Cluster Formation In An Acyclic Digraph Adding New Edges

Gurami Tsitsiashvili, Marina Osipova

Institute for Applied Mathematics,
Far Eastern Branch of Russian Academy Sciences
guram@iam.dvo.ru, mao1975@list.ru

Abstract

In this paper, we construct an algorithm for converting an acyclic digraph that defines the structure of a complex system into a class of cyclically equivalent vertices by adding several additional edges to the digraph. This addition of the digraph makes it possible to introduce negative feedbacks and, consequently, to stabilize the functioning of the complex system under consideration and so to increase its reliability.To do this, the original digraph is transformed into a bipartite undirected graph, in which only the input and output vertices and the edges between them remain. In the constructed bipartite undirected graph, we search for the minimal edge cover and restore the orientation of the edges in it. Next, we construct an algorithm for adding new edges, based on the search for Hamiltonian (or Eulerian) paths and turning the minimum edge cover into a class of cyclically equivalent vertices. The minimal number of edges to be added is not larger than the number of edges in the minimum edge cover.

Keywords: cluster, maximal matching, increasing alternating path, minimal edge cover, bipartite graph, star, Hamiltonian path, Eulerian path.

I. Problem statement and solution idea

Suppose that a complex system, such as a protein network, is represented by an acyclic digraph G without isolated vertices. In particular, using the algorithm built in [1] for identifying cyclic equivalence classes (clusters) in a digraph, the original digraph is transformed into an acyclic digraph whose vertices are clusters.

Let's call the vertices of the digraph G, from which only the edges come out, input and denote the set of input vertices U_*.Let's call the vertices that only the edges come in, output and denote the set of output vertices U^*. As the digraph G has not isolated vertices so from any input vertex there is a path to some output vertex and to any output vertex there is a path from some input vertex.

Our task is to add new directed edges to the digraph G so that there is a path from any vertex of the resulting digraph to any other vertex. This addition of the digraph makes it possible to introduce negative feedbacks and, consequently, to stabilize the functioning of the complex system under consideration and so to increase its reliability. In a sense, such a problem is the inverse of the digraph clustering problem considered in [1].

In this paper, this is achieved in two stages. At the first stage, we construct a bipartite digraph with vertex fractions U_*, U^* and edges (u_*,u^*), $u_* \in U_*$, $u^* \in U^*$ that are entered if there is a path from vertex u_* to vertex in u^* the digraph G. We remove the orientation of the edges (u_*,u^*), $u_*\in U_*, u^*\in U^*$, from the bipartite graph and find the minimum edge cover [2], [3]. Its connectivity components are star graphs (connected graphs where all edges originate from a single vertex). In the minimal edge cover, we restore the orientation of the edges and denote the resulting bipartite digraph $\hat{G}$.

At the second stage, a minimal set of edges is introduced into the digraph $\hat{G}$, which turns all the vertices into a cluster. To do this, we first add a minimal set of edges in the constructed star graphs that generate a Hamiltonian or Eulerian path with a starting and ending vertices. Then edges are added that connect these paths in a Hamiltonian or Eulerian cycle. All the edges entered in the digraph $\hat{G}$ are added to the original digraph G, turning it into a cluster.

II. Finding feedbacks in a digraph $\hat{G}$

Consider a bipartite digraph $\hat{G}$, represented by a collection of unrelated stars. Let the star G_1 (Figure 1), have a vertex $1_*\in U_*$, as its root, leaves $1^*,\dots,m^*$ and edges $1_*\to 1^*,\dots,1_*\to m^*$. Let's add a minimal set of $m-1$ edges $1^*\to 2^*, 2^*\to 3^*,\dots,(m-1)^*\to m^*$ to this star (coming out of the vertices $1^*,2^*,3^*,\dots,(m-1)^*$), building a Hamiltonian path in it (a simple path that passes through all the vertices once):

$$1_*\to 1^*\to 2^*\to\dots\to(m-1)^*\to m^*.$$

Let's call the vertex 1_* the starting point and the vertex m^* the ending point in this path.

This star can also be supplemented with a minimal set of $m-1$ edges $1^*\to 1_*, 2^*\to 1_*,\dots,$ $(m-1)^*\to 1_*$ (coming out of the vertexes $1^*, 2^*, 3^*,\dots, (m-1)^*$), building an Euler path in it (a path that passes through all the edges once):

$$1_*\to 1^*\to 1_*\to 2^*\to 1_*\to\dots\to(m-1)^*\to 1_*\to m^*.$$

Let us call the vertex 1_* the starting point, and the vertex m^* the ending point in this path.

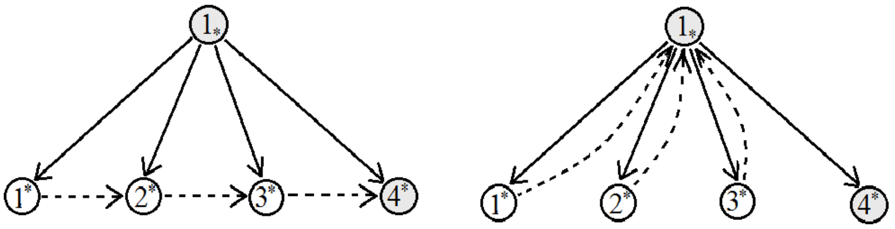

Figure 1. The Hamiltonian (left) and Eulerian (right) paths for the star G_1, $m=4$.

Let the star G_2, (Fig. 2), have a vertex $1^*\in U^*$, leaves $1_*,\dots,n_*$ and edges $1_*\to 1^*,\dots,n_*\to 1^*$. Let`s add this star by minimal set of $n-1$ edges $1^*\to 2_*, 2_*\to 3_*,\dots,(n-1)_*\to n_*$, (included in the vertices $2_*,3_*,\dots,n_*$), building a Hamiltonian path in it:

$$1_*\to 1^*\to 2_*\to 3_*\to\dots\to(n-1)_*\to n_*.$$

Let's call the vertex 1_* as the starting point and the vertex n^* as the ending point in this path.

This star can also be supplemented with a minimal set of $n-1$ edges $1^*\to 2_*, 1^*\to 3_*,\dots,1^*\to n_*$ (included in the vertices $2_*,3_*,\dots,n_*$) building an Eulerian path in it:

$$1_*\to 1^*\to 2_*\to 1^*\to 3_*\to\dots\to n_*\to 1^*.$$

Let us call the vertex 1_* as the starting point and the vertex 1^* as the ending point in this path.

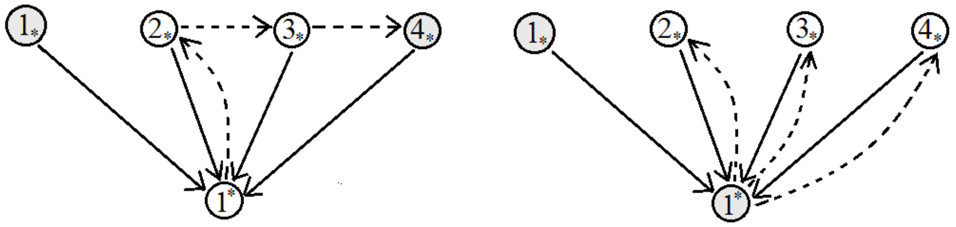

Figure 2. Hamiltonian (left) and Eulerian (right) paths for the star G_2, $n=4$.

Suppose now that the bipartite digraph $\hat{G}$ consists of several stars with roots from the set U_* and with roots from the set U^*. We connect by additional edge the final vertex of the Hamiltonian (Eulerian) path in the first star with the initial vertex of the Hamiltonian (Eulerian) path in the second star, etc. Then the final vertex of the Hamiltonian (Eulerian) path constructed for the last star we connect with the initial vertex of the Hamiltonian (Eulerian) path constructed for the first star.

As a result, we get a Hamiltonian (Eulerian) cycle passing through all the vertices of a bipartite digraph $\hat{G}$ (for an example, see the Hamiltonian cycle in Fig.3). In this case, the number of additional edges is equal to the number of edges in the bipartite digraph $\hat{G}$, which is the minimal edge covering of a bipartite digraph G. In addition to the Hamiltonian or Eulerian cycle, you can build a mixed-type cycle by connecting the Hamiltonian and Eulerian paths in series. Denote $n(\hat{G})$ number of edges in digraph $\hat{G}$, and $N(\hat{G})$ minimal number of new edges, the introduction of which in the digraph $\hat{G}$ leads to the formation of cycles containing all the vertices of the digraph $\hat{G}$ and consequently $N(\hat{G}) \le n(\hat{G})$. If all the stars in the minimal edge cover $\hat{G}$ are of the same type, then

$$N(\hat{G}) = n(\hat{G}). \qquad (1)$$

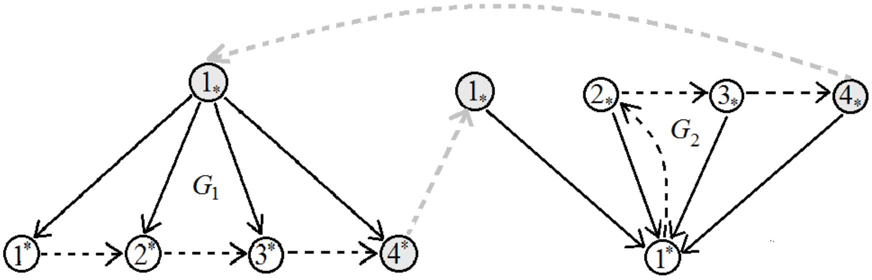

Figure 3. The Hamiltonian cycle for stars G_1, $m=4$, and G_2, $n=4$.

Indeed, let the digraph $\hat{G}$ consists of isolated stars of the type G_1. Then the total number of added edges (marked with dotted lines in Fig. 1 -- 3) coming out of the leaves of these stars in the cluster is equal to the total number of these leaves and cannot be less and so $N(\hat{G}) \ge n(\hat{G})$. Connecting the inequalities $N(\hat{G}) \le n(\hat{G})$and $N(\hat{G}) \ge n(\hat{G})$ we obtain the equality (1).

Similarly, let the digraph $\hat{G}$ consists of isolated stars of type G_2. Then the total number of added edges included in the cluster leaves is the same as the total number of these leaves, so $N(\hat{G}) \ge n(\hat{G})$. Connecting this inequality with the opposite inequality, $N(\hat{G}) \le n(\hat{G})$ we obtain the equality (1).

But if the digraph $\hat{G}$ consists of isolated stars of type G_1 and stars of type G_2 then it is possible to decrease the number of added edges to make this digraph a cluster (see Figure 4). Therefore, in the general case, the ratio between $N(\hat{G})$ and $n(\hat{G})$ is as follows $N(\hat{G}) \le n(\hat{G})$.

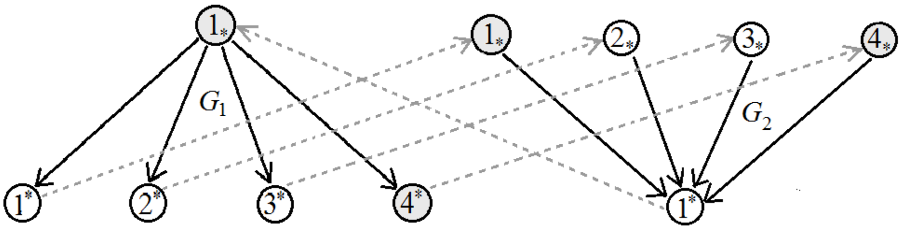

Figure 4. Example when $N(\hat{G}) < n(\hat{G})$ for stars G_1, $m = 4$, and G_2, $n = 4$.

III. Algorithms for finding the minimum edge cover in a bipartite digraph

Following [2] -- [4] to determine the minimum edge cover in an undirected bipartite graph $\overline{G}$, we first find the maximum matching, i.e. the maximum volume set of non-adjacent edges. For each vertex that does not belong to the maximum matching, some edge is selected that connects this vertex to the maximum matching. The maximum matching, together with the so-chosen edges, forms the minimum edge covering. The maximum matching in an undirected bipartite graph can be constructed in the following ways.

One way is to find the maximum flow in the graph $\overline{G}$. By adding the source s and edges from s to all the vertices from U_*, the drain t and edges from all the vertices of the fraction U^* to t. We assign each edge a throughput of one and find the maximum flow between the vertices and sequentially determine the paths that increase the flow. Then the edges between U_* and U^*, on which the flow is equal to one, form the maximum matching.

Another way to find the maximum matching is based on the construction of increasing alternating paths. Let some matching in the graph $\overline{G}$ be given (for example, one edge). We will call the edges of the matching strong, and the other edges of the graph weak. A vertex is called free if it does not belong to a matching. An alternating path is a simple path in which strong and weak edges alternate (i.e., a strong edge is followed by a weak one, and a weak one is followed by a strong one). An alternating path is called an increasing path if it connects two free vertices. If there is such a path relative to a given match, then you can build a larger match. By turning weak edges into strong ones, and strong edges into weak ones, we increase the number of matching edges by one. A match is maximal if and only if there are no increasing alternating paths relative to it.

References

[1]. Tsitsiashvili G. Sh. (2013). Sequential algorithms of graph nodes factorization. *Reliability: Theory and Applications,* 8 (4): 30-33.

[2]. Alekseev V. E., Zakharova D. V. Graph Theory: A textbook. Nizhny Novgorod: Nizhny Novgorod State University, 2017. (In Russian).

[3]. Cormen T. H., Leiserson Ch. E., Rivest R. L., Stein Cl. Introduction to Algorithms, 3rd Edition. The MIT Press, 2009.

[4]. Ford L. R., Fulkerson D. R. (1956). Maximal flow through a network. *Canadian Journal of Mathematics*, 8: 399–404.