sciences*, vol. 63, pp. 2315–2326, 2020.
- [11] C. Huang, J. Lu, D. W. C. Ho, G. Zhai, and J. Cao, "Stabilization of probabilistic Boolean networks via pinning control strategy," *Information Sciences*, vol. 510, pp. 205–217, 2020.
- [12] Y. Li, H. Li, and X. Ding, "Set stability of switched delayed logical networks with application to finite-field consensus," *Automatica*, vol. 113, pp. 168–179, 2020.
- [13] M. Meng, J. Lam, J. Feng, and K. Cheung, "Stability and stabilization of boolean networks with stochastic delays," *IEEE Transactions on Automatic Control*, vol. 64, pp. 790–796, 2020.
- [14] Y. Wu, X.-M. Sun, X. Zhao, and T. Shen, "Optimal control of Boolean control networks with average cost: a policy iteration approach," *Automatica*, vol. 100, pp. 378–387, 2019.
- [15] Q. Zhang, J. Feng, and Y. Yan, "Finite-time pinning stabilization of Markovian jump Boolean networks," *Journal of the Franklin Institute-Engineering and Applied Mathematics*, vol. 357, pp. 11325–11337, 2020.
- [16] Z. Zhang, C. Xia, S. Chen, T. Yang, and Z. Chen, "Reachability analysis of networked finite state machine with communication losses: a switched perspective," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 845–853, 2020.
- [17] Y. Zou, J. Zhu, and Y. Liu, "Cascading state-space decomposition of Boolean control networks by nested method," *Journal of the Franklin Institute*, vol. 356, no. 16, pp. 10015–10030, 2019.
- [18] Y. Wang, C. Zhang, and Z. Liu, "A matrix approach to graph maximum stable set and coloring problems with application to multi-agent systems," *Automatica*, vol. 48, no. 7, pp. 1227–1236, 2012.
- [19] M. Xu, Y. Wang, and A. Wei, "Robust graph coloring based on the matrix semi-tensor product with application to examination timetabling," *Control Theory and Technology*, vol. 12, no. 2, pp. 187–197, 2014.
- [20] M. Xu and Y. Wang, "T-coloring of graphs with application to frequency assignment in cellular mobile networks," in *Proceedings of the 33rd Chinese Control Conference*, pp. 2536–2541, Nanjing, China, July 2018.
- [21] M. Xu, Y. Wang, and P. Jiang, "Fuzzy graph coloring via semi-tensor product method," in *Proceedings of the 34th Chinese Control Conference*, pp. 973–978, Hangzhou, China, July 2019.
- [22] J. Zhong, J. Lu, C. Huang, L. Li, and J. Cao, "Finding graph minimum stable set and core via semi-tensor product approach," *Neurocomputing*, vol. 174, pp. 588–596, 2016.
- [23] M. Meng, J. Feng, and X. Li, "A matrix method to hypergraph transversal and covering problems with application in simplifying boolean functions," in *Proceedings of the 35th Chinese Control Conference*, pp. 2772–2777, Chengdu, China, July 2016.
- [24] J. Yue, Z. Chen, Y. Yan, and X. Jin, "Solvability of k -track assignment problem: a graph approach," *Control Theory & Applications*, vol. 34, pp. 457–466, 2017.
- [25] Y. Yan, J. Yue, Z. Chen, and Y. Liu, "Algebraic expression and construction of control sets of graphs using semi-tensor product of matrices," *IEEE Access*, vol. 7, pp. 113440–113451, 2020.
- [26] X. Li, C. Magnant, and Z. Qin, "Introduction," *Properly Colored Connectivity of Graphs*, vol. 7, pp. 1–8, 2018.
- [27] O. Bleblou, B. Šešelja, and A. Tepavčević, *Generalized Boolean Algebras and Applications*, Chapman & Hall, Boca Raton, USA, 2019.
- [28] W. Fang, "The relationship between Boolean derivative, e-derivative and c-derivative of Boolean function," *Journal of Zhejiang University (Science Edition)*, vol. 3, pp. 124–131, 2019.
- [29] X. Liu, Z. Ji, and T. Hou, "Graph partitions and the controllability of directed signed networks," *Science China-Information Sciences*, vol. 62, pp. 1467–1476, 2020.
- [30] L. Zhang, M. Song, L. Xiao, J. Bu, and C. Chen, "Fast multi-view segment graph kernel for object classification," *Signal Process*, vol. 93, pp. 1597–1607, 2018.