On Minimizing the Maximum Congestion for Weighted Hypergraph Embedding in a Cycle

SingLing Lee Hann-Jang Ho
Department of Computer Science and Information Engineering
National Chung-Cheng University
Chiayi 621, Taiwan, R.O.C.
{singling,hjho}@cs.ccu.edu.tw

Abstract

The problem of Weighted Hypergraph Embedding in a Cycle (WHEC) is to embed the weighted hyperedges of a hypergraph as the paths in a cycle, such that the maximum congestion of any physical link in the cycle is minimized. A simpler version of this problem is the Weighted Graph Embedding in a Cycle (WGEC) that embeds the weighted edges of a normal graph as the paths in a cycle. The WHEC and WGEC problems have applications in design automation, parallel computing and computer communication. In this paper, we show that both WHEC and WGEC problems are NP-Complete. Also, we formulate WHEC problem as an integer linear programming (ILP). Therefore, the approximation solution can be obtained by using LP-relaxation and rounding heuristic. Our LP-approximation algorithm generates an embedding with congestion at most two times the optimal solution.

Index Terms: *Embeding, Hypergraph, LP-Relaxation, NP-Complete.*

1 Introduction

The problem of Minimum-Congestion Hypergraph Embedding in a Cycle (MCHEC) was proposed by Ganley and Cohoon [1]. In MCHEC, we wish to embed the m-hyperedges of an n-vertex hypergraph as paths in an n-vertex cycle, such that the maximum number of paths that use any physical link in the cycle is minimized. Note that the vertex set of the hypergraph actually is same as the vertex set of the cycle. This problem has applications in electronic design automation, specifically in routing nets around a rectangle [2, 3, 4] and moat routing [5, 6]. Both applications model the process of connecting a concentric rectangle to a surrounding cycle of pads. Since there is a number of nets whose pins must connected to the padframe by wires, we would like to route the nets to consume the space between the concentric rectangle and the padframe as little as possible. In other words, the objective of these applications is to realize an optimal area routing that minimizes the width of the moat. Although this objective is different from the minimum-congestion, they can be still formulated as the MCHEC problem [5].

Several studies related to the MCHEC problem have been performed. Canley and Cohoon [1] showed that the MCHEC problem is NP-complete in general, but solvable in polynomial time when the congestion is at most k. For any fixed k, a solution can be computed in $O((nm)^{k+1})$ time. They also proposed an approximation algorithm that is guaranteed to find a solution within a factor of three times the optimal solution. Moreover, they formulated the moat routing as an MCHEC problem and proposed a modified version of the approximation algorithm to provide solutions within a factor of four times optimum [5]. Subsequently, Gonzalez [7] proposed two improved approximation algorithms that both generate solutions within two times the optimal solution. The first algorithm transforms the MCHEC problem to a linear programming formulation, and the other one solves the problem in linear time by transforming hyperedges to the normal edges. In the meantime, Carpenter et al. [8] provided a very simple approximation algorithm which runs in linear time and provides solutions that are guaranteed to be within two times the optimal solution. This algorithm just routes the hyperedges in the clockwise direction starting from the lowest numbered vertex to the highest numbered vertex. Moreover, a simple version of EHCMC problem, in which all hyperedges contain exactly only two vertices, is solvable in polynomial time. Frank et al. [9] developed a linear time algorithm for the about problem. Note that previous work [2] had been studied a similar problem and also proposed a linear time algorithm that routes around a rectangle to minimize the enclosing area. However, the problem is different from the EHCMC problem in that a minimum-area routing typically does not have minimum congestion.

In this paper, we are concerned with the problem of Weighted Hypergraph Embedding in a Cycle (WHEC). This problem is a weighted version of MCHEC problem and the objective is to embed the weighted hyperedges of a hypergraph as the paths in a cycle, such that the maximum congestion of any physical link in the cycle is minimized. The WHEC problem has applications in parallel computing and computer communication on a ring network. Here, each vertex of the hypergraph represents an individual processor in the ring, the hyperedge of the hypergraph indicates a data channel between a set of processors that need to communicate with one another, and the weight of the hyperedge represents the traffic flow over the data channel. The optimal solution to the WHEC problem specifies a routing for inter-processor communications that minimize the maximum total flow on any physical link in the ring network. This optimality criterion is motivated by the fact that a single heavily congested link is a bottleneck in the network. On the other hand, we also consider a simpler version of the WHEC problem, where all hyperedges contain exactly only two vertices. The problem of Weighted Graph Embedding in a Cycle (WGEC) can be solved in polynomial time [9] when the edges of the graph are unweighted. Nevertheless, we will show that the WGEC problem is NP-complete when the edges of the graph are weighted.

The remainder of this paper is organized as follows: Section 2 formally describes the WHEC and WGEC problems and proves that they are NP-complete. Section 3 shows that the WHEC can be treated as a problem of integer linear programming and then solved by using LP-relaxation and rounding heuristic. Conclusions are given in Section 4.

2 Notation and NP-Completeness

In this section, we formally define the WHGE and WGEC problems, and prove the NP-completeness of the problems. Note that the following notation will also be used throughout the paper. First of all, a cycle with n vertices is denoted as $C = (V, E_c)$, where $V = \{1, 2, \cdots, n\}$ is a set of vertices and $E_c = \{(i, i+1) | 1 \le i \le n-1\} \bigcup \{(n, 1)\}$ is a set of undirected links. We refer an edge in the cycle as an undirected link and the vertices are labeled around the cycle from 1 to n in the clockwise direction (see Fig. 1). Secondly, a hypergraph $H = (V, E_h)$ with m weighted hyperedges is defined over the same set of vertices V, where $E_h = \{h_1, h_2, \cdots, h_m\}$ is a set of hyperedges. The hyperedge h_i consists of $|h_i|$ vertices with a non-negative weight w_i for interconnecting these vertices. In particular, these interconnected vertices of hyperedge h_i is represented as an ordered sequence $(v_1^i, v_2^i, \cdots, v_{|h_i|}^i)$, i.e. $v_1^i \le v_2^i \le \cdots \le v_{|h_i|}^i$. Finally, we define the j-th adjacent path of hyperedge h_i as p_j^i, where the p_j^i is a clockwise connecting path between v_j^i and v_{j+1}^i in the cycle C, and the last adjacent path $p_{|h_i|}^i$ of the hyperedge h_i is connected between $v_{|h_i|}^i$ and v_1^i. We define an assignment of adjacent paths to embed the hypergraph H in the cycle C as a set of binary variables $Y = \left[y_{p_j^i}\right]$, where $y_{p_j^i} = 1$ if an adjacent path p_j^i is embedded in the cycle C, and $y_{p_j^i} = 0$ otherwise. Obviously, all the vertices $v_1^i, v_2^i, \cdots, v_{|h_i|}^i$ in the hyperedge h_i can be continuously interconnected by embedding at least $|h_i| - 1$ of its adjacent path (see Fig. 1). Therefore a feasible assignment of adjacent paths can be expressed as $\sum_{1 \le j \le |h_i|} y_{p_j^i} \ge |h_i| - 1, \forall h_i \in E_h$. Moreover, we denote $P(e)$ as a set of adjacent paths that pass through the link $e \in E_c$. Now, the WHEC and WGEC problems can be formally defined as follows:

Definition 1 *The problem of minimizing the maximum congestion for Weighted Hypergraph Embedding in a Cycle (WHEC):*
Given: *A cycle $C = (V, E_c)$ and a weighted hypergraph $H = (V, E_h)$.*

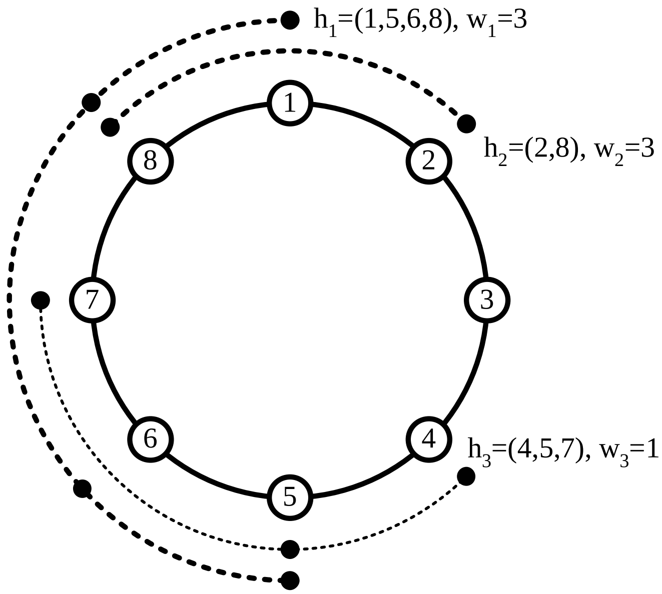

Figure 1: A weighted hypergraph $H = \{h_1, h_2, h_3\}$ is embedded in an 8-vertex cycle. The link $(8, 1)$ has the maximum-congestion of 6 that is an optimal solution.

Question: *Find a feasible assignment* $Y = \left[y_{p^i_j}\right]$ *of adjacent paths to embed the hypergraph* H *in the cycle* C *such that the maximum congestion* $\Phi(Y)$ *over the links* E_c *is minimized. Note that the maximum link congestion of an assignment* Y *can be expressed as* $\Phi(Y) = \max_{e \in E_c} \left\{ \sum_{p^i_j \in P(e)} y_{p^i_j} w_i \right\}$ *, where* $y_{p^i_j} = 1$ *if the* p^i_j *is embedded in the cycle* C*, otherwise* $y_{p^i_j} = 0$.

Definition 2 *The problem of minimizing the maximum congestion for Weighted Graph Embedding in a Cycle (WGEC):*
Given: *A cycle* $C = (V, E_c)$ *and a weighted graph* $H = (V, E_h)$ *where each edge in* E_h *contains a subset of only two vertices, i.e.* $|h_i| = 2, \forall h_i \in E_h$.
Question: *Find a feasible assignment* $Y = \left[y_{p^i_j}\right]$ *of adjacent paths to embed the weighted graph* H *in the cycle* C *such that the maximum congestion* $\Phi(Y)$ *over the links in* E_c *is minimized.*

When the edges of the embedded graph are unweighted, the WGEC problem can be solved in polynomial time by using the Okamura-Seymour planar multicommodity flow theorem [9]. We now show that the following DECISION-WGEC problem is NP-complete if each edge of the embedded graph can contain an arbitrary weight.

Definition 3 *The DECISION-WGEC problem :*
Given: *A cycle* $C = (V, E_c)$*, a weighted graph* $H = (V, E_h)$ *with* $|h_i| = 2, \forall h_i \in E_h$*, and a congestion bound* k.
Question: *Find a feasible assignment* $Y = \left[y_{p^i_j}\right]$ *of adjacent paths to embed the weighted graph* H *in the cycle* C *such that the maximum congestion* $\Phi(Y)$ *over the links in* E_c *is at most* k*, i.e.* $\Phi(Y) \le k$.

Theorem 1 *The DECISION-WGEC problem is NP-complete.*

Proof: We first show that the DECISION-WGEC problem belongs to NP. Suppose we are given a cycle $C = (V, E_c)$, a weighted graph $H = (V, E_h)$ with $|h_i| = 2, \forall h_i \in E_h$, and a congestion bound k. To show that the problem can be solved by non-deterministic turing machines in polynomial time, we need only to guess an assignment $Y' = \left[y'_{p^i_j}\right]$ of the adjacent paths. Now, checking

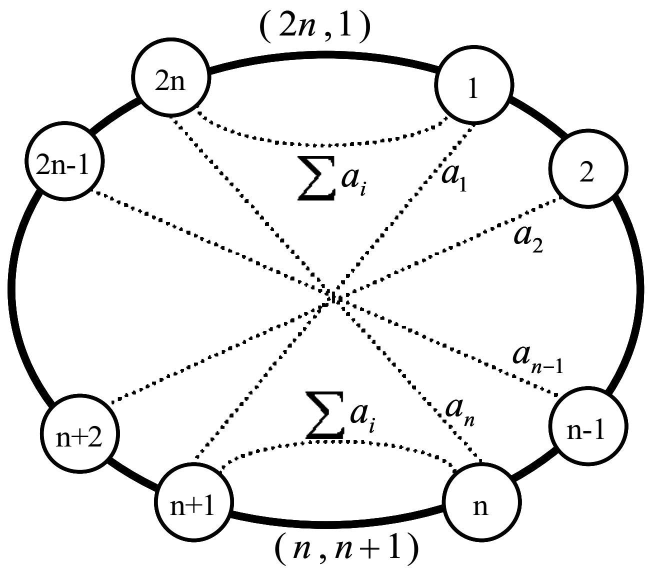

Figure 2: Reducing PARTITION to DECISION-WGEC.

whether Y' is a feasible embedding can be accomplished in polynomial time by verifying whether, for every edge $h_i \in E_h$, at least one adjacent path is assigned to embed in the cycle C, i.e. $y'_{p_1^i} + y'_{p_2^i} \geq 1$, $\forall i$. On the other hand, checking whether the link congestion $\sum_{p \in P(e)} y'_p$, $e \in E_c$ is within the bound of k can also be verified in polynomial time. We next show that the DECISION-WGEC problem is NP-hard by proving that PARTITION $\propto$ DECISION-WGEC. As widely known, PARTITION is NP-complete and requires us to decide whether a given instance $A = \{a_1, a_2, \cdots, a_n\}$ of n positive integers has a partition B such that $\sum_{i \in B} a_i = \sum_{i \notin B} a_i$. The PARTITION problem is polynomially transferable to the subproblem of DECISION-WGEC. For an instance $A = \{a_1, a_2, \cdots, a_n\}$ of PARTITION problem, we construct a DECISION-WGEC instance of $C' = (V', E'_c)$, $H' = (V', E'_h)$ and $k' = \sum_{1 \leq i \leq n} 3a_i/2$, where $V' = \{1, 2, \cdots, 2n\}$ with $2n$ vertices, $E'_c = \{(1,2), (2,3), \cdots, (2n-1, 2n), (2n, 1)\}$ with $2n$ physical links and $E'_h = \{(1, n+1), (2, n+2), \cdots, (n-1, 2n-1), (n, 2n)\} \bigcup \{(n, n+1), (1, 2n)\}$ with $n+2$ embedding edges. For each item a_i in A, we assign a weight a_i to the embedding edge $(i, n+i)$ and the sum of weights $\sum_{i \in A} a_i$ is assigned to edges $(n, n+1)$ and $(1, 2n)$ as shown in Fig. 2. This construction can easily be transferred from A in polynomial time. Afterwards, we claim that A has a partition B with $\sum_{i \in B} a_i = \sum_{i \notin B} a_i$ if and only if the graph $H' = (V', E'_h)$ has an embedding in the cycle $C' = (V', E'_c)$ with a congestion bound $k' = \sum_{1 \leq i \leq n} 3a_i/2$.

($\Rightarrow$) Each edge $(i, n+i)$ with weight a_i is necessary embedded by an adjacent path to go either through link $(n, n+1)$ or link $(2n, 1)$, hence the maximum congestion occur certainly in ether link. Suppose that A has a partition B with $\sum_{i \in B} a_i = \sum_{i \notin B} a_i$. Each edge $(i, n+i)$ with $i \in B$ passes through the link $(n, n+1)$ and each edge $(i, n+i)$, $i \notin B$ passes through the other link $(2n, 1)$. Therefore, the maximum congestion is bounded by $k' = \sum_{1 \leq i \leq n} 3a_i/2$.

($\Leftarrow$) Conversely, suppose that the graph H' has an embedding with $k' = \sum_{1 \leq i \leq n} 3a_i/2$. We assign $B = \{\ i \mid \text{edge } (i, n+i) \text{ passes through link } (n, n+1)\ \}$, then A has a partition B with $\sum_{i \in B} a_i = \sum_{i \notin B} a_i$. Finally, since the construction can be done in polynomial time, we conclude that

PARTITION $\propto$ DECISION-WGEC. □

Since the WGEC problem is a subproblem of WHEC, we have the following corollary:

Corollary 1 *Both WGEC and WHEC problems are NP-complete.*

3 An LP-based approximation algorithm

In this section, an integer linear programming (ILP) model is derived to solve the WHEC problem. An assignment of adjacent paths to embed the hypergraph H in the cycle C can be representd by a set of binary variables $Y = \left[y_{p^i_j}\right]$, where $y_{p^i_j} = 1$ if an adjacent path p^i_j is embedded in the cycle C, and $y_{p^i_j} = 0$ otherwise. Our objective function is to minimize the maximum congestion $\Phi(Y)$ over the links E_c. Expanding the LP formulation of the EHCMC problem in [7], a new ILP formulation of the WHEC problem can be expressed as follows:

(NILP) Minimize Φ
Subject to

Connectivity Constraints:
$\sum_{1 \le j \le |h_i|} y_{p^i_j} \ge |h_i| - 1$, $\forall h_i \in E_h$, $i \in \{1, 2, \cdots, m\}$
Capacity Constraints:
$\sum_{p^i_j \in P(e)} y_{p^i_j} w_i \le \Phi$, $\forall e \in E_c$
Binary Variables:
$0 \le y_{p^i_j} \le 1$ integer, $\forall i \in \{1, 2, \cdots, m\}, j \in \{1, 2, \cdots, |h_i|\}$

There are two classes of constraints in this problem. First, the connectivity constraints mean that each hyperedge $h_i \in E_h$ must be connected by at least $m_i - 1$ adjacent paths. Secondly, the capacity constraints ensure that each physical link $e \in E_c$ can only be embedded at most the maximum congestion Φ of weighted edges. Finally, the objective of the NILP formulation is to minimize the maximum congestion Φ. Note that, for all $i \in \{1, 2, \cdots, m\}$, NILP with $w_i = 1$ is the MCHEC problem, NILP with $|h_i| = 2$ is the WGEC problem. Both MCHEC and WGEC problems are known as NP-complete. Nevertheless, NILP with $w_i = 1$ and $|h_i| = 2$ is solvable in polynomial time.

Suppose that we relax the integer constraints of Y and require only that $0 \le y_{p^i_j} \le 1, \forall i, j$; this linear program is called the LP relaxation of the NILP formulation, and the LP solution immediately provides a lower bound on the minimum congestion. Moreover, finding a tight upper bound for the NILP formulation presents a different challenge. We apply a rounding approach to generate an approximate solution as follows:

Algorithm 1 *The NILP-based rounding algorithm.*

Step (1). *Solve optimally the LP relaxation of the NILP formulation. Let the optimal solution be $\Phi(Y^L)$, where $Y^L = \left[y^L_{p^i_j}\right]$ and $0 \le y^L_{p^i_j} \le 1$.*

Step (2). *Let $y^R_{p^i_j} = 1$ if $y^L_{p^i_j} \ge 1/2$, and $y^R_{p^i_j} = 0$ otherwise.*

Step (3). *Output the approximate solution $\Phi(Y^R)$ of the maximum congestion, where $Y^R = \left[y^R_{p^i_j}\right]$.*

Corollary 2 *Let $\Phi(Y^*)$ be the optimal solution of the NILP formulation. We have $\Phi(Y^L) \le \Phi(Y^*) \le \Phi(Y^R)$.*

Theorem 2 *The maximum congestion of the NILP-based rounding algorithm is at most two times the optimum for the MCHEC, WGEC and WHEC problems.*

Proof: Obviously, the WHEC problem represents an MCHEC problem when $\forall w_i = 1$ and a WGEC problem when $\forall |h_i| = 2$. Hence, we need only to show that the NILP-based rounding algorithm has at most two times the maximum congestion for the WHEC problem. First, we show that $\Phi(Y^R)$ is a feasible solution to the WHEC problem. According to connectivity constraints, i.e. $\sum_{1 \le j \le |h_i|} y^L_{p^i_j} \ge |h_i| - 1$, any solution to LP relaxation must be satisfy that at least $|h_i| - 1$ of the $y^L_{p^i_j}$ have value at least 1/2. Then, for each value of i, the sum of rounded values $\sum_{1 \le j \le |h_i|} y^R_{p^i_j}$ is at least $|h_i| - 1$. Each hyperedge h_i is continuously connected, hence the solution $\Phi(Y^R)$ is feasible. Next, we show that $\Phi(Y^R) \le 2\Phi(Y^*)$. From the capacity constraints, we have $\sum_{p_j^i \in P(e)} y_{p_j^i}^L w_i \leq \Phi(Y^L)$. Since $y_{p_j^i}^L$ is rounded to $y_{p_j^i}^R$ and connectivity constraints guarantee that at least $|h_i| - 1$ of the $y_{p_j^i}^L$ have values at least 1/2. We have $y_{p_j^i}^R = \lceil y_{p_j^i}^L \rceil \leq 2y_{p_j^i}^L$. Therefore, $\sum_{p_j^i \in P(e)} y_{p_j^i}^R w_i \leq 2 \sum_{p_j^i \in P(e)} y_{p_j^i}^L w_i, \forall e \in E_c$. Moreover, from Corollary 2, we conclude that $\Phi(Y^R) \leq 2\Phi(Y^L) \leq 2\Phi(Y^*)$. $\square$

4 Conclusion

This paper concerns with the problem of Weighted Hypergraph Embedding in a Cycle (WHEC) and focuses on the impact of the maximum congestion. We have shown that even when hyperedges contain exactly two vertices, the WGEC problem is NP-complete. An ILP formulation for the WHEC problem is also derived to minimize the maximum congestion. By using LP relaxation and rounding, we propose an LP-approximation algorithm which generates an embedding with congestion at most twice of that in the optimal solution.

Acknowledgement

The authors would like to thank Prof. T. Gonzalez for the helpful discussions.

References

[1] J. L. Ganley and J. P. Cohoon, Minimum-congestion hypergraph embedding in a cycle, *IEEE Trans. Comput.* 46 (5) (1997) 600-602.

[2] T. Gonzalez and S. L. Lee, A linear time algorithm for optimal wiring around a rectangle, *J. ACM* 35 (4) (1988) 810-832.

[3] T. Gonzalez and S. L. Lee, A 1.60 approximation algorithm for routing multiterminal nets around a rectangle, *SIAM J. Comput.* 16 (4) (1987) 669-704.

[4] M. Sarrafzadeh and F. P. Preparata, A bottom-up layout technique based on two-rectangle routing, *Integration: VLSI J.* 5 (1987) 231-246.

[5] J. L. Ganley and J. P. Cohoon, A provably good moat routing algorithm, *Proc. Sixth Grate Lakes Symp. VLSI* (1996) 86-91.

[6] C. C. Tsai and S. J. Chen, A linear time algorithm for planner moat routing, *J. Inform. Sci. Eng.* 10 (1) (1994) 111-128.

[7] T. Gonzalez, Improved approximation algorithms for embedding hyperedges in a cycle, *Infomation Processing Letters* 67 (4) (1998) 267-271.

[8] T. Carpenter, S. Cosares, J. L. Ganley and I. Saniee, A simple approximation algorithm for two problems in circuit design, *IEEE Trans. Comput.* 47 (11) (1998) 1310-1312.

[9] A. Frank, T. Nishizeki, N, Saito, H. Suzuki, E. Tardos, Algorithms for routing around a rectangle, Discrete Appl. Math. 40 (3) (1992) 363-378.