Constraint Inversion Model of Core Science Complex Network

Yuzhe Jia¹, Junkai Yi¹ and Shuan Yan²

¹ School of Automation Beijing Information Science and Technology University, Beijing, China
² Institute of public security, Tsinghua University, Haidian District, Beijing, China
Email: jyzzgzyx@outlook.com; yijk@bistu.edu.cn; 1608146643@qq.com

Abstract. In order to reduce the research costs, increase the efficiency of scientific researches, and improve the inversion abilities of core technologies, in the paper, a constrained inversion model aiming at tracking the source of core technology was proposed. Firstly, a complex network was built and it treated the most important product or technology as the core node vector. In the network, the social and technical attributes of the product were assigned as the elements of the core node vector. Therefore, the relationships among the core node and other nodes represented connective strength which were fairly important for calculating the path weights. Essentially, the weights were integrated attributes derive from a called score matrix. Then a constrained optimization criterion was employed to evaluate the scores of paths so that the best optimal path was determined. For reducing the number of searching and the times of calculating, all the constraints relationships of node vectors were classified into several types. The experimental results demonstrate that the proposed approach has a superior performance against the current state-of-the art methods over a real case.

Keywords. Complex networks; constrained optimization; inversion; technical; social attributes.

1. Introduction

With the high-speed development of information technology, scientific and technological level of start-up company is still far behind the large firms in many aspects and they have been in a state of learning and following others for a long time. This situation leads many aspects to become the production workshop of foreign companies rather than the science research institution. However, in these fields, large enterprises take a highly technical blockade which leads to a state of very slow progress developing in these start-up company for a long time. In order to overcome this situation, we need to study the theory of inversion model of original science complex network. This theory can help us change the traditional way [1-2] of directly acquiring technology information to an indirect way, and we can also track the development of world advanced technology by inversion reasoning the core products that have got.

It is difficult for us to directly obtain those advanced scientific and technological achievements. But we can track the entire process of products studying, producing, manufacturing, selling, and gathering as much information related to the core product as possible in big data case, and clarifying all the external relational of those companies that have partnerships with the core enterprise. A complex network model [3-4] was built by the aforementioned steps. The process of constrained inversion reasoning will base on model. Above all, we will try to break the blockade of large firm and obtain details of the technology and production as much as possible from all aspects of the core product. These details will provide an
adequate technical support and guidance for the start-up company to develop and produce the products [5].

The constrained inversion process [6-7] herein need to deeply study the technology and social attributes of the core node as the first step. Then we can build the association network for the core node and those nodes on the second layer based on the constraint relationships of those nodes vectors. Then we can cope with nodes on the second layer in a similar manner and a complex constraint inversion model can be built in the end. The pair-wise weights of path between one and other nodes in the network can be calculated based on a standard scoring matrix. Then the constraint optimization criterion will be conducted. This criterion will be used to evaluate the scores of paths so that the best optimal path was determined. It has a direct influence on the correctness of the result. Then the reasoning was conducted on the complex map [8] that has been created above depends on the inversion strategies. In order to speed up the reasoning process, the constrained inversion events have been classified into different classes [9] which can accelerate the speed by reducing the number of searching and the times of calculating.

In order to verify the correctness of the theory above, a real case was engaged to illustrated effectiveness of the proposed approach. In the case, the nodes in different layers and constrains relating relationship among nodes were calculated according to the process of the constrained inversion reasoning [10]. Then the combination of three strategies was used to calculate the total weight of each path and the results of each path were evaluated by constraint optimization criterion at the same time. In order to highlight the effect of the proposed reasoning strategies, the ant colony algorithm (ACO) [11-12] and the genetic algorithm (GA) [13] were used as comparisons [14-15]. ACO and GA are typical methods [16] in optimal path searching.

The main contributions of this study are as follows:

This paper created a new concept of constrained inversion model. Firstly, we transformed the core product into a vector, then a score matrix is proposed to calculate the weight between nodes. After these steps, a whole process of creating a complex network is shown behind. Based on the complex network created above, a constrained optimization criterion is employed to evaluate the score of path whether it is the optimal one. For speeding up the calculating and reducing the number of searching, all the node variable events are classified into several specific types. The combining of three strategies was proposed to get a much more correct result. Finally, a real case study is carried out to evaluate and demonstrate the effectiveness of the proposed mode.

The rest of this paper is organized as follows. Section 2 presents some feathers of complex networks and a brief overview on the procedure of establishing a complex network. In Section 3, some detailed descriptions of the proposed constraint optimization criterion and Inversion reasoning strategies are presented. A real case is studied and the results are shown in Section 4, followed by Section 5 as conclusions.

Related works: There are a lot of ways to collect and deduce information and many people insist on doing research works [17] in this area for a long time. Due to its critical importance, Inversion reasoning of information has received significant attention from the research community. The typical method includes inverting tree and vertical tree and so on. The process of using inverted tree to deduce information is to collect the information that has achieved and extract new information from them. The process above will create a pattern like an inverted tree. Inverted tree has two basic characteristics: the first one is multiple layers. The high value of the information node in the higher layers is abstracting from those nodes in the lower layers by combining and extracting their information. The more layers there are, the final node information and the corresponding optimal path will be more trustworthy. The second characteristic is constraint. From the previous analyzing, we know that the result is relatively precisely because there are many constraints and restriction among each node and these restrictions will help derive a more accurate intelligence result and a special pattern of inverted tree. On the contrary, the shape of the vertical tree is different. The root of the vertical tree is under the treetop. Vertical tree is also opposite to the inverted tree in the aspect of application areas. Vertical tree need not to extract information from any nodes and it just need to derive all possible results through the node that has
achieved. The result may be a divergent set instead of a single one node. These kinds of ideas above can be found in many information collecting or reasoning systems.

The advantage of the inverted tree and the vertical tree in the aspect of intelligence reasoning is directly seeing and simply using. The relationship between child node and the core node can be very clearly found and easily understood. But the obvious shortcoming is inaccuracy. This theory is an idealized concept and can not widely used in practical calculation circumstance.

Some methods that used for searching optimal path also can be found from the map solving algorithms, such as Dijkstra [18-19] algorithm. However, Dijkstra algorithm only can choose the smallest edge of the next step, but it can’t easily get the global optimal result [20]. And these kinds of algorithms usually can’t apply to the evaluation criterion of the optimal path which provides later on. So its results can not directly compare with the searching strategies proposed in this paper, and have not been listed in the final test as well.

2. Model of Original Science Complex Network

2.1. Definition of Complex Networks

The network which derived from constraint inversion theory belongs to a complex network map. And the goal of studying complex networks is to give a more scientific and reasonable explanation of a network based on its quantitative and characteristics features. Usually the complex network has some important features as follows:

The structure of the network is complex. There isn’t a general simple rule about the structure of the network, and the distribution of the edge in the network is constantly changing. The connection path may frequent change in direction and weight. For instance, the number of nodes in the network may be always changing.

The type of nodes in the network is very complex. There may be many kinds of nodes in the network and there may have many differences in weight and direction of connection specifying to a same type node. Connection and disconnection of each node may affect the existence of a number of nodes associated, it may also lead to some weight values and edges changed which related to this node.

The influence of interaction factors is complex. There are many complicated factors which have a lot of influence and restriction on each other within the network. The external factors which impact on the network once may cause an unexpected chain changing. For example a large area pruning of a network happened sometimes just because of a single node’s removing.

In order to establish the original science complex network map, we propose three concepts of entities according to the characteristics of complex networks.

Original entity: It is the organization who has core techniques. The Organization may be a company, research institute or research group, etc.

Middle entity: It is referred to those organizations who create critical component by packaging the core techniques which come from the original entity.

Target entity: This organization will produce core product by purchasing the critical component of the middle entity and processing deeply.

Any entity may be included in other entities among those three entities. For example the middle entity may contain the original entity. Figure 1 is a forward deduction process of the product.

Figure 1. Forward deduction of the product.

Figure 1 shows a forward deduction process of the product: original entity A will provide the middle entity B with core techniques, the middle entity B will produce critical component for target entity C by absorbing techniques from entity A. The product D which come from Entity C is assembled with the
critical component from B. The real situation may be more complicated: the number of the middle entity B may be more than one, the process that core techniques which coming from original entity will become a part of the critical product may need to go through several middle entities or the original entity may be a part of the target entity at the beginning.

From the view point of users, they can only see the product D which coming from target entity C, but they don’t know the middle entity B and original entity A. This is also the technical secret of target entity C. And it will be very expensive if we directly buy the product D. As figure 2 shows, the study of the inversion model of original science complex network in this paper mainly wants to find out the critical path towards the middle entities x and original entity y through a constrained inversion research specific to product D which owned by the target entity C. It will be low-cost when we purchase or study the core techniques which is the main composition of the product D after reasoning successfully.

![Figure 2. The inversion deduction of target entity products.](image)

2.2. Establishing of Complex Networks
Undirected complex network map is composed of many complex nodes and paths with weight values. So the undirected complex network map can be expressed as the expression:

\[ G = (V', E', W') \]

G represents the undirected network map that will be created; V’ represents all the associated nodes in the map; E’ represents the connection edge between the nodes. There must be an undirected path if those nodes have a constraint connection relation with each other, no matter the path is inside a layer or across the layers, and this path is also the direction that the searching methods will mainly depend on; W’ represents all the weight value of edges in the map. The value of the weight is determined by the strength level of relationship between nodes. And the strength level of relationship is also determined by the correlation degree of the technology and social attribute among nodes. So the process to find the optimal path between core node and target node is also a procedure to find the optimal path on the map G.

Each of nodes in the layers of map G is a vector, and they all contains several common categories of properties. Suppose the node vector is expressed as follows:

\[ X_{ij} = \{a_1, a_2, \ldots, a_k\} \]

The \( X_{ij} \) is a node vector. \( a_1, a_2, \ldots, a_k \) is the attribute value of node \( X_{ij} \). Each node is composed of \( k \) attributes. K represents an amount. All nodes have equal number of attribute types. The type of these properties includes technology attribute and social attribute.

The forming process of complex network map can be described as follows basing on the above definition of the complex network map G and node \( X_i \): Suppose the core node is \( X_i \) which is locating on the first layer; the nodes on the second layer are those nodes who have directly relations with the core node. The weight value between the core node and the second layer node is determined by the level of strength which is also calculated relying on the relation score matrix between the two nodes. Therefore, all the nodes that directly associated with the node \( X_i \) can be classified as the second layer nodes set and we can name it as \( U_2 \). The possible status of connection between core node and the second layer node is shown in figure 3.

\[ U_2 = \{X_{2,1}, X_{2,2}, \ldots, X_{2,n}\} \]
Figure 3. The probable relationship between Core node and the second level nodes.

The set $U_3$ is consist of nodes which are directly associated with any of those nodes locating on the second layer (The nodes in set $U_3$ are not including the nodes in layer 1 and layer 2), as shown in figure 4.

$$U_3 = \{X_{3,1}, X_{3,2}, \ldots, X_{3,k}\}$$

Figure 4. Some probable relationship among all three level nodes in this example.

Deducing by the same idea, we can know that the set $U_m$ is consist of nodes which are directly associated with any of those elements locating on the $m-1$ layer (The nodes in set $U_m$ are not including the nodes among layer 1 and layer $m-1$).

$$U_m = \{X_{m,1}, X_{m,2}, \ldots, X_{m,j}\}$$

It will be better for us to think the distribution of all those associated nodes according to the inversion process if we use $U_1, U_2, \ldots, U_m$ to categorize all nodes, and this will be benefit to fully describe the spreading around connection status of core node.

Relation score matrix is consist of horizontal ordinate and vertical coordinates which represent all attributes of the two nodes. The value at the intersection of horizontal and vertical coordinates in the matrix represents the size of strength between two properties of connecting nodes. The weight value is the total sum of all connection intensity values between two nodes’ attributes. The value will be bigger if it has more deep impact on the entire inversion deduction.
By analyzing the description about the definition of complex network map, definition of node, and the process of create complex network map, the nodes in the map has the following properties:

Property 1 The same node may belong to different node sets at the same time. In other words, any one node \( X_{m,n} \) can meet the following expression.

\[
X_{m,n} \in U_i \text{ and } X_{m,n} \in U_j
\]

Property 2 The current node has no directly associated connection path with the node which is separated by one layer in the depth direction. In other words, there is no one node \( X_{m,n} \) can meet the following expression:

\[
X_{m,n} \in U_j \text{ and } X_{m,n} \in U_{j-2}
\]

For example, there is no directly connection path between the core node and those nodes on the third layers. Complex network map has no isolated node and edge. All the nodes make up a total node set called NodeSet.

It can be expressed as below:

NodeSet \( \left\{ X_{1,1}, X_{2,1}, \ldots, X_{m,n} \right\} = \left\{ U_1, U_2, \ldots, U_n \right\} \)

All paths and its corresponding weight make up a set called EdgeSet. It can be expressed as below:

EdgeSet \( \left\{ E\left(X_{1,1}, X_{2,1}\right), E\left(X_{2,1}, X_{3,1}\right), \ldots, E\left(X_{m-1,m}, X_{m,n}\right) \right\} \)

All the constraint between nodes vector make up a set called ConstraintSet. It can be expressed as below:

ConstraintSet \( \left\{ C\left(X_{1,1} X_{2,1}\right) = W_{1,2}, C\left(X_{2,1} X_{3,1}\right) = W_{2,3}, \ldots, C\left(X_{x,j} X_{m,n}\right) = W_{x,j-m,n} \right\} \)

There is no relationship if the weight value of the path is zero. So this can be expressed in a property as below:

Property 3 The case that constraint between two nodes is equal to zero means no edge is existing between the two nodes. It can be expressed as below:

\[
C\left(X_{x,j} X_{m,n}\right) = 0
\]

It also means that there is no path called \( E\left(X_{x,j} X_{m,n}\right) \) in the set EdgeSet.

3. **Constrained Inversion Method**

3.1. **Constraint Optimization Criterion**

Optimization goal is the optimal value of a path which is a route between the core node and target node and the goal can be represented by letter \( F \). The lower of the optimal value, the better of the path. And it also means that this path is much closer to the final result than others. The optimal value of the path can be expressed as below:

\[
F = A - D
\]

Letter \( A \) represents the maximum theoretical value of a path. Letter \( D \) represents the real total weight of the path. The definition of this method can be expressed as below:

\[
D = \sqrt{W_{1,2} W_{2,3} \cdots W_{m,n}}
\]

This expression shows that the larger of the value \( D \) is, the higher possible correctness of the path will be. Meanwhile, this path will be likely to have higher overlap ratio with the correct path either.
3.2. Inversion Reasoning Strategy

Constraint node set is a node classification method basing on the position of node on the path. Every step of the path can form a constraint node set. For example, any nodes that use to be the first step will become a tuple of set $V_1 \{x_{1,1}\}$. However the set $V_i$ has only one node $X_{i,1}$ because the starting node of all the paths must be the core node $X_{i,1}$. Basing on the same idea, we can form the final constraint node set $V_1, V_2, \ldots, V_n$.

The calculating of the path also needs to follow some certain strategies after sorting the nodes into different sets. Here are three strategies which can be used for searching and calculating the optimal path.

Definition 1. Suppose the constraints set $V_1, V_2, \ldots, V_n$ are consist of node vector $x$ in the limited domain of $D_x$. $V_i$ is the constraint set of searching and it will also satisfy the following requirements:

$$V_1, V_2, \ldots, V_n \subseteq D_x$$
$$X_{1,1} \in V_1$$
$$X_{m,n} \in V_n$$
$$V_s = \text{Min}\{\text{Num}(V_i), \text{Num}(V_j), \ldots, \text{Num}(V_n)\}$$

It means that the core node $X_{1,1}$ in set $V_1$ is the starting node of all searching paths, and it is also the single one node in the set $V_1$. The nodes on the middle path belong to set $V_2 \ldots V_{n-1}$ respectively. The target node $X_{m,n}$ belongs to set $V_n$. $\text{Num}(V_i)$ represents the number of nodes in the set $V_i$. When we start to search the optimal path, the least number of nodes among set $V_2$ and $V_n$ will be used at each time until the optimal path is found. This kind of method is called “Firstfail” strategy. As shown in figure 5, for the convenience of presenting, the figure hides part of the nodes, paths and weights in order to easily demonstrate.

![Figure 5. Example of Firstfail strategies.](image)

Definition 2. Suppose the constraints set $V_1, V_2, \ldots, V_n$ are consist of node vector $x$ in the limited domain of $D_x$. $V_i$ is the constraint set of searching and it will also satisfy the following requirements:

$$V_1, V_2, \ldots, V_n \subseteq D_x$$
$$X_{1,1} \in V_1$$
$$X_{m,n} \in V_n$$
$$V_s = \{V_1, V_2, \ldots, V_n\}$$

It means that the core node $X_{1,1}$ in set $V_1$ is the starting node of all search paths, and it is also the single one node in the set $V_1$. The nodes on the middle path belong to set $V_2 \ldots V_{n-1}$ respectively. The
target node \( X_{\text{mn}} \) belongs to set \( V_n \). When we start to search the optimal path, the set \( V_i \) will be used at each time until the optimal path is found (Or the set \( V_n \) will be used at each time until the optimal path is found). This kind of method is called “Naive” strategy. As shown in figure 6 below.

Definition 3. Suppose the constraints set \( V_1, V_2, \ldots, V_n \) are consist of node vector \( x \) in the limited domain of \( D_x \). \( V_i \) is the constraint set of searching and it will also satisfy the following requirements:

\[
V_1, V_2, \ldots, V_n \subset D_x
\]

\[
X_{1,1} \in V_1
\]

\[
X_{\text{mn}} \in V_n
\]

\[
D'(V_1, V_2, \ldots, V_n) > D_{\text{max}}
\]

It means that the core node \( X_{1,1} \) in set \( V_1 \) is the starting node of all search paths, and it is also the single one node in the set \( V_1 \). The nodes on the middle path belong to set \( V_2 \ldots V_{n-1} \) respectively. The target node \( X_{\text{mn}} \) belongs to set \( V_n \). \( D' \) represents the total weight of the current path. \( D_{\text{max}} \) represents the largest weight of the paths that has been calculated before. When we start to search the optimal path, we will try to know whether the total weight of the current path is bigger than any of the total weights before. If it is right, we will go on this searching. But if it is false, we will stop the searching process at once and start the new searching again. This kind of method is called “BranchAndBound” strategy. As shown in figure 7 below. In the figure, suppose that when we use the largest weight between node \( X_{4,2} \) and any of those nodes in set \( V_5 \), however the total weight of path that node \( X_{4,2} \) locating on is always less than the result of the path that node \( X_{4,1} \) locating on, at this time, we will stop to use nodes in set \( V_5 \) to match the path that the node \( X_{4,2} \) locating on right now. This kind of judging can help reduce the number of calculating times within the procedure of selecting path.

![Figure 6](image6.png) Example of Naive strategies.

![Figure 7](image7.png) Example of Increase strategies.

Throughout the searching process, the number of path searching that different nodes trigger can be a very large amount if the type of path searching can not be classified and optimized. And in order to reduce the times that the variable trigger, the event of the node selecting will be classified into different classes. Firstly, assuming the initial discrete range of node \( X \) is the domain of set \( V \{ X_{1,1}, X_{1,2}, \ldots, X_{mn} \} \).

The changing of variable \( X \) which is operated by the constraint reasoning machine can be expressed as:

\[
X \in \{ X_{1,1}, X_{1,2}, \ldots, X_{mn} \} \rightarrow X \in \{ X_{1,1}, X_{1,2}, \ldots, X_{m,k-1}, X_{m,k+1}, X_{mn} \}
\]

From the above formula, we can find the changing of the expression is that some of values have been removed from the domain. This kind of change can be donated by the phrase of \text{DOMCHG}. As shown in figure 8, and we can get a new definition as follow:
Definition 4. Assume that variable node $X$ belongs to the domain of $D_X$, If $D'_X \subset D_X$ and $D_X \neq \emptyset$, then the process of $D_X \rightarrow D'_X$ has the type of node variable event called $\text{DOMCHG}$.

If the variable tuple which is removed from variable set is $X_{1,1}$ or $X_{m,n}$, as shown in the following two expressions, and it means that the boundaries of variable range has changed. This type of node variable event is called $\text{BOUND}$. As shown in figure 8.

\[
X \in \{X_{1,1}, X_{1,2}, \ldots, X_{m,n}\} \rightarrow X \in \{X_{1,2}, \ldots, X_{m,k-1}, X_{m,k}, X_{m,k+1}, X_{m,n}\}
\]

\[
X \in \{X_{1,1}, X_{1,2}, \ldots, X_{m,n}\} \rightarrow X \in \{X_{1,1}, X_{1,2}, \ldots, X_{m,k-1}, X_{m,k}, X_{m,k+1}\}
\]

Assume that variable node $X$ belongs to the domain of $D_X$, and $\max(D)$ is the maximum value in variable domain, $\min(D)$ is the minimum value in variable finite domain. So we can draw the definition below:

Definition 5. Assume that variable node $X$ belongs to the domain of $D_X$, If $D'_X \subset D_X$ and $D_X \neq \emptyset$, it can also satisfy any of the following requirements:

$\max(D_X) > \max(D'_X)$

or $\min(D_X) > \min(D'_X)$

Then the process of $D_X \rightarrow D'_X$ has the type of node variable event called $\text{BOUND}$.

When the discrete range of variable has reduced from lots of values to only one, the constraint reasoning machine will regard this kind of node variable event as a type of $\text{SINGLE}$. As shown in figure 8 and the expression below:

\[
X \in \{X_{1,1}, X_{1,2}, \ldots, X_{m,n}\} \rightarrow X \in \{X_{m,n}\}
\]

Assume $\text{Num}(D)$ represents the number of variable finite domain, and the definition 6 can be expressed as follow:

Definition 6. Assume that variable node $X$ belongs to the domain of $D_X$, If $D'_X \subset D_X$ and $\text{Num}(D'_X) = 1$, then the process of $D_X \rightarrow D'_X$ has the type of node variable event called $\text{SINGLE}$.

**Figure 8.** A simple example of node variable event type $\text{DOMCHG}$, $\text{BOUND}$, $\text{SINGLE}$. Removing the node $X_{1,1}$ will cause event of $\text{BOUND}$. Removing any of nodes $X_{2,1}$, $X_{3,1}$, $X_{4,1}$ will cause event of $\text{DOMCHG}$. Removing the node $X_{5,1}$ will cause event of $\text{DOMCHG}$. The case of remaining node $X_{1,1}$ only will cause event of $\text{SINGLE}$. 
Analyzing from the three types of event variables $E_x$ above, we can know variables event $E_x$ may have multiple events at the same time. For example, the change of a variable which belongs to a certain event type can also belong to another type of event. Deducing from the range variable of events, we can know $\textit{SINGLE~BOUND~DOMCHG}$.

Definition 7 Assume that variable $X$ and $X'$ belongs to the domain of $D_X$, if variable $X$ and $X'$ can not exist at the same time, then we recognize that the variable $X$ and $X'$ have the mutex relationship. As shown in figure 9 and the expression below:

$$X \in \{X_{1,1}, X_{1,2}, \ldots, X_{m,n}\} \rightarrow X' \notin \{X_{1,1}, X_{1,2}, \ldots, X_{m,n}\}$$

Definition 8 Assume that variable $X$ and $X'$ belongs to the domain of $D_X$, if variable $X'$ can not appear at the same time when variable $X$ is existing. Then we recognize that the variable $X$ and $X'$ have the compatible relationship. As shown in figure 9 and the expression below:

$$X \in \{X_{1,1}, X_{1,2}, \ldots, X_{m,n}\} \text{ and } X' \in \{X_{1,1}, X_{1,2}, \ldots, X_{m,n}\}$$

Definition 9 Assume that variable $X$ and $X'$ belongs to the domain of $D_X$, if variable $X'$ must appear at the same time when variable $X$ is existing, then we recognize that the variable $X$ and $X'$ have the front-and-back relationship. As shown in figure 9 and the expression below:

$$X \in \{X_{1,1}, X_{1,2}, \ldots, X_{m,n}\} \rightarrow X' \in \{X_{1,1}, X_{1,2}, \ldots, X_{m,n}\}$$

![Figure 9](image-url)

**Figure 9.** A simple example of mutex, compatible, front-and-back relationship.

### 4. Experimental Result and Analysis

In order to test the feasibility of the search method, this experiment uses a company’s image recognition technology as the target package that we want to reasoning. We will try to find the associated institution in the middle stage and the original development institution which has the largest constraint optimization value through the methods described herein. When we build the complex network map which core node is the image recognition product, we are required to analyze the technical and social attributes of the product at first. This product is the core node $X_{1,1}$ of the complex network which also belongs to the constraint node set $U_1$ and it is located on the first layer of all paths either. The node $X_{1,1}$ is a vector and the its elements represent the technical and social attributes. And the technical attributes contain technical parameters and product information. Technical parameters include some aspects such as the platform based on, developing language, calculating speed, variety types of pictures that can be identified, identification precision, the different functions and compatibility; Product information contains product location, the date of production, product quotation, the style of product’s designing and so on. And the social attribute contains business relation such as whether the project has similar
cooperation relationship previous, competition relationship, the same investment and other emotional contacting relationship such as whether there are relatives, friends, workmates inside and so on. Finally, a total of 60 kinds of technical and social attributes are selected for reference.

According to the feature of technical and social attributes of node $X_{1,1}$, we collected all the related companies to form constraint node set $U_2$ on the second layer. Each of the weight value between node $X_{1,1}$ and the second layer node is determined by the relation score matrix which is constructed by the technical attributes and social attributes between the two nodes. The closer between the different attributes, the stronger on the relation of connecting, and the higher of the score in the matrix. Therefore, after such reasoning, we can form a complex network map $G$ at the end, and all the nodes on the different position of path will be classified into constraint nodes set $V_1, V_2, \ldots, V_n$, respectively.

As shown in figure 10, all nodes in the figure have the same kind of connection edge with left and right adjacent nodes. But in order to clearly show in the image, a part of connecting lines and weights have been omitted.

Figure 10. The simple network structure of experiment in the paper.

All the nodes of each layer are formed a total nodes set called $Node\{X_{1,1}, X_{2,1}, \ldots, X_{m,n}\} = \{U_1, U_2, \ldots, U_m\}$; All the path between nodes and the corresponding weights constitute a total path set called $Edge\{E(X_{1,1}, X_{2,1}), E(X_{2,1}, X_{3,1}), \ldots, E(X_{n-1,m}, X_{n,m})\}$; All the constraints between nodes constitute a constraint set called $Constraint\{C(X_{1,1}, X_{2,1}) = W_{1,1-2,1}, C(X_{2,1}, X_{3,1}) = W_{2,1-3,1}, \ldots, C(X_{i,j}, X_{m,n}) = W_{i,j-m,n}\}$.

The horizontal and vertical coordinates of the scoring matrix are all the vector elements of node A, B. It is mainly used to calculate the weight value among the two nodes vector. Suppose $W_{AB}$ represents the weight between node A and B. Its calculating formula can be expressed as:

$$W_{AB} = \sum a_{ij} f(x_i, y_j)$$
In the expression above, $a_{i,j}$ is the influence factor of each element of the node attribute. And different elements of the attribute have different strength level influence on the weight. $f(x_i,y_j)$ which is the original weight value of each element.

In this study, the constraint node set in the first layer only contains the image recognition product package providing by the company that is currently researching on; the constraint node set on the second layer contains six products which are closely associated with the core node. These six products have very closely contact with the core node in the aspect of technical and social attributes, and they have a highly possible to become the technical supporters for the core product; the constraint node set on the third layer contains some associated agencies, companies or research institutes which are providing technical and development service to any of the node on the second layer. All these institutions have the ability of developing and packaging product, and they also have their own research and development department with similar experience; the constraint node set on the fourth layer is the technology providers for the companies like research institutions on the third layer. The business of some companies on the third layer are outsourcing mode. But there also have real technology provider inside of them; The fifth layer represents the experts in scientific research, development team or the core backbone members. Not every path will include the node in the fifth constraint nodes set. It also means that node in the middle layer is very likely to be the end of a path. At this time, the process of establishing the entire complex network has completed.

In this article, we use the combination of the three strategies such as Firstfail, Naive and enhance BranchAndBound to constraint reasoning search. The first step is to build a complex network, then to search all the map according to Firstfail strategy, and then use the naive strategy to select the least amount nodes of the set within the constrained nodes set. Finally using the enhanced BranchAndBound strategy to search on the result above and then we can get the optimal path. We can check the performance of the method by comparing with the results of ant colony algorithm and genetic algorithm. In the process of comparative evaluation of the results, overlap ratio is an important criterion to evaluate the correctness of the path. The expression can be expressed as:

$$f = \frac{n_i}{N_i}$$

In the expression above, $f$ represents the overlap ratio, $n_i$ represents the number of nodes that the current path is the same with the optimal path, $N_i$ represents the total number of nodes on the current path. Therefore overlap ratio reflects that how many nodes in the current path is overlapping with those nodes in the optimal path (not including the core node and the target node), and it is a very important criterion to judge whether a path is the most optimal path among all the paths that has been searched out. Table 1 is the result of searching the optimal path in the complex network map which is depending on the three methods of genetic algorithm and ant colony algorithm, combining of three strategies.

### Table 1. Use the method of path’s total weight to calculate the number of iterations and coincidence degree.

| Algorithm                | Times of calculating | Overlap ratio (%) |
|--------------------------|----------------------|-------------------|
| Combining of three strategies | 122664               | 65.25             |
| Ant colony algorithm     | 1100000              | 22.31             |
| Genetic algorithm        | 2000000              | 16.82             |

In the experiment, the correctness of reasoning is the very important for using the theory of constraint inversion. So the overlap ratio calculating is more important than the number of iterations for the requirement in this experiment. The experimental results have shown above in table 1. The number of iteration mainly includes the searching times of global constraints and the number of constraints.


calculating on each layer nodes. However the total iteration times will not get the same result in different complex network map which have the different weights, node distribution or using different pruning strategies.

The changing curve of overlap ratio coming from the three calculation methods are different from each other while the total number of iterations is increasing, as shown in figure 11. Furthermore, the ant colony algorithm and genetic algorithm haven’t reached the average effect when the total computation times of constraint calculating reached 120 thousand times. But combining of three strategies can achieve the optimal performance that required. This result fully reflects the good capacity of the method that combine the three search strategies.

In this study, the selection of platform and software program which the constraint logic programming needed is listing as follows:

Hardware: PC Pentium4, CPU 2.1G, 4096M Memory
Software: Ubuntu 14.04, kernel 3.13.9, C++

In the experiment, the time that each of the three searching methods used is very closely to each other, and the difference is far from the order of magnitude. In order to compare to some artificial intelligence searching algorithms, this article uses a simple case which whole relations of the complex network are not very complicated. And all the information of the nodes in the network are coming from public channels. The final optimal path also can be found in open channel, and this is very convenient for us to verify the results as well. Analyzing from the experimental process, results and principle of searching methods, it can be found that the result of the combining of three strategies have a higher overlap ratio than others on the aspect of searching effect. Furthermore, the combining of three strategies also need less time than other strategies because this method can forecast the possibility whether the path turn into the optimal path rather than blind attempting again and again. On the aspect of accuracy rate, combining of three strategies have the advantage of getting the global optimal result much easier and it can show good ability to filter suboptimal result.

![Figure 11. The relation between the overlap ratio of three kinds of searching methods and calculating constraint times.](image)

5. Conclusions

This paper proposes an inversion model of the original technology complex network. This skill changes the traditional direct obtaining to multi-step indirect acquiring. By the way of establishing the score matrix, the model solves the problem that it is difficult to describe the social relations according to quantization criterion. We set the technology and social attribute as elements of the node vector and this way help solve the problem that it is difficult to make the attributes of every node vector quantization. Meanwhile, it is convenient to use those algorithms which suitable for map after establishing the complex network relying on the node vector and weights. Using the way of combining of three strategies and variable event classification can accelerate the searching and pruning on the network map. The network consists of node set, path set, and constraint set. The constraint set is very important that the whole network searching will base on it. Path set is the foundation of searching the optimal path, and
the constraint optimizing strategy will also depend on it. During the process of whole inversion, the higher correctness rate is more important than the less searching time. And it is also a very effect way to use overlap ratio as an evaluating criterion during the process of finding the optimal path. Although some effective strategies have been proposed above, but some aspects will be continue improved after this paper such as the improving of constrained criterion, proposing a better search strategy, a better classifying method of constraints, a better way to adapt to the changing of network etc.

Acknowledgments
Our work has been supported by project (U1636208) funded by National Natural Science Foundation of China (NSFC). Authors thank Editors and Reviews for their very constructive feedback. Their insight and comments lead to a better expression of the paper.

References
[1] Sarica S, Luo J and Wood K L 2020 TechNet: Technology semantic network based on patent data Expert Systems with Applications 142 112995.
[2] Ayoubi S, Limam N, Salahuddin M A, et al. 2018 Machine learning for cognitive network management IEEE Communications Magazine 56 (1) 158-165.
[3] Jin J, Ji P and Gu R 2016 Identifying comparative customer requirements from product online reviews for competitor analysis Engineering Applications of Artificial Intelligence 49 61-73.
[4] Calzarossa M C, Massari L and Tessera D 2016 Workload characterization: A survey revisited ACM Computing Surveys 48 (3).
[5] Panda S K and Jana P K 2018 Normalization-based task scheduling algorithms for heterogeneous multi-cloud environment Information Systems Frontiers 20 (4) 1-33.
[6] Li J, Wang Y, Yang S, et al. 2016 A comparative study of constraint-handling techniques in constrained multiobjective evolutionary optimization IEEE Congress on Evolutionary Computation.
[7] Chang C H, Lin N W, Chang C H, et al. 2017 Constraint-based test case generation for white-box method-level unit testing IEEE Computer Symposium.
[8] Li J and Huang Y 2016 A hybrid genetic algorithm for dual-resource constrained job shop scheduling problem Computers & Industrial Engineering 102 113-131.
[9] Baptista L M T 2017 Using restarts in constraint programming over finite domains-An experimental evaluation.
[10] Chuang Y C, Chen C T and Hwang C 2016 A simple and efficient real-coded genetic algorithm for constrained optimization Applied Soft Computing 38 87-105.
[11] Yue L and Chen H 2019 Unmanned vehicle path planning using a novel ant colony algorithm EURASIP Journal on Wireless Communications and Networking 2019 (1).
[12] Santis R D, Montanari R, Vignali G, et al. 2018 An adapted ant colony optimization algorithm for the minimization of the travel distance of pickers in manual warehouses European Journal of Operational Research 267 (1) 120-137.
[13] Sheng X and Li Q 2016 Template-based genetic algorithm for QoS-aware task scheduling in cloud computing International Conference on Advanced Cloud & Big Data (IEEE).
[14] Zhang Z, Hu F and Zhang N 2018 Ant colony algorithm for satellite control resource scheduling problem Applied Intelligence 48 (10) 1-11.
[15] Zheng F, Zecchin A, Newman J, et al. 2017 an adaptive convergence-trajectory controlled ant colony optimization algorithm with application to water distribution system design problems IEEE Transactions on Evolutionary Computation (99) 1-1.
[16] Wang L, Xia X H, Cao J H, et al. 2018 Improved ant colony-genetic algorithm for information transmission path optimization in remanufacturing service system Chinese Journal of Mechanical Engineering 31(1).
[17] Ramadan R M, Gasser S M, El-Mahallawy M S, et al. 2018 A memetic optimization algorithm for multi-constrained multicast routing in ad hoc networks PLoS ONE 13 (3) e0193142.
[18] Maleki S, Nguyen D, Lenharth A, et al. 2016 DSMR: A shared and distributed memory algorithm for single-source shortest path problem *ACM SIGPLAN Notices* **51** (8) 1-2.

[19] Dhulkef E J, Durdu A and Terzioğlu H 2019 A* and Dijkstra algorithm (Engineering Research Papers).

[20] Gayathri R G, Nair J J and Kaimal M R 2016 Extending full transitive closure to rank removable edges in GN algorithm *Procedia Computer Science* **93** 995-1002.