2 Basic Definitions and Notations

A directed graph G(V, E) consists of a finite non-empty set V of vertices (nodes) and a finite non-empty set E of edges. A directed graph G(V, E) has |V|= n nodes and |E|= m edges. We have assumed that each node V_i, 1 ≤ i ≤ n, has weight 1 and each edge e_j, 1 ≤ j ≤ m has also weight 1. A edge e_j is said to be incident to the node V_i if $V_i \in e_j$. If an edge e_j is incident to a node V_i, then we say that e_j is on V_i, or V_i is on e_j. Nodes on an edge are called its terminals, and edges on a node are called its pins. Nodes that share terminals are called neighbor nodes. The degree d_i of a node V_i is equal to the number of edges incident to V_i. The maximum node degree d_{max} is the biggest value of node degree among all nodes of set V. The degree $|e_j|$ of an edge e_j is equal to the number of its terminals. For graph, each edge has 2 terminals. Thus, for graph, degree $|e_j|$ of each edge is 2. The total number p of pins, or the total number of terminals, in G(V, E) is defined as

$$\mathbf{p} = \sum_{j=1}^{m} |\mathbf{e_j}| = \sum_{i=1}^{n} \mathbf{d_i} = \mathbf{2 \times m} \tag{1}$$

and is taken as the size of the graph. The density D of directed graph G(V, E) with n ≥ 2 is defined as

$$\mathbf{D} = \frac{\sum_{j=1}^{m}(|\mathbf{ej}|-1)}{n(n-1)} = \frac{m}{n(n-1)} \tag{2}$$

The density of a graph determines how sparse the graph is, and we say that the smaller the density of a graph, sparser the graph.

Given a directed graph G(V, E), we say that π = (P_1, • • • , P_k) is a k-way partition of G(V, E) if the following three properties hold: each part P_i, 1 ≤ i ≤ k, is a subset of V, parts are pairwise disjoint, and the union of k parts is equal to V. A k-way partition is also called a multiway partition if k > 2, and a bipartition if k = 2.

Consider a k-way partition π = (P_1, • • • , P_k) of a graph G(V, E). The size w(P_i) of a part P_i is equal to the total number of nodes in P_i. An edge that has at least one pin in a part is said to connect that part. An edge that connects more than one part is said to be cut, otherwise uncut. The set E(i) of external edges of a part P_i is defined as E(i) = { e_j ϵ E | e_j ∩ P_i ≠ ∅ ^ e_j - P_i ≠ ∅ }, which consists of those cut edges that connect P_i. The set I(i) of internal edges of a part P_i is defined as I(i) = {e_j ϵ E | e_j ∩ P_i ≠ ∅ ^ e_j - P_i = ∅ }, which consists of those uncut edges that connect only P_i. The cost x($\boldsymbol{\pi}$) of a partition $\boldsymbol{\pi}$, also called the cut-size, is equal to the number of cut edges. More formally, x($\boldsymbol{\pi}$) = total number of edges connected to partition $\boldsymbol{\pi}$ - total number of internal edges of partition π (i.e. I(π)). Each cut edge e_j contributes an amount of 1 to the cut-size regardless of the number of parts that e_j connects.

2.1 Concept of Cost, Gain and Cut-size

The cutstate of an edge indicates whether the edge is cut or uncut. The cutset of a partition is the set of all edges that are cut. Notice that an edge in the cutset must be in the set of external edges of at least two parts. An edge is critical if it has a node such that the node would change the cutstate of the edge if it is moved. Such a move either adds the edge to the cutset or removes the edge from the cutset. We now give the necessary and sufficient condition for an edge to be critical in a k-way partition.

Proposition 2.1 An edge e_j is critical if and only if either there exists a part P_s such that j(s) =2 or there exist two different parts P_s and P_t such that j(s) = 1 and j(t) = 1, where j(s) and j(t) are the number of terminals of edge e_j that lie in the partition P_s and P_t respectively. The cutstate of an edge that is not critical cannot be affected by a node move by the definition of a critical edge, and so such a move cannot have any effect (a decrease or an increase) on the cut-size. A move of a node can change the cut-size if the node removes some edges from the cutset, or adds some edges to the cutset, that is, if it alters the cutstate of some edges, which should then be critical edges. Hence, we proved the following proposition.

Proposition 2.2 The effect of a move of a node on the cutsize depends only on the critical edges incident to that node. We reflect the effect of a node in the cut-size in terms of its gains, but the gains of a node depend on its costs. Let P_s and P_t be two parts. The cost C_i (s, t) of a node V_i in P_s with respect to P_t is called its external cost if s≠t and is defined as

$$\mathbf{C_i\,(s,\,t) = |E_i\,(s,\,t)|} \tag{3}$$

where the set E_i (s, t) is the subset of external edges of P_s that would be deleted from the cutset if V_i is moved from P_s to P_s. Hence, the external cost C_i (s, t) of a node V_i in P_s with respect to P_t is equal to the number of edges present in the set E_i (s, t). The cost C_i (s, t) of a node V_i in P_s is called its internal cost if s = t, and is defined as

$$\mathbf{C_i\,(s,\,s) = |I_i\,(s)|} \tag{4}$$

where the set I_i (s) is the subset of internal edges of P_s that would be added to cutset if V_i is moved from P_s to any other part. Hence, the internal cost C_i (s, s) of a node V_i in P_s is equal to the number of edges present in the set I_i (s). Since V_i can change the cutstate of edges in both E_i (s, t) and I_i (s), those edges are all critical edges. In a k-way partition, each node has only one internal cost but (k- 1) external costs, each of which corresponds to a move direction towards remaining (k-1) parts. The move gain (or gain) G_i (s, t) of a node V_i in P_s with respect to P_t, i.e., the gain of the move of V_i from P_s to P_t, is defined as

$$\mathbf{G_i\ (s, t) = C_i\ (s, t) - C_i\ (s, s)} \qquad \mathbf{(5)}$$

where s ≠ t. Note that each node has (k – 1) move gains in a k-way partition. The maximum move gain is denoted by G_{max}, and is equal to the product of the maximum node degree and the maximum edge weight. i.e. G_{max} = d_{max} × 1 = d_{max} , where maximum edge weight =1. All the gains fall in the interval [-G_{max} , G_{max}]. Proposition 2.3 Consider the move of node from P_s to P_t. Let the cut-size before and after the move be denoted by x(π) and x'(π), respectively. Then

$$\mathbf{x'(\pi) = x(\pi) - G_i\ (s, t)} \qquad \mathbf{(6)}$$

where G_i (s, t) is the move gain of V_i before the move. As this proposition shows, the gain of a node determines the amount of benefit to be obtained by moving that node. If the gain is positive, the cut-size decreases, but if the gain is negative, it increases.

2.2 Concept of convex and non-convex partition

A partition P_{cp} ∈ π is said to be a convex partition if and only if for each path p_{AB} between node A ∈ P_{cp} and node B ∈ P_{cp} , there exist a node C ∈ p_{AB} then C ∈ P_{cp}, where A ≠ B, C ≠ A and C ≠ B. If a partition P_{ncp} ∈ π doesn't follow above condition then the partition P_{ncp} is known as non-convex partition. The union of all convex partitions is represented as convex partition set P_c and the union of all non-convex partitions is represented as non-convex partition set P_{nc} . Thus, we can say that union of set P_c and set P_{nc} represents the partition set π which is given as

$$\mathbf{P_c \cup P_{nc} = \pi} \qquad \mathbf{(7)}$$

2.3 Concept of Input and Output node

The input node v_{ip} of partition P_i is a node which has either no incoming edge or if an edge e_{ip} ∈ E is incoming on v_{ip} then there doesn't exist a node v_t ∈ P_i for which e_{ip} is outgoing edge. Similarly, the output node v_{op} of partition P_i is a node which has either no outgoing edge or if an edge e_{op} ∈ E is outgoing from v_{op} then there doesn't exist a node v_t ∈ P_i for which e_{op} is incoming edge. Union of all input nodes of partition set π is represented as input set v_i ∈ V and union of all output nodes of π is represented as output set v_o ∈ V. We now define the graph covering problem as follows.

Given a big directed graph G(V, E) or application graph and a small directed graph R(V′, E′) or resource graph, we initially produce a k-way partition set π = (P_1, • • •,P_k as the initial solution in which the size w(P_i) of each part P_i, 1 ≤ i ≤ k, satisfies the balancing constraint L(P_i) ≤ w(P_i) ≤ U(P_i). Here, L(P_i) and U(P_i) are integral lower and upper bounds of size of P_i, respectively. Then we minimize cut-size of partition set π as well as longest depth L_{nc} of non-convex partition set P_{nc} ∈ $\boldsymbol{\pi}$ while maintaining the balancing constraints. All three problems, the multiway graph partitioning problem, convexity detection problem and longest path problem fall under the category of NP-hard problem. We say that a partition is acceptable if it satisfies the given balance criterion as well as longest depth L_{nc} of non-convex partition set P_{nc} ∈ $\boldsymbol{\pi}$ ≤ longest depth L_c of convex partition set P_c ∈ $\boldsymbol{\pi}$ and unacceptable otherwise. A partition is said to be balanced if the part sizes are in interval [0, n′], and unbalanced otherwise. A partition in which the part sizes are the same as the size of resource graph n′ is called a perfectly balanced partition, but such partition is difficult in some cases because value of n mod n′ may not be 0 and it also depends upon initial random partitioning.