The Eighth International Symposium on Operations Research and Its Applications (ISORA'09)
Zhangjiajie, China, September 20–22, 2009


# Approximation Algorithms for Embedding a Weighted Directed Hypergraph on a Mixed Cycle*

Jianping Li[1] Weidong Li[1] Jianbo Li[2]
Juanping Zhu[1]

[1]Department of Mathematics, Yunnan University, Kunming 650091, PR China
[2]School of Management and Economics, Kunming Uni. of Sci. and Tech., PR China

**Abstract** Given a weighted directed hypergraph $H = (V, E_H; w)$, where $w : E_H \to R^+$, we consider the problem of embedding all weighted directed hyperedges on a mixed cycle, which consists of undirected and directed links. The objective is to minimize the maximum congestion of any undirected or directed link in the mixed cycle. In this paper, we first formulate this new problem as an integer linear program, and by utilizing a nontrivial LP-rounding technique, we design a 2-approximation algorithm. Then, we design a combinatorial algorithm with approximation ratio 3 for the problem, whose running time is $O(nm)$. Finally, we present a polynomial time approximation scheme (PTAS) for the special version where each directed hyperedge only contains one sink.

**Keywords** Mixed cycle; Directed hypergraph embedding; LP-rounding; Approximation algorithm; PTAS

## 1 Introduction

Due to the extensive applications in various areas such as computer networks, multicast communication, parallel computation, electronic design automation, the hypergraph embedding problem received more and more attention in past twenty years. The most important problem, called as weighted hypergraph embedding in a cycle (WHEC), is to embed hyperedges of the hypergraph $H = (V, E_H; w)$ as adjacent paths of a cycle $C = (V, E)$ such that the maximum congestion of any physical link on the cycle is minimized.

For the WHEC problem, the hyperedges of the hypergraph and the links in a cycle are both undirected. For the version where all hyperedges are unweighted, Ganley and Cohoon [4] proved that the WHEC problem is $NP$-hard and gave a 3-approximation algorithm. Genzalez [5] designed two improved 2-approximation algorithms for the WHEC

*The work is fully supported by the National Natural Science Foundation of China [No.10861012], Natural Science Foundation of Yunnan Province [No.2006F0016M] and Foundation of Younger Scholar in Science and Technology of Yunnan Province [No.2007PY01-21].
E-mail addresses: jianping@ynu.edu.cn (J. Li), jeeth@126.com (W. Li), lijianbo@public.km.yn.cn (J. Li), zhujuanping@126.com (J. Zhu)

problem. Gu and Wang [6] presented a 1.8-approximation algorithm by a reembedding approach. Recently, Deng and Li [2] proposed a polynomial time approximation scheme (PTAS) by using randomized rounding technique. For the weighted version, Lee and Ho [9] first proposed an LP-based rounding algorithm and then developed a linear-time approximation algorithm to provide an embedding with congestion at most two times the optimum. The same authors [7] recently designed an $1.5+\varepsilon$-approximation algorithm for any $\varepsilon > 0$. The well-known question whether there exists a PTAS for the WHEC problem is still open. For each hyperedge only consisting of two nodes, the WHEC becomes the well-known ring loading problem, and some classical results can be found in [8, 12].

Similar to the WHEC problem, Li and Wang [10] first defined a directed hyperedge in the following way: a directed hyperedge $h = (u,S)$ is a pair, where $u \in V$ is indicated as the *source* of the directed hyperedge $h$ and $S \subseteq V - \{u\}$ is the set of *sinks*. In communication applications, each directed hyperedge $h$ represents a request that asks to send a message from $u$ to every vertex in $S$. Afterward, they considered the directed hypergraph embedding in a cycle (DHEC), where the hyperedges of the hypergraph and the links in the cycle are directed (this definition can be found in [10] or in the next section), and they present a PTAS to solve the DHEC problem by extending the randomized rounding method in [2]. But the *NP*-hardness of the DHEC problem is still open. For each directed hyperedges only consisting of two nodes, the DHEC becomes the well-known directed ring loading problem, and some related results can be found in [1, 11].

In this paper, we introduce a new cycle model, called *mixed cycle*, which contains some undirected links and the other bidirected links in the cycle such that this model generalizes both the undirected cycle and directed cycle model. To the best of our knowledge, there are no related results about such a mixed cycle model. We consider a new problem, called as weighted directed hypergraph embedding in a mixed cycle (WDHEMC). The objective is to embed all directed hyperedges of the hypergraph $H = (V,E_H;w)$ as adjacent "paths" of a mixed cycle $C = (V,E,A)$ such that the maximum congestion of any undirected link or directed link in the mixed cycle is minimized.

We can construct an integer linear programming to present the WDHEMC problem, and we know the fact that the basic LP-rounding technique for the undirected version in [5, 9] can not be extended to the WDHEMC problem. By utilizing a nontrivial LP-rounding technique, we obtain a new LP-based 2-approximation algorithm to solve the WDHEMC problem. We also present a combinatorial algorithm with an approximation bound 3 for the WDHEMC problem, and show that this bound is tight. By utilizing the standard technique in [1, 8, 11, 12], we obtain a PTAS to solve the WDHEMC problem for the version where each directed hyperedge exactly consists of two nodes.

This paper is organized as follows: we first give some definitions in Section 2. Then we present two approximation algorithms in Section 3 for the WDHEMC problem. A PTAS for the special version where each directed hyperedge contains only two nodes is given in Section 4. We conclude our work with some remarks and discussions about future research directions in the last section.

# 2 Preliminaries

A *mixed cycle* of $n$ nodes is a mixed graph $C = (V,E,A)$, where $V = \{1,2,\dots,n\}$ is the set of $n$ vertices on the mixed cycle $C$, and $E$ ($A$, resp.) is the set of undirected (directed, resp.) links. For each $i = 1,2,\ldots,n$, either undirected link $e_i = (i,i+1)$ belongs to $E$, or both directed links $e_i^+ = (i,i+1)$ and $e_i^- = (i+1,i)$ belong to $A$, where we treat the node $n+i$ as the node $i$ for $1 \leq i \leq n$. For the version where $A = \emptyset$, the mixed ring becomes a undirected cycle, and for the version where $E = \emptyset$, the mixed ring becomes a bidirected cycle. As in [10], a directed hyperedge $h = (u,S)$ is a pair, where $u \in V$ is indicated as the *source* of the directed hyperedge $h$ and $S \subseteq V - u$ is the set of *sinks*. In communication applications, each directed hyperedge $h$ represents a request asked to send a message from $u$ to every vertex in $S$. Let $H = (V, E_H; w)$ denote a directed hypergraph with the same set of vertices $V$ and a set of $m$ directed hyperedges $E_H = \{h_1, h_2, \ldots, h_m\}$.

For each directed hyperedge $h_j = (u_j, S_j) \in E_H$, where $S_j = \{i_1^j, i_2^j, \ldots, i_{k_j}^j\}$ and $k_j = \mid S_j \mid$, let $w_j$ denote the weight of $h_j$, and we sometimes treat such weight as the *load* of $h_j$ on the mixed cycle $C$. For convenience, we use $i_0^j$ to denote $u_j$. Assume that the $k_j+1$ vertices $i_0^j, i_1^j, \ldots, i_{k_j}^j$ are sorted in clockwise order on the cycle $C$ and $i_0^j$ is the source of the directed hyperedge $h_j$. For each $k = 0,1,\ldots,k_j-1$, let $P_k^j$ be the embedding of directed hyperedge $h_j$ which can be obtained by cutting the segment of vertices on the cycle from vertex $i_k^j$ to vertex $i_{k+1}^j$. It is easy to verify that the embedding $P_k^j$ consists of two directed path: one is from $i_0^j$ to $i_k^j$ in the clockwise direction, and the other is from $i_0^j$ to $i_{k+1}^j$ in the counterclockwise direction. The embedding $P_{k_j}^j$ is obtained by cutting the segment of vertices on the cycle from vertex $i_{k_j}^j$ to vertex $i_0^j$, *i.e.*, $P_{k_j}^j$ consists of one directed path which is from $i_0^j$ to $i_{k_j}^j$ in the clockwise direction. For convenience, we also treat $P_k^j$ as the set of links used in the embedding.

For two directed links $e_i^+ = (i,i+1)$ and $e_i^- = (i+1,i)$, define $C_i^+ = \cup_{j=1}^m C_{ij}^+$ and $C_i^- = \cup_{j=1}^m C_{ij}^-$, where $C_{ij}^+ = \{P_k^j \mid e_i^+ \in P_k^j,\ k = 0,1,2,\ldots,k_j\}$ and $C_{ij}^- = \{P_k^j \mid e_i^- \in P_k^j,\ k = 0,1,2,\ldots,k_j\}$, and then we set $C_i = C_i^+ \cup C_i^-$. For each embedding $P_k^j$ of the directed hyperedge $h_j$, we introduce a binary variable $x_k^j$ ($k = 0,1,\ldots,k_j$) to represent such an embedding $P_k^j$, where $x_k^j = 1$ if we choose $P_k^j$ as the embedding of the directed hyperedge $h_j$ and $x_k^j = 0$ otherwise. Given an embedding $x$ of all the directed hyperedges in $E_H$, the *congestion* of the link $e_i$ ($e_i^+$ or $e_i^-$) is the overall weight of the directed hyperedges whose embedding contain such a link.

Now, we construct an integer linear program to present the WDHEMC problem as follows (for convenience, we denote this programming as ILP):

$$
\begin{array}{rcll}
 & \min & B & \\
c(e_i^+) = \sum_{P_k^j \in C_i^+} w_j x_k^j & \leq & B; & e_i^+ = (i, i+1) \in A \\
c(e_i^-) = \sum_{P_k^j \in C_i^-} w_j x_k^j & \leq & B; & e_i^- = (i+1, i) \in A \\
c(e_i) = \sum_{P_k^j \in C_i^+} w_j x_k^j + \sum_{P_k^j \in C_i^-} w_j x_k^j & \leq & B; & e_i = (i, i+1) \in E \\
\sum_{k=0}^{k_j} x_k^j & = & 1; & j = 1, 2, \ldots, m \\
x_k^j & \in & \{0, 1\}; & j = 1, 2, \ldots, m; k = 0, 1, \ldots, k_j
\end{array}
$$

It is easy to find that WDHEMC problem is NP-hard by reduction from the PARTITION problem as in [9]. When we relax the preceding ILP as the relaxed LP and require only that $0 \leq x_k^j \leq 1$ for all integers $j$ and $k$, it is easy to find that the LP-based rounding algorithm in [5] can not be extended here to produce a 2-approximate solution. Hence, we need some new ideas. The nontrivial LP-based rounding algorithm is presented in the following section.

# 3 Two Approximation Algorithms for the WDHEMC problem

For any feasible solution $x$ to the relaxation of ILP, we define $x(C_{ij}^+) = \sum_{P_k^j \in C_{ij}^+} x_k^j$ and $x(C_{ij}^-) = \sum_{P_k^j \in C_{ij}^-} x_k^j$. For each $j = 1, 2, \ldots, m$, it is easy to verify that $w_j x(C_{ij}^+)$ and $w_j x(C_{ij}^-)$ are the contributions of directed hyperedge $h_j$ to the directed link $e_i^+$ and to the directed link $e_i^-$, respectively.

**Lemma 1.** Given a feasible solution $x$, for each directed hyperedge $h_j$, we have $x(C^+_{(i_t^j-1)j}) + x(C^-_{i_t^j j}) = 1$ for each $t = 1, 2, \ldots, k_j$.

**Proof.** For each directed hyperedge $h_j$, as every embedding $P_k^j$ must contain the vertex $i_t^j$, we have the fact $P_k^j$ is either in $C^+_{(i_t^j-1)j}$ or in $C^-_{i_t^j j}$. Combining the fact that $x$ is a feasible solution subject to $\sum_{k=0}^{k_j} x_k^j = 1$, the lemma holds. □

**Lemma 2.** For each directed hyperedge $h_j$, we have $x(C^+_{i_0^j j}) \geq x(C^+_{(i_0^j+1)j}) \geq \cdots \geq x(C^+_{(i_1^j-1)j}) \geq x(C^+_{i_1^j j}) \geq \cdots \geq x(C^+_{(i_{k_j}^j-1)j}) \geq x(C^+_{i_{k_j}^j j}) = \cdots = x(C^+_{(i_0^j-1)j}) = 0$.

**Proof.** According to the definitions of $P_k^j$ and $C_{ij}^+$, we have $P_k^j \in C_{ij}^+$ $(i_0^j \leq i < i_k^j)$ and $P_k^j \notin C_{ij}^+$ $(i_k^j \leq i < i_0^j)$, which implies that $C^+_{i_0^j j} \supseteq C^+_{(i_0^j+1)j} \supseteq \cdots \supseteq C^+_{(i_1^j-1)j} \supseteq C^+_{i_1^j j} \supseteq \cdots \supseteq C^+_{(i_{k_j}^j-1)j} \supseteq C^+_{i_{k_j}^j j} = \cdots = C^+_{(i_0^j-1)j} = \varnothing$. Hence, the lemma holds. □

**Corollary 1.** For each directed hyperedge $h_j$, we have $0 = x(C^-_{i^j_0 j}) = x(C^-_{(i^j_0+1)j}) = \cdots = x(C^-_{(i^j_1-1)j}) \le x(C^-_{i^j_1 j}) \le \cdots \le x(C^-_{(i^j_{k_j}-1)j}) \le x(C^-_{i^j_{k_j} j}) \le \cdots \le x(C^-_{(i^j_0-1)j})$. □

Let $\widetilde{x}$ denote the fraction optimal solution to the relaxed LP. Our LP-based algorithm is constructed as follows:

***LP-based*** *Algorithm*

Step 1 Solve the relaxed LP, and then get an optimal fraction solution $\widetilde{x}$.

Step 2 For each directed hyperedge $h_j$, find smallest $k$ such that $\widetilde{x}(C^+_{i^j_k j}) \le 0.5$, and set $\overline{x}^j_k = 1$ and $\overline{x}^j_{k'} = 0$ $(k' \ne k)$.

**Theorem 1.** The maximum congestion of the feasible solution $\overline{x}$ produced by the LP-based algorithm is at most $2{\cdot}OPT$, where $OPT$ denotes the value of the optimal solution.

**Proof.** We rewrite the representation of $c(e^+_i)$ and $c(e^-_i)$ as $c(e^+_i) = \sum_{k=1}^m w_j x(C^+_{ij})$ and $c(e^-_i) = \sum_{k=1}^m w_j x(C^-_{ij})$. For each directed hyperedge $h_j$, we assume that we get $\overline{x}^j_k = 1$ after running the LP-based algorithm. We distinguish the following three cases.

*Case 1.* $i^j_0 \le i < i^j_k$. According to the choice of LP-based algorithm, we have $\widetilde{x}(C^+_{(i^j_k-1)j}) \ge 0.5$. By Lemma 2, we have $\widetilde{x}(C^+_{ij}) \ge 0.5$, which implies $\overline{x}(C^+_{ij}) = 1 \le 2\widetilde{x}(C^+_{ij})$. It is easy to verify that $\overline{x}(C^-_{ij}) = 0 \le 2\widetilde{x}(C^-_{ij})$.

*Case 2.* $i^j_k \le i < i^j_{k+1}$. We have $\overline{x}(C^+_{ij}) = 0 \le 2\widetilde{x}(C^+_{ij})$ and $\overline{x}(C^-_{ij}) = 0 \le 2\widetilde{x}(C^-_{ij})$.

*Case 3.* $i^j_{k+1} \le i < i^j_0$. According to the choice of LP-based algorithm and Lemma 2, we have $\widetilde{x}(C^+_{(i^j_{k+1}-1)j}) \le 0.5$. Combining Lemma 1 and Corollary 1, we have $\widetilde{x}(C^-_{ij}) \ge 0.5$. Hence, $\overline{x}(C^-_{ij}) = 1 \le 2\widetilde{x}(C^-_{ij})$. It is easy to verify that $\overline{x}(C^+_{ij}) = 0 \le 2\widetilde{x}(C^+_{ij})$.

By the fact that the value of the optimal solution $\widetilde{x}$ to the relaxed LP provides a lower bound on $OPT$, for each $e^+_i \in A$, we have

$$\overline{c}(e^+_i) = \sum_{k=1}^m w_j \overline{x}(C^+_{ij}) \le 2\sum_{k=1}^m w_j \widetilde{x}(C^+_{ij}) \le 2OPT;$$

and for each $e^-_i \in A$, we have

$$\overline{c}(e^-_i) = \sum_{k=1}^m w_j \overline{x}(C^-_{ij}) \le 2\sum_{k=1}^m w_j \widetilde{x}(C^-_{ij}) \le 2OPT;$$

and for each $e_i \in E$, we have

$$\overline{c}(e_i) = \overline{c}(e^+_i) + \overline{c}(e^-_i) = \sum_{k=1}^m w_j \overline{x}(C^+_{ij}) + \sum_{k=1}^m w_j \overline{x}(C^-_{ij}) \le 2(\sum_{k=1}^m w_j \widetilde{x}(C^+_{ij}) + \sum_{k=1}^m w_j \widetilde{x}(C^-_{ij})) \le 2OPT.$$

This establishes the conclusion of the theorem. □

In order to improve the efficiency, by utilizing the strategy similar to that in [9], we can embed each directed hyperedge $h_j$ in the shortest way which means using links in the mixed cycle as few as possible, we call it **shortest embedding** algorithm. We obtain the following results.

**Theorem 2.** For any instance of the WDHEMC problem, the shortest embedding algorithm can produce an embedding of directed hyperedges on a cycle with the congestion at most 3 times the optimum value. The running time is $O(nm)$, and the bound 3 is tight. □

**Corollary 2.** For the case where $E = \emptyset$, the shortest embedding algorithm can produce an embedding of directed hyperedges on a cycle with the congestion at most 2 times the optimum value. The running time is $O(nm)$, and the bound 2 is tight. □

# 4 A PTAS for the Special Version

In this section, we study the problem for the version $k_j = 1$ for all $j = 1, 2, \ldots, m$, and we denote this version as the mixed ring loading problem. When $A = \emptyset$, this problem is deeply studied in [8, 12] and there is a PTAS to solve it completely. Based on the same ideas in [8, 12], Becchetti *et al.* [1] designed a PTAS to solve the problem for the version $E = \emptyset$. We extend the method to our mixed ring and present a PTAS to solve the WDHEMC problem for the version $k_j = 1$ for all $j = 1, 2, \ldots, m$.

By solving the LP-based algorithm in Section 3, we can get a feasible solution with objective value $\widehat{B} \leq 2OPT$. Note that each hyperedge has two routing ways. For any fixed positive number $\varepsilon < 1$, following from the ideas in [1, 8, 12], a directed hyperedge is routed *long-way* if it uses the longer path to connect its end-vertices (ties are broken arbitrarily), and it is routed *short-way* otherwise. For convenience, a directed hyperedge $h_j$ is called *heavy* if $w_j \geq \varepsilon\widehat{B}/5$ and otherwise as *light*.

**Lemma 3.** In any optimal solution, there are at most $20/\varepsilon$ heavy, long-way routed directed hyperedges. □

Let $H \subseteq E_R$ and $L = E_R - H$ denote the set of heavy directed hyperedges and the set of light directed hyperedges, respectively. For each subset $S \subseteq H$, we route the hyperedges in $S$ in long way. Let $s(e_i^+)$ ($s(e_i^-)$, resp.) denote the congestion of $e_i^+$ ($e_i^-$,resp.) resulting from routing the directed hyperedges in $S$ in the long-way and the directed hyperedges in $H - S$ in the short-way. In this special case, for each directed hyperedge $h_j$, there are only two indicator variables: $x_1^j$ (clockwise) and $x_0^j$ (counterclockwise).

For any set $S$, we get the following integer linear program (denote it as $LP_S$):

$$\begin{array}{rcll}
 & \min & B & \\
c(e_i^+) = s(e_i^+) + \displaystyle\sum_{x_1^j \in C_i^+, h_j \in L} w_j x_1^j & \leq & B; & e_i^+ = (i, i+1) \in A \\
c(e_i^-) = s(e_i^-) + \displaystyle\sum_{x_0^j \in C_i^-, h_j \in L} w_j x_0^j & \leq & B; & e_i^- = (i+1, i) \in A \\
c(e_i) = c(e_i^+) + c(e_i^-) & \leq & B; & e_i = (i, i+1) \in E \\
x_0^j + x_1^j & = & 1; & h_j \in L \\
x_0^j, x_1^j & \in & \{0, 1\}; & h_j \in L
\end{array}$$

Finally, let $\tilde{x}$ denote the optimal fractional solution of $LP_S$. By using the original notations similar to [12], we call that two directed hyperedges $h_j = (u_j, \{s_j\})$ and $h_k =$

$(u_k, \{s_k\})$ are *parallel* if the intervals $[u_j, s_j]$ and $[s_k, u_k]$ or the intervals $[s_j, u_j]$ and $[u_k, s_k]$ intersect at most at their endpoints. We call a directed hyperedge $h_j$ parallel to $e_i^+$, if $P_1^j \in C_{ij}^+$; otherwise it is parallel to $e_i^-$. For any optimal solution $\widetilde{x}$ to $LP_S$, a directed hyperedge $h_j$ is called *split* if $0 < \widetilde{x}_1^j < 1$.

**Lemma 4.** There exists an optimal solution $\widetilde{x}$ such that no pairs of parallel hyperedges are both split. □

As the strategy in [12], without loss of generality, we assume that the split directed hyperedges are numbered as $L_S = \{h_1, h_2, \ldots, h_q\}$. Since no split directed hyperedges are parallel, we may order them clockwise simultaneously by source $u_j$ and by sink $s_j$. For any link $e_i^+$, there is a interval $[l_i, r_i] \subseteq \{1, 2, \ldots, q\}$, interpreted if necessary "around the corner" modulo $q$, which contains exactly the indices of the hyperedges in $L_S$ which are parallel to the link $e_i^+$. For the link $e_i^-$,the indices are $[r_i + 1, l_i - 1]$.

Using the rounding technique as in [12], we give an integer solution $x'$ recursively by putting:

$$x_1'^j = \begin{cases} 1 & \text{if } -w_j\widetilde{x}_1^j + \sum_{k=1}^{j-1} w_k(x_1'^j - \widetilde{x}_1^j) < -\frac{w_j}{2} \\ 0 & \text{otherwise} \end{cases}$$

Correspondingly, we set $x_0'^j = 1 - x'_{j1}$ and denote $W = \varepsilon\widehat{B}/5$. By induction, we can conclude that any partial sum $\sum_{k=1}^{j} w_k(x_1'^j - \widetilde{x}_1^j)$ lies in the half-open real interval $[-\frac{W}{2}, \frac{W}{2})$.

As in [1, 12], we can prove the following theorem.

**Theorem 3.** For any link $e_i$ ($e_i^-$ or $e_i^+$), we have $c'(e_i) \leq \widetilde{c}(e_i) + \varepsilon OPT$. □

We now construct our PTAS to solve the mixed ring loading problem as follows:

*Algorithm* ***Mixed Ring Loading***

*Step 1.* Route the hyperedges in each subset $S$ in the long way and the hyperedges in $H - S$ in the short way respectively.

*Step 2.* Solve $LP_S$ corresponding to $S$, and obtain an optimal fraction solution $\widetilde{x}_S$.

*Step 3.* Among the fraction solutions, choose the solution $\widetilde{x}$ with the minimum congestion, then convert it to an integer solution $x'$ by utilizing the previous rounding technique.

**Theorem 4.** The Mixed Ring Loading algorithm produces a feasible solution $x'$, whose value is at most $(1+\varepsilon)OPT$, where $OPT$ is the optimal value, and the time complexity is polynomial, for any fixed $\varepsilon > 0$. □

# 5 Conclusion

In this paper, we study the problem of embedding a directed hypergraph in a mixed ring. The objective is to minimize the maximum congestion of the links in the mixed ring. We derive a 2-approximation algorithm based on linear programming, a 3-approximation combinatorial algorithm in $O(mn)$ time, and a PTAS for the special case where each directed hyperedge contains only two nodes.

Since the undirected version of weighted hypergraph embedding only has a $1.5+\varepsilon$-approximation algorithm [7], can we design a better approximation algorithm with approximation ratio less than 2 to the weighted directed version? Can we find a polynomial algorithm for the unweighted version of directed hypergraph embedding problem? These problems are interesting to be discussed in further study.

# References


[1] L. Becchetti, M.D. Ianni, and A.M. Spaccamela. Approximation algorithms for routing and call scheduling in all-optical chains and rings. *Theoretical Computer Science*, 287(2), 429-448, 2002.

[2] X. Deng and G. Li. A PTAS for embedding hypergraph in a cycle (extended abstract). *Proceedings of the 31st International Colloquium on Automata, Languages and Programming (ICALP)*, 433-444, 2004.

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

[4] J.L. Ganley and J.P. Cohoon. Minimum-congestion hypergraph embedding in a cycle. *IEEE Transactions on Computers*, 46(5), 600-602, 1997.

[5] T. Gonzalez. Improved approximation algorithm for embedding hyperedges in a cycle. *Information Processing Letters*, 67(5), 267-271, 1998.

[6] Q.P. Gu and Y. Wang. Efficient algorithms for minimum congestion hypergraph embedding in a cycle. *IEEE Transactions on Parallel and Distributed Systems*, 17(3), 205-214, 2006.

[7] H. Ho and S. Lee. Improved approximation algorithms for weighted hypergraph embedding in a cycle. *SIAM Journal on Optimization*, 18(4), 1490-1500, 2007.

[8] S. Khanna. A polynomial time approximation scheme for SONET ring loading problem. *Bell Labs Technical Journal*, 2(2), 36-41, 1997.

[9] S.L. Lee and H.-J. Ho. On minimizing the maximum congestion for weighted hypergraph embedding in a cycle. *Information Processing Letters*, 87, 271-275, 2003.

[10] K. Li and L. Wang. A polynomial time approximation scheme for embedding a directed hypergraph on a ring. *Information Processing Letters*, 97, 203-207, 2006.

[11] G. Wilfong and P. Winkler. Ring routing and wavelength translation. *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 333-341, 1998.

[12] A. Schrijver, P. Seymour, and P. Winkler, The ring loading problem. *SIAM Review*, 41(4), 777-791, 1999.