Journal of The Korea Society of Computer and Information
Vol. 21 No. 7, pp. 47-52, July 2016
www.ksci.re.kr
http://dx.doi.org/10.9708/jksci.2016.21.7.047

An Eulerian Cycle Algorithm for Chinese Postman Problem

Sang-Un Lee*

Abstract

This paper introduces an algorithm to construct an Eulerian cycle for Chinese postman problem. The Eulerian cycle is formed only when all vertices in the graph have an even degree. Among available algorithms to the Eulerian cycle problem, Edmonds-Johnson's stands out as the most efficient of its kind. This algorithm constructs a complete graph composed of shortest path between odd-degree vertices and derives the Eulerian cycle through minimum-weight complete matching method, thus running in $O(|V|^3)$. On the contrary, the algorithm proposed in this paper selects minimum weight edge from edges incidental to each vertex and derives the minimum spanning tree (MST) so as to finally obtain the shortest-path edge of odd-degree vertices. The algorithm not only runs in simple linear time complexity $O(|V|\log|V|)$ but also obtains the optimal Eulerian cycle, as the implementation results on 4 different graphs concur.

▸ Keywords : Chinese postman problem, Eulerian cycle, Degree, Minimum Weighted edge

• First Author: Sang-Un Lee , Corresponding Author: Sang-Un Lee
*Sang-Un Lee (sulee@gwnu.ac.kr), Dept. of Multimedia Engineering, Gangneung-Wonju National University
• Received: 2016. 03. 20, Revised: 2016. 04. 11, Accepted: 2016. 06. 07.

I. Introduction

본 논문은 주어진 연결된 무방향 그래프 $G=(V,E)$ 에 대해 특정 정점(vertex, $v \in V$)에서 출발하여 모든 간선(edge, $e=\{u,v\}$) 들을 한 번씩만 방문하고 출발 정점으로 되돌아오는 중국인 우체부 문제(chinese postman problem, CPP)를 연구 대상으로 한다[1-2].

CPP 문제의 해를 구하기 위해서는 모든 정점들의 차수(degree, $d_G(v)$)가 짝수가 되도록 홀수 차수(odd degree)를 가진 정점들 간에 최소 가중치 간선을 추가하여 짝수 차수(even degree)정점으로 만들어야 한다. 이를 오일러 사이클(Eulerian cycle, tour 또는 circuit)이라 한다[3,4]. 이들 문제는 특히, 쓰레기 수집, 도로 청소, 학교 버스 운행, 버스 안내전화 서비스, 장애자 수송, 판매원 여행 경로 또는 유지보수 분야 등 많은 수송 분야에 적용되고 있다[5-6].

주어진 $G=(V,E)$ 에 대해 모든 정점의 차수가 짝수개가 되도록 최소의 오일러 사이클을 만드는 대표적인 방법으로 Edmonds-Johnson 알고리즘(EJA)[7,8]이 제안되어 있다. 그러나 EJA는 홀수 차수를 가진 정점들 간의 최단 거리(shortest path)를 계산하고, 최소 가중치 완전 매칭(minimum-weight perfect matching, MWPM)을 수행하여 모든 정점들이 짝수 차수를 가지도록 만든다. 이 알고리즘의 수행 복잡도는 $O(|V|^3)$으로 복잡함과 더불어 MWPM을 수행하는 방법이 어렵다[8].

한 정점에서 시작하여 모든 정점을 한 번씩만 방문하고 출발 정점으로 되돌아오는 외판원문제(traveling salesperson problem, TSP)에 대해서는 최근 들어 연구가 수행되고 있다[9,10]. 그러나 모든 간선을 최단 길이로 한 번씩만 방문하는 오일러 사이클에 대한 연구는 더 이상 진행되지 않고 있다.

본 논문은 주어진 연결된 무방향 그래프에 대해 선형시간 복잡도 $O(|V|)$로 모든 정점들이 짝수개의 차수를 가지는 최소 길이의 오일러 사이클을 구성하는 알고리즘을 제안한다. 2장에서는 EJA를 데이터에 적용하여 문제점을 고찰해 본다. 3장에서는 선형시간 복잡도 $O(|V|\log|V|)$로 오일러 사이클을 단순히 구하는 알고리즘을 제안한다. 4장에서는 실제 문제에 적용하여 제안된 알고리즘의 적용성을 평가해 본다.

II. Related Works and Problems

오일러 사이클을 얻으려면 모든 정점의 차수가 짝수이어야만 한다. 만약, 주어진 그래프 $G=(V,E)$ 에서 홀수 차수를 가진 정점들이 존재한다면 이 정점들 간에 간선을 추가하여 짝수 차수를 갖도록 해야만 한다. CPP 역시 오일러 사이클을 찾는 문제이다. 따라서 CPP에 대한 알고리즘들의 차별성은 '홀수 차수 정점들 간에 어떤 방법으로 간선을 추가하여 모든 정점을 짝수 차수로 만들 수 있는가?' 로 결정된다.

주어진 연결된 무방향 그래프 $G=(V,E)$, 간선 가중치 $w: E \rightarrow R_0^+$ 에 대해 1명의 우체부가 정점 v_1에서 출발하여 모든 간선들을 정확히 한 번씩만 방문하고 다시 v_1으로 되돌아오는 CPP의 최적 1-우체부 순회(optimal 1-postman tour) C^* 를 구하는 EJA[7,8]는 그림 1에 제시되어 있다.

EJA는 모든 정점의 차수를 짝수개로 만들어 오일러 사이클을 찾는 방법으로 홀수 차수 정점들 간에 최단경로(shortest path, SP)를 갖는 간선을 추가하는 방식을 적용하였다. EJA는 홀수 차수 정점들 간 완전 그래프 G' 를 생성하고, 두 정점 v_i와 v_j 간의 최단경로 길이 $w(SP_G(\{v_i,v_j\}))$ 를 G로부터 계산한 후, G'에서 최소 가중치 완전 매칭 M을 계산하여 매칭된 간선 $w'\{v_i,v_j\}$ 선택하여 G에 추가한 $\tilde{G}$ 그래프를 얻는다. 여기서 완전매칭을 수행하는데 $O(|V|^3)$의 알고리즘 복잡도가 요구된다[8]. 마지막으로 $\tilde{G}$ 그래프에서 각 간선을 정확히 한 번씩만 방문하는 폐쇄경로인 오일러 사이클 C^* 를 얻는다.

입력 : 연결된 무방향 그래프 $G=(V,E)$, 간선 가중치 $w: E \rightarrow R_0^+$
출력: 최적 1-우체부 순회 C^*
Step 1. $G=(V,E)$ 에서 홀수 차수를 가진 m개의 정점들 간의 완전 그래프 G' 생성.
G' 그래프의 두 정점 v_i와 v_j 간의 간선 $w'\{v_i,v_j\}=w(SP_G(\{v_i,v_j\}))$ 을 G 그래프의 두 정점 v_i와 v_j 간의 최단경로 길이로 계산.
Step 2. G'에서 최소 가중치 완전 매칭 M을 계산하여 매칭된 간선 $w'\{v_i,v_j\}$ 선택. /* $O(|V|^3)$ */
Step 3. $SP_G(\{v_i,v_j\})$ 의 선택된 간선 $w'\{v_i,v_j\}$ 을 G에 추가하여 $\tilde{G}$ 그래프를 얻음.
Step 4. $\tilde{G}$ 그래프에서 각 간선을 정확히 한 번씩만 방문하는 폐쇄경로인 오일러 사이클 C^* 생성.

Fig. 1. Edmonds-Johnson algorithm(EJA)

Ahr[7]에서 인용된 그림 2의 G_1 그래프에 대해 EJA를 적용한 결과는 그림 3과 같다.

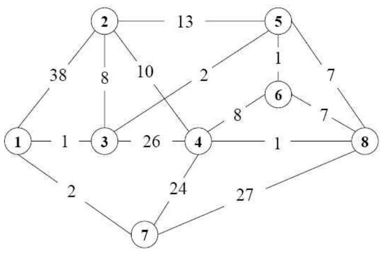


Fig. 2. G_1 graph

그림 2에서 8개의 각 정점은 $v_1,v_2,\cdots,v_8$ 로 명기되어 있고, 두 정점 v_i,v_j의 거리는 간선 가중치 $w\{v_i,v_j\}$ 로 주어졌다. G_1에서 홀수 차수 정점은 v_1,v_4,v_6,v_7 임을 알 수 있다. 따라서 그래프 G' 는 v_1,v_4,v_6,v_7 로 구성된다. 이들 4개 정점간의 최단거리 간선 가중치는 G로부터 얻으면 $w(SP(v_1,v_4))=11$, $w(SP(v_1,v_6))=4$, $w(SP(v_1,v_7))=2$, $w(SP(v_4,v_6))=8$, $w(SP(v_4,v_7))=13$, $w(SP(v_6,v_7))=6$ 을 계산할 수 있다. G' 그래프에 대해 최소 가중치 완전 매칭을 수행한 결과 $2\{v_1,v_7\}$ 과 $8\{v_4,v_6\}$ 을 얻었다. G_1 그래프에 $2\{v_1,v_7\}$ 과 $8\{v_4,v_6\}$ 간선을 추가하면 최

종적으로 모든 정점의 차수가 짝수인 $\tilde{G}$를 얻는다. 최종적으로 얻은 $\tilde{G}$ 그래프에 대해 임의로 간선을 선택하면서 모든 간선을 한 번씩만 방문하고 출발 정점으로 되돌아오는 오일러 사이클 C^* 를 얻을 수 있다.

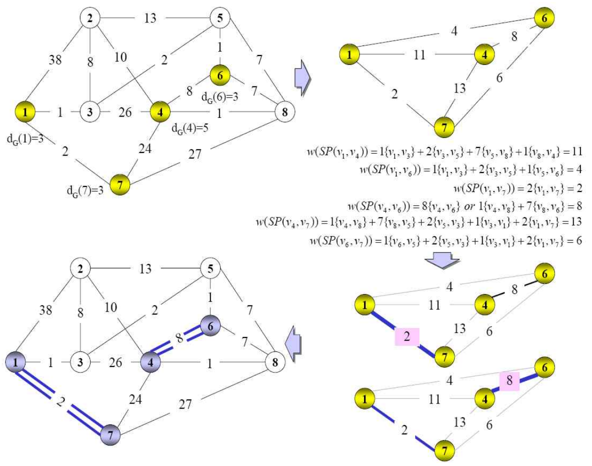


Fig. 3. Eulerian cycle of G_1 graph using EJA

그림 3에서 알 수 있듯이 EJA는 홀수 차수 정점들 쌍 간의 최단경로를 얻는데 복잡한 계산 과정을 수행해야 하며, 최소 가중치 완전 매칭을 수행하는데도 어려움이 있다. 따라서 이 알고리즘의 수행 복잡도는 $O(|V|^3)$으로 정점 개수가 100개이면 10^6 을 수행해야 한다[7,8].

3장에서는 EJA 수행 복잡도 $O(|V|^3)$을 $O(|V|\log|V|)$인 선형 복잡도로 쉽게 오일러 사이클을 얻는 알고리즘을 제안한다.

III. Minimum Weighted Edge Selection Eulerian Cycle Algorithm

EJA는 홀수 차수 정점들 간 최단경로를 계산하여 추가될 간선을 결정하는 방법을 제안하였다. 본 장에서 제안되는 알고리즘은 최소신장트리(minimum spanning tree, MST)를 쉽게 얻는 Borůvka 알고리즘(BA)[11,12]을 적용한다.

BA는 2단계를 수행한다. 첫 번째 단계(1st stage)에서 각 정점 v에 부속된(incident) 간선들 중 최소 가중치 간선(minimum weighted edge, MWE)를 중복에 무관하게 선택하여 연결한 부분 신장 트리(partial spanning tree, PST)를 얻는다. 만약 PST에서 사이클이 발생하면 최대 가중치 간선을 삭제한다. 2번째 단계(2nd stage)에서는 PST들 간의 MWE를 선택하여 최종적으로 MST를 얻는다. BA의 수행 복잡도는 $O(|V|\log|V|)$ 또는 $O(|E|\log|E|)$로 알려져 있다.

제안되는 알고리즘은 그림 4에 제시되어 있으며, 첫 번째로, BA의 1st Stage를 수행하여 PST를 얻는다. 만약, 단일 PST 내에 홀수 차수 정점이 짝수개가 존재하고 $w\{v_i,v_j\}$ 가 존재하면 선택한다. 만약, 홀수개가 존재하면 선택하지 않고 두 번째 단계를 수행한다. 두 번째 단계에서는 홀수 개를 포함한 PST의 홀수 차수 정점의 두 번째 MWE(2nd MWE)를 선택하여 PST들을 연결한다. 이와 같이 하여 홀수 차수 정점들 간에 연결이 되면 해당 간선을 선택한다. 제안된 알고리즘을 최소 가중치 간선 선택 알고리즘(minimum weighted edge selection algorithm, MWESA)라 한다.

입력: 연결된 무방향 그래프 $G=(V,E)$, 간선 가중치 $w: E \rightarrow R_0^+$
출력: 최적의 1-우체부 순회 C^*

Step 1. $G=(V,E)$에서 홀수 차수를 가진 m개 정점들 파악.
  /* 모든 정점에 부속된 간선들 중 최소 가중치 간선(MWE) 선택, 부분 신장트리(PST) 생성 */
  for $i=1$ to n
    choose MWE $_{\min} w\{i,j\} \in v_i$
  end
  connect selected $w\{i,j\} \in v_i$ into PST
  /* 단일 PST 내에서 짝수 개의 홀수 차수 정점이 존재하면 해당 정점 쌍의 간선 선택, 그렇지 않으면 Step 2 수행 */
  if ∃ odd degree vertices in even number in a PST
    then select the edge of it's pair of vertices
  else goto Step 2.
Step 2. /* 선택되지 않은 홀수 차수 정점의 2nd MWE 선택, PST들 연결 */
  Connect the PSTs with 2nd MWE of unselected odd degree vertices.
  /* 만약 단일 PST 내에서 선택되지 않은 홀수 차수 정점들 간 간선이 존재하면 해당 정점 쌍 간선 선택 */
  if ∃ odd degree vertices in a PST then
    select the edge of it's pair of vertices
Step 3. /* 선택된 간선 $w'\{v_i,v_j\}$ 을 G에 추가하여 $\tilde{G}$ 그래프를 얻음 */
  Get the graph $\tilde{G}$ from G + selected edge $w'\{v_i,v_j\}$
Step 4. /* $\tilde{G}$ 그래프에서 각 간선을 정확히 한 번씩만 방문하는 폐쇄경로인 오일러 사이클 C^* 생성 */
  Create closed path Eulerian cycle C^* that we visit every edge exactly once in graph $\tilde{G}$.

Fig. 4. Eulerian cycle algorithm using Minimum weighted edge selection

제안된 알고리즘을 G_1 그래프에 적용한 결과는 그림 5에 제시되어 있다.

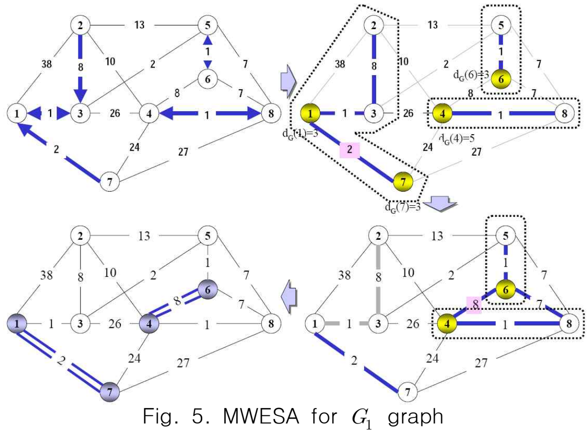


Fig. 5. MWESA for G_1 graph

먼저, Step 1에서 각 정점에 부속된 간선들 중 MWE를 선택한 결과 $P_1=\{v_2,v_3,v_1,v_7\}$, $P_2=\{v_5,v_6\}$, $P_3=\{v_4,v_8\}$ 의 3개 PST를 얻었다. 홀수 차수 정점 v_1,v_4,v_6,v_7 을 살펴보면

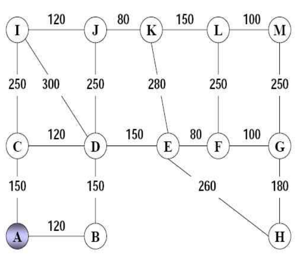
$v_1, v_7 \in P_1$ 이며, $\exists, 2\{v_1, v_7\} \in E$ 로 $2\{v_1, v_7\}$ 이 선택되어 $2\{v_1, v_7\}$ 은 2중간선으로 결정되었다. $v_6 \in P_2, v_4 \in P_3$ 로 Step 2에서 v_4 와 v_6 의 2^{nd} MWE인 $8\{v_4, v_6\}$ 과 $7\{v_6, v_7\}$ 을 선택하여 P_2 와 P_3 가 병합되었다. 병합된 PST에 홀수 차수 정점 v_4 와 v_6 쌍 간에 $8\{v_4, v_6\}$ 간선이 존재하여 선택되어 $8\{v_4, v_6\}$ 역시 이중 간선으로 결정되었다. 이 결과 모든 정점들의 차수가 짝수가 되어 오일러 사이클 C^* 를 쉽게 얻을 수 있다.

결국, 제안된 알고리즘은 BA를 적용하여 Step 1과 Step 2는 수행하는 관계로 수행 복잡도는 $O(|V|\log|V|)$이다.

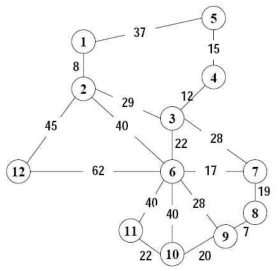
EJA는 m 개의 홀수 차수 정점들 간에 K_m-완전 그래프인 G' 를 생성하고 m 개 정점에 대한 두 정점 간 최단경로를 계산하여 최단경로인 완전 매칭 간선을 선택하여 G에 추가한 후 오일러 사이클 C^* 를 얻는 방법인데 반해, 제안된 MWESA는 m 개의 홀수 차수 정점들 간에 처음부터 각 정점의 간선들 중 최소가중치 간선(MWE)를 선택하여 연결한 PST를 생성하고, PSTSO의 최소차수 정점 쌍을 선택하고, 선택되지 않은 홀수 차수 정점은 2번째의 MWE를 선택하여 PST에 연결하는 단순한 방법을 적용하여 오일러 사이클 C^* 를 얻었다.

IV. Applications and Evaluation

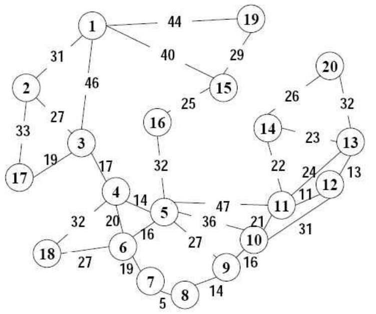
제안된 알고리즘의 적용성을 평가하기 위해 그림 6의 3개 그래프를 대상으로 실험을 수행한다.



(a) G_2 graph



(b) G_3 graph

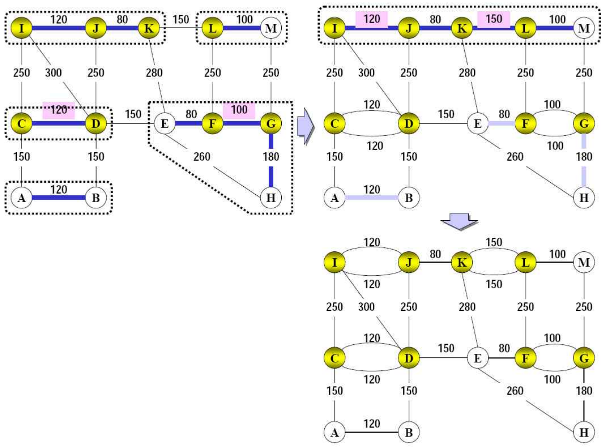


(c) G_4 graph

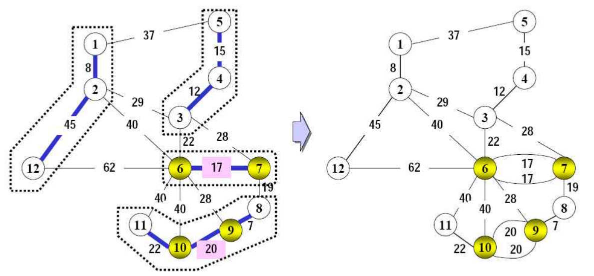
Fig. 6. Experimental graph

G_2 그래프는 Larson과 Odoni[1]와 Bricker[13]에서 인용되었다. 13개 지점으로 구성된 한 도시에서 우체국이 A에 존재하며, 19개 도로를 모두 방문하여 편지를 배달하는 문제이다. 이 문제에서 홀수 차수를 가진 정점은 C,D,F,G,I,J,K,L로 $m=8$ 이다. 이들 정점 쌍 간에 에 대해 짝수개의 차수를 갖도록 최소 길이의 중복된 간선을 $m/2=4$ 개를 추가해야 오일러 사이클이 생성된다. G_3, G_4 그래프는 Bricker[13]에서 인용되었다.

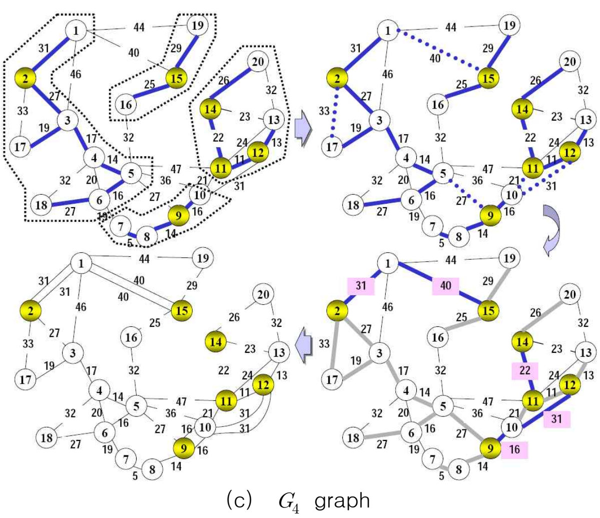
그림 6의 3개 그래프에 대해 제안된 최소 가중치 간선 선택 알고리즘을 적용한 결과는 그림 7에 제시되어 있다.



(a) G_2 graph



(b) G_3 graph



(c) G_4 graph

Fig. 7. MWESA for experimental graph

G_2 그래프는 각 정점의 MWE를 선택한 결과 PST는 5개가 구성되었다. 홀수 차수 정점 C,D,F,G,I,J,K,L 중에서 짝 수개의 홀수 차수 정점을 가진 PST에 대해 정점 쌍의 간선120{C,D}와

100{F,G}가 결정되었다. 나머지 홀수 차수 정점 I,J,K,L에 대해 남은 PST들 중 2nd MWE인 150{K,L}이 선택되어 120{I,J}와 150{K,L}이 결정되었다.

G_3 그래프는 각 정점의 MWE를 선택한 결과 PST는 4개가 구성되었다. 홀수 차수 정점 6,7,9,10 중에서 6,7과 9,10이 각각 다른 PST에 포함되어 있고 각 PST내의 정점 쌍 간에 간선이 존재하여 17{6,7}과 20{9,10}이 결정되었다.

G_4 그래프는 각 정점의 MWE를 선택한 결과 PST는 4개가 구성되었으나 각 PST가 짝수개의 홀수 차수 정점을 포함하지 않고 있다. 따라서 각 홀수 차수 정점의 2nd MWE를 선택하여 $SP((v_2,v_{15}))=16\{9,10\}+31\{10,12\}, SP((v_{11},v_{14}))=22[11,14\}$, $SP((v_9,v_{12}))=16\{9,10\}+31\{10,12\}$로 결정되었다.

EJA는 홀수 차수 정점들 간의 최단경로를 모두 구하고, 최소 가중치 완전 매칭 방법을 적용하여 복잡하게 모든 정점들의 차수가 짝수가 되도록 오일러 사이클을 결정하였다. 반면에, 제안된 알고리즘은 단순히 각 정점의 MWE를 선택하여 추가할 간선을 결정하였다. 만약, MWE 만으로 모든 홀수 차수 정점들을 짝수 차수 정점으로 만들지 못하면 2nd MWE를 다시 선택하여 추가할 간선을 결정하는데 성공하였다. 결국, 제안된 알고리즘은 EJA의 수행 복잡도 $O(|V|^3)$을 $O(|V|\log|V|)$로 단순화시키는 효과를 나타내었다.

V. Conclusions and Future Works

1명의 우체부가 모든 간선을 단지 한 번씩만 방문하고 출발지점으로 되돌아오는 CPP나 다수의 우체부가 수행하는 $k-$CPP($k\geq 2$) 문제에 있어 최우선적으로 수행하야 하는 것이 모든 정점의 차수를 짝수로 만드는 것이다. 이를 수행하기 위해 지금까지는 대부분 EJA를 적용하고 있지만 수행 복잡도가 $O(|V|^3)$로 많은 어려움이 있다. 또한, 이 방법은 최소 가중치 완전 매칭 방법을 적절히 적용해야 한다.

본 논문은 이러한 문제점을 해결하기 위해 각 정점에서의 MWE를 선택하여 수행 복잡도 $O(|V|\log|V|)$로 단순히 오일러 사이클을 결정하는 방법을 제안하였다. 본 논문에서 거론된 4개 그래프에 대해 제안된 알고리즘을 적용한 결과 모든 그래프에 대해 최소치를 가진 오일러 사이클을 쉽게 구하는데 성공하였다.

제안된 알고리즘을 적용하여 CPP나 $k-$CPP($k\geq 2$) 문제에 대해 쉽게 최적 해를 얻을 수 있을 것이다. 따라서 추후에는 $k-$CPP($k\geq 2$) 문제의 최적 해를 얻는데 본 제안된 알고리즘을 적용할 수 있는지를 검증할 예정이다.

REFERENCES

[1] R. C. Larson and A. R. Odoni," Urban Operations Research: Logistical and Transportation Planning Methods," Messachusetts Institute of Technology, Prentice-Hall, 1981.

[2] B. Kallehauge, J. Larsen, and O. B. G. Madsen, "Largrangean Duality Applied on Vehicle Routing with Time Windows: Experimental Results," Technical University of Denmark, 2001.

[3] Wikipedia, "Hamiltonian Path Problem," http://en.wikipedia.org/wiki/Hamiltonian_path_problem, Wikimedia Inc., 2016.

[4] Wikipedia, "Eulerian Path," http://en.wikipedia.org/wiki/Eulerian_path, Wikimedia Inc., 2016.

[5] G. Hasle, "Vehicle Routing and Traveling Salesperson Problems," Department of Optimization, SINTEF Applied Mathematics, Oslo, Norway, 2002.

[6] T. V. Hoai, "Vehicle Routing Problem: General Problem for TSP," Faculty of Computer Science and Engineering, Ho Chi Minh University of Technology, 2008.

[7] D. Ahr, "Contributions to Multiple Postmen Problems," Institute of Computer Science University of Heidelberg, 2004.

[8] A. Osterhues and F. Mariak, "On Variants of the k-Chinese Postman Problem," Universitat Dormund, Operations Research und Wirtshaftsinformatik, Vogelporthweg, Dormund, Germany, 2005.

[9] S. U. Lee, "The Extended k-opt Algorithm for Traveling Salesman Problem," Journal of KSCI, Vol. 17, No. 10, pp. 155-165, Oct. 2012.

[10] S. U. Lee, "A Polynomial Time Algorithm of a Traveling Salesman Problem," Journal of KSCI, Vol. 18, No. 12, pp. 75-82, Dec. 2013.

[11] O. Borůvka, "O Jistem Problemu Minimalnim," Prace Mor. Prrodved. Spol. V Brne (Acta Societ. Natur. Moravicae), Vol. III, No. 3, pp. 37-58, 1926.

[12] J. Nešetřil, E. Milková, and H. Nešetřilová, "Otakar Borůvka on Minimum Spanning Tree Problem (Translation of the both 1926 Papers, Comments, History)," DMATH: Discrete Mathematics, Vol. 233, 2001.

[13] D. Bricker, "The Chinese Postman Problem," Department of Industrial Engineering, University of Iowa, 2000. reference: H. A. Eiselt, M. Gendreau, and G. Laporte, "Arc Routing Problems, Part 1: The Chinese Postman Problem," Operations Research, Vol. 43, pp. 231-242, 1995.

Authors

Sang Un Lee received the B. Sc. degree in avionics from the Korea Aerospace University in 1997. He received the M. Sc. and Ph. D. degrees in Computer Science from Gyeongsang National University, Korea, in 1997 and 2001, respectively.
He is currently Professor with the Department of Multimedia Science, Gangneung-Wonju National University, Korea. He is interested in software quality assurance and reliability modeling, software engineering, software project management, neural networks, and algorithm.