Weight Constrained Travelling Salesman Problem on Halin Graphs

Dharmananda Gahir

University of Delhi, New Delhi-110026, India

Abstract: ***We prove that the Weight Constrained Travelling Salesman Problem is NP- Complete by polynomially transforming the 0-1 Knapsack Problem to it and vice-versa. We present a pseudo-polynomial time algorithm for computing a weight constrained minimum cost Hamilton cycle in a Halin graph and then present a fully polynomial time approximation scheme for this NP-hard problem.***

Keywords: Travelling Salesman Problem, Halin graph, NP-Complete, Approximation scheme, pseudo-polynomial time algorithm**.**

1. Introduction

Combinatorial optimization problems form a central object of study in operation research. Unfortunately, as for most of these problems the underlying decision problems are NP-complete, there is very little hope of finding feasible algorithm that produce optimal solutions (unless, P=NP). Research interest has therefore shifted from hunting after optimal solutions, to directions in which near optimal solutions were expected to be found. One approach for attacking the combinatorial optimization problems is using approximation algorithms that produce not optimal solutions, but solutions that are guaranteed to be a fixed percentage away from the actual optimum.

The classical travelling sales man problem associates an edge cost c ≥ 0 with each edge e ∊ E and asks for a minimum cost tour. We refer the readers to the books by Reinelt [8] and Lawer et.el. [11] for the details of TSP. It is well known that TSP is NP-hard Lawer et.el. [11]. Also researchers have identified various special cases which can be solved in polynomial time [1], [9], [10].

In this paper we consider the weight constrained travelling salesman problem (WCTSP) i.e. each edge has a weight besides cost, and we want to find a minimum cost hamilton tour with a total weight no more than a given weight constant W. We call this problem the *weight constrained travelling salesman problem* (WCTSP). We consider a special case of WCTSP, in which the underlying graph is a Halin graph. Cornuejols et.el. [1] have given $O(n)$ time algorithm for solving TSP on a Halin graph.

1.1 Halin graph

A Halin graph H_0 is obtained by embedding a tree T_0 having no nodes of degree 2 in the plane, and then adding a cycle C_0 to join the leaves of H_0 in such a way that the resulting graph is planar (see fig 1). We write $H_0 = T_0 \cup C_0$. Halin graphs are nontrivial generalizations of tree and ring networks, since a Halin graph is obtained by connecting the leaves of a tree by a cycle. These graphs are edge minimal 3-connected, and in general have a large number of Hamilton cycles. In fact, Halin graphs are 1-hamiltonian, i.e. they have Hamilton cycles and if any node is deleted, the resulting graph still has a Hamilton cycle.

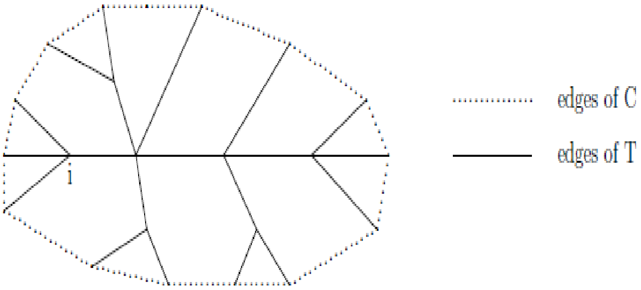

Figure 1: A Halin graph

Let $H_0 = T_0 \cup C_0$ be a Halin graph. If T_0 is a star i.e., a single node v connected to all other nodes, then H is called a wheel. We call v centre of the wheel. Now suppose H_0 has at least two nonleaves. Then the set of leaves of T adjacent to w, which we denote by $C(w)$, consecutive subsequence of cycle C. We call the subgraph of H induced by $\{w\} \cup C(w)$ a fan and w as the centre of the fan. Fig 2 is a fan with centre v.

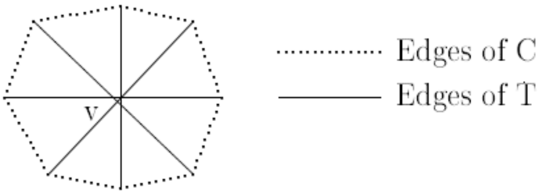

Figure 2: A wheel with centre v

1.2 WCTSP in Halin Graph

Given a Halin graph $H = (V, E)$ and an integer weight $M \geq 0$, we are interested in finding a minimum cost Hamilton cycle HC of H subject to the constraint that $w(HC) \leq$ M. We call such a tour a Weight Constrained Minimum Cost Hamilton cycle, and the corresponding problem Weight Constrained Travelling Salesman Problem (WCTSP).

Given $\mathrm{H} = (\mathrm{V,E})$ is a Halin graph, $\{c_{ij}, w_{ij}\}$ set of cost and weight respectively associated with edges ij Î E . Below we give a linear description of the WCTSP on a Halin graph.

min $\underset{ij\text{ Î }E}{\text{å}}$ $c_{ij}x_{ij}$ such that $\underset{ij\text{ Î }E}{\text{å}}$ $w_{ij}x_{ij}$ £ M and the constraints

$$x \,\text{Î}\, R^E : \begin{cases} x_{ij} \text{ £ } 1 \text{ " } ij \,\text{Î}\, E \\ \text{å}\,(x_{ij} : ij \,\text{Î}\, \text{d}(i)) = 2 \text{ " } i \,\text{Î}\, V \\ \text{å}\,(x_{ij} : ij \,\text{Î}\, \text{G}) = 2 \textit{ for every } 3\textit{-edge cut-set } \text{G} \textit{ of } H \end{cases}$$

where δ(i) is the set of edges incident on node i.

Theorem 1.1. The WCTSP in a Halin graph is NP-Complete.

Proof. We prove the theorem by showing that an instance of a 0-1 knapsack problem can be transformed to a WCTSP and vice-versa. Suppose we are given the instance of the 0-1 Knapsack Problem as:

Given integers $c_1, c_2, ..., c_n, w_1, w_2, ..., w_n ; K$

$\max \text{å}_{j=1}^{n} c_j x_j$

subject to $\text{å}_{j=1}^{n} w_j x_j \text{ £ } K \quad x_j = 0, 1 \quad \text{"}j = 1,2,..n$

We polynomially transform the 0-1 Knapsack Problem to WCTSP. We construct the corresponding Halin graph H = (V,E) as follows:

Corresponding to each element $i \in \{1, 2, ..., n\}$, construct the fan F_i with leaf nodes i_1, i_2, i_3 and v_i as its centre. Let v_0 be the node connecting all fans, i.e. $(v_0, v_i) \,\text{Î}\, E \text{ " } i \,\text{Î}\, \{1,2,...,n\}$.

Thus $V = \text{U}_{i=1}^{n} \{i_1, i_2, i_3, v_i\} \text{U} \{v_0\}$ and

$E = \{[i_1,i_2],[i_2,i_3],[v_i,i_1],[v_i,i_2],[v_i,i_3],[v_0,v_i]\}$

for $i = \{1,2,...,n\}$.

Define weights and costs for edges in E as follows:

cost $[i_1,i_2]$ = $-c_i$

weight $[i_1,i_2]$ = w_i

cost $[i_2,i_3]$ = L

weight $[i_2,i_3]$ = 0

All the remaining edges have cost zero and weight zero. R.H.S. number of the weight constraint is K, and L is a large number. This completes the construction of the Halin graph, Fig. 3 and thus describing the corresponding weight constrained problem on H.

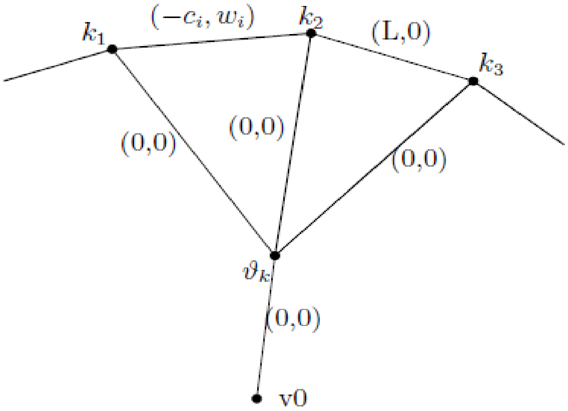

Figure 3: A WCTSP instance constructed from a 0 – 1 Knapsack problem

Suppose that the constructed instance of WCTSP has an optimal tour T^*. Let P = $\{j : [j_i, j_2] \,\text{Î}\, T^*\}$ and we write $T^* = A \text{U} B \text{U} C \text{U} D$ where

$$A = \text{U}_{j \text{Î} P} \{[j_1, j_2] \text{U} [j_2, v_j] \text{U} [v_j, j_3]\}$$

$$B = \text{U}_{j \text{Ï} P} \{[j_2, j_3] \text{U} [j_1, v_j] \text{U} [v_j, j_2]\} \quad \text{and}$$

$$C = \text{U} \; [n_2,n_3]\text{U}[n_3,v_n]\text{U}[v_n,v_0]\text{U}[v_0,v_1]\text{U}[v_1,1_1]\text{U}[1_2,1_3] \\ \text{U}[1_1,1_2]$$

$$D = \text{U}_{j=1}^{n-1} [j_3, (j+1)_1]$$

Cost of T^*,

$$C(T^*) = \text{å}_{j \text{Î} P} - c_j + (n + 2 - |P|)L \qquad (*)$$

Weight of T^*,

$$W(T^*) = \text{å}_{j \text{Î} P} w_j \text{ £ } K$$

Define the solution X = $\{x_j\}$ for the knapsack problem as:

$$x_j = \begin{cases} 1, & \text{for } j \,\text{Î}\, P \\ 0, & \text{for } j \,\text{Ï}\, P \end{cases} \qquad (**)$$

This solves the 0-1 knapsack problem with cost $\text{å}_{j \text{Î} P} c_j$ and weight $\text{å}_{j \text{Î} P} w_j \text{ £ } K$

Suppose X is not an optimal solution to the 0-1 knapsack problem, then we can get a contradiction. Let $X\text{¢} = \{x\text{¢}_j\}$ be an optimal solution to the knapsack problem with cost greater than $\text{å}_{j \text{Î} P} c_j$ and satisfying the weight constraint. $X\text{¢}$ will differ from X in at least two indices $r, s \,\text{Î}\, \{1,2,...n\}$ such that

$x\text{¢}_r = 0, x_r = 1 \text{ for } r \,\text{Î}\, P$

$x\text{¢}_s = 1, x_s = 0 \text{ for } s \,\text{Ï}\, P$

Cost of solution $X\text{¢} C(X\text{¢}) = \text{å}_{j \text{Î} P} c_j - c_r + c_s$ and weight

$W(X\text{¢}) = \text{å}_{j \text{Î} P} w_j - w_r + w_s$

By assumption,

$\sum_{j \in P} -c_j \leq \sum_{j \in P} -c_j - c_r + c_s$ (Since knapsack is maximization problem)

$\Rightarrow c_s > c_r$

Now the corresponding tour $T\text{¢} = T^* + \{[s_1,s_2] - [s_2,s_3]$

$+ [r_2,r_3] - [r_1,r_2]\}$ with cost of $T\text{¢}$given by

$$C(T\text{¢}) = \text{å}_{j \text{Î} P} - c_j + c_r - c_s + (n + 2 - |P|L)$$

$< \text{å}_{j \text{Î} P} - c_j + (n + 2 - |P|L) = C(T\text{¢})$ since $c_s > c_r$

This contradicts the optimality of T^*. Hence, if solution to Weight Constrained TSP is obtained in polynomial time, then we shall have solution to 0-1 knapsack problem, also in polynomial time. But 0-1 knapsack problem is NP-Complete. Thus, Weight Constrained *TSP* is also in NP-C.

2. A Pseudo Polynomial Time Algorithm for WCTSP

In the definition of the Weight Constrained Travelling Salesman Problem (WCTSP), the weight and cost of an edge are symmetric. We will present a pseudo polynomial time algorithm for computing a weight constrained minimum cost hamilton cycle and then apply standard technique of scaling and rounding to turn the pseudo polynomial time algorithm into a PTAS for WCTSP in the next section.

2.1 Fan Contractions

Let $G = H_0 = T_0$ U C_0 , when no fan contraction has taken place in the Halin graph H and let i be a non-leaf of T_0 which is adjacent to at most one other non-leaf j of T_0. If no such j exists, H_0 is a wheel. Let Γ_i denote the set of neighbours of i. The subgraph F_i of H_0 induced by Γ_i U $\{i\} - \{j\}$ is called a fan, and vertex i is called the centre of the fan F_i. It is easy to verify that every Halin graph has at least two fans. Our algorithm is *based on the idea of fan contraction.*

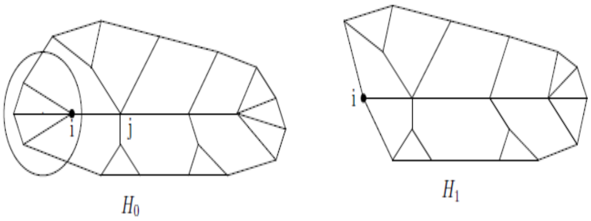

Figure 4: A Halin graph H_0 and its fan contraction H_1

A fan contraction of Halin graph $H_0 = T_0$ UC_0 along the fan F_i is as follows:

1) Attach the edges incident to F_i to the vertex i.
2) Delete vertices in F_i I C_0 from H_0

The resultant graph , denoted by H_1 (see figure 4), is clearly a Halin graph unless H_0 is a wheel. If H_0 is a wheel, let i be the centre and j be leaf node of the wheel. Then G_i U $\{i\}$- $\{j\}$ is a fan consisting of all nodes of the wheel except node j. By fan contraction, G_i U $\{i\}$- $\{j\}$ is contracted to node i and thus we get two vertices i and j joined by three parallel edges.

Let $H_k = T_k$ U C_k denote a Halin graph obtained from the original graph $H_0 = T_0$ U C_0 by k fan contractions. Let F_i denote a fan of H_k and assume that 1,2,...,r denote the vertices in F_i I C_k (in clockwise order) and node i its centre. Note that some of the vertices may represent contracted fans. For every s, $1 \leq s \leq r$, let s_f and s_g denote the first and last vertex (in clockwise order) on C_0, respectively, of the fan. which has been contracted into s in H_k. If $s \in C_0$, then $s_f = s = s_g$. Let G(s) denote the subgraph of H_0 induced by all the vertices that have been contracted into s. In case $s \in C_0$, $G(s) = s$.

For every s, $1 \leq s \leq r$ and for every W, W = 0, 1,..., M, consider the following subgraphs of G(s). We are in particular interested in the cost of the subgraphs $HPjl(s, W)$, $HPjk(s, W)$, $HPkl(s, W)$ of G(s), where $\{j, k, l\}$ is a 3-edge cut-set for the fan contracted into node 's' in H_0. Note: As 's' varies, edge j of C_0 which is incident on 1_f remains the same where as edge 'l' shifts from being incident on 1_g to r_g and 'k' is the edge connecting i and v_0. We define

$hpjl(s, W)$ be the minimum cost of hamilton path passing through 's' and containing the edges j and l with weight at most W.
$hpjk(s, W)$ be the minimum cost of hamilton path passing through 's' and containing the edges j and k with weight at most W.
$hpkl(s, W)$ be the minimum cost of hamilton path passing through 's' and containing the edges k and l with weight at most W.

2.2 Recurrence Rules

Let $G_i(s), 1£ s£ r$, denote a subgraph of H_0 induced by the vertex i together with the vertices of $G(1), G(2)$,..., $G(s)$. Define $HPjl_i(s, W)$, $HPjk_i(s, W)$, $HPkl_i(s, W)$, in $G_i(s)$ in the same manner as their counter-part in $G(s)$, all occurrence of s_f are replaced by 1_f. In fact, $G_{i(}r) = F_i = G(i)$.
Now we shall show how to determine $hpjl(i, W)$, $hpjk(i, W)$, $hpkl(i, W)$ in a Halin graph $H_k = T_k$ U C_k, provided $hpjl(1, W)$, $hpjk(1, W)$, $hpkl(1, W)$,..., $hpjl(r, W)$, $hpjk(r, W)$, $hpkl(r, W)$ are available by initialization or by previous computations.
First we initialize the variables for each vertex s on the cycle C_0 of the original Halin graph $H_0 = T_0$ U C_0, and for every W, W = 0, 1 ,..., M, as follows $hpjl(s, W) := hpjk(s, W) := hpkl(s, W) := 0$

Getting minimum hamiltonian path in a fan computing $hpjl(s, W)$
Let F_i be a fan with outer nodes 1, 2, ..., r(in clockwise order) on C with centre i. Since the hamilton path contains edge j and l we have r-1 choices(sub tours) in F_i through which hamilton path can traverse. Let these sub paths be denoted by $T_1, T_2, ..., T_{r-1}$ where T_s obtained by ejecting the edge e_s = (s-1, s) and adding edges (s-1, i) and (i, s). Let $w_1, w_2, ..., w_{r-1}$ denote the corresponding weights of these paths. Let t of them satisfy the weight constraint and let them be ordered as $T_1, T_2, ..., T_t$ with increasing cost. Then, $hpjl(s, W) = T_1$.

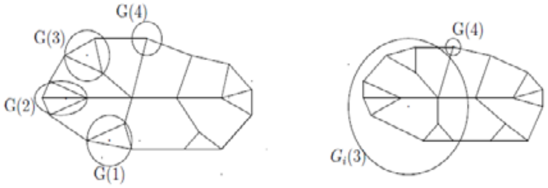

Figure 5: $G(s)$ and $G_i(s)$ in a Halin graph

The recurrence rule for $hpjl_i(1, W)$, $hpjk_i(1, W)$, $hpkl_i(1, W)$ are as follows:
$hpjl_i(1, W) = c(j) + c(l) + hpjl(1, W - w(j) - w(l))$

$hpjk_i(1, W) = c(j) + c(k) + hpjk(1, W - w(j) - w(k))$
$hpkl_i(1, W) = c(k) + c(l) + hpkl(1, W - w(k) - w(l))$.
For $2 \leq s \leq r$, let q = s-1. Suppose that $hpjl_i(q, W)$, $hpjk_i(q, W)$, $hpkl_i(q, W)$ and $hpjl(s, W)$, $hpjk(s, W)$, $hpkl(s, W)$ for every W, W = 0, 1, ..., M are given. We want to determine $pjl_{i(}s, W)$, $hpjk_i(s, W)$, $hpkl_i(s, W)$. $G_i(s)$ consists of $G_i(q)$, $G(s)$ and the edges(q_g, s_f) and(i, s). i.e. $\{j, k, l\}$ becomes cut set of $G(s)$ with $j = (q_g, s_f)$, $k = (i, s)$ and $l = (s_g, s + 1_f)$.

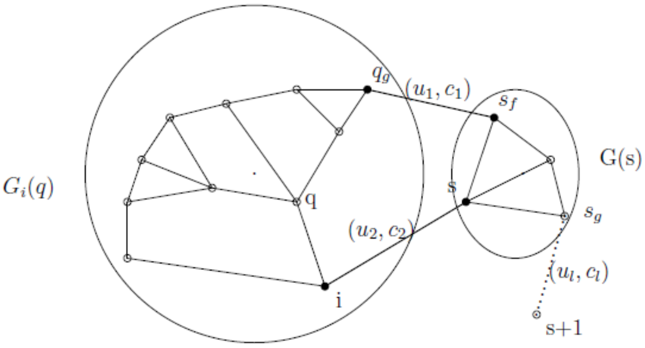

Figure 6: The configuration of $G_i(s)$

We determine $hpjl_i(s, W)$ in terms of $hpjl_i(q, W)$ and $hpjl(s, W)$. Let (u_1, c_1) and (u_2, c_2) be the weight and cost of edges (q_g, s_f) and (i, s) respectively. Let u_l and c_l be the weight and cost of edge $l = (s_g, s + 1_f)$. Let $HPjl_i(s, W)$ be the minimum cost hamilton path of $G_{i(}s)$ with weight at most W. i.e. hamilton path $HPjl_{i(}s, W)$ contains sub paths from $G_i(q)$ and $G(s)$. Thus one of the following three exhaustive cases applies to the sub-graph $HPjl_i(s, W)$.

Case 1: $HPjl_{i(}s, W)$ contains edge (q_g, s_f) but not (i, s) ;
Case 2: $HPjl_i(s, W)$ contains edge (i, s) but not (q_g, s_f);
Case 3: $HPjl_{i(}s, W)$ contains edges (q_g, s_f) and (i, s);

Here only case (i) is valid for $HPjl_i(s, W)$. Case (ii) and Case (iii) will occur in the last fan reduction, when the centre node of the Halin graph is to be accessed.

Case 1:
It means that $HPjl_i(s, W)$ contains edge (q_g, s_f) and vertices q_g ,s_f but not (i, s). Edge (q_g, s_f) connects two parts of $HPjl_i(s, W)$.
The first part is a sub path in $G_i(q)$ which contains the vertex q_g. It must be a $HPjl_{i(}q, W_1)$ for some $0 \leq W_1 \leq W$.
The second part is sub path in G(s), which contains the vertex s_f. It must be a $HPjl(s, W_2)$ for some $W_2 \geq 0$, where $W_1 + W_2 = W - u_1 - u_l$, $W_1 \geq 0$, $W_2 \geq 0$.
Thus $hpjl_i(s, W)$ is the minimum of the set:
$\{c_1 + c_l + hpjl_i(q, W_1) + hpjl(s, W_2) : W_1 + W_2 = W - u_1 - u_l \geq 0, W_1 \geq 0, W_2 \geq 0\}$
The recurrence rules for, $hpjk_i(s, W)$, $hpkl_i(s, W)$ can be obtained in a similar way.

2.3 Determination of Hamilton cycle

After contraction of F_i, i becomes a vertex on the cycle C_{k+1} of H_{k+1} and $G_i(r)$ is equal to $G(i)$ which form a node of C_{k+1}. Consequently for every W, W = 0, 1, ..., M, we get $hpjl(i, W) = hpjl_i(r, W)$, $hpjk(i, W) = hpjk_i(r, W)$, $hpkl(i, W) = hpkl_i(r, W)$ and this process can be repeated until the graph H_0 is reduced to a graph H_t consisting of two vertices with i and j joined by three parallel edges. Let $G(i)$, $G(j)$ be two subgraphs of H_0 corresponding to i and j in H_t respectively. Let w_1, w_2, w_3 and c_1, c_2, c_3 be weights and costs of three edges (i_f, j), (i, j), (i_l, j) respectively. For every W, W = 0, 1, ..., M, let $c(HC, W)$ be the minimum cost hamilton cycle with weight no more than W and the corresponding hamilton cycle is denoted by HC(W). The configuration of HC(W) must contains three cases.

Case 1: (i_f, j),(i, j)
Case 2: (i_f, j),(i_l, j)
Case 3: (i, j),(i_l, j)
Case 1: we get, c(HC,W) is
$c_1 + c_2 + hpjk_i(i, W - w_1 - w_2)$ with $W - w_1 - w_2 \geq 0$

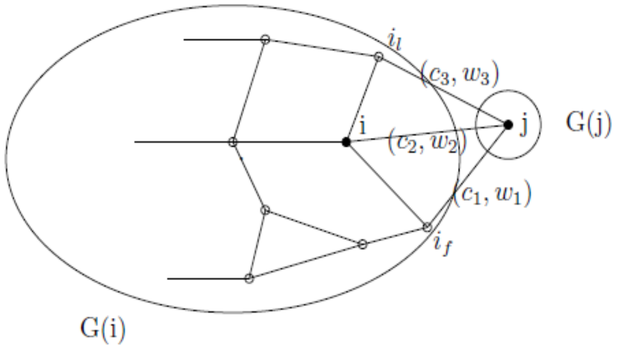

Figure 7: The configuration of $G(i)$ and $G(j)$

Case 2: we get, c(HC,W) is
$c_1 + c_3 + hpjl_i(i, W - w_1 - w_3)$ with $W - w_1 - w_3 \geq 0$
Case 3: we get, c(HC,W) is
$c_2 + c_3 + hplk_i(i, W - w_2 - w_3)$ with $W - w_2 - w_3 \geq 0$

Based on above we know that c(HC,W) is the minimum of the following numbers and negative weight indicates the nonexistence of a weight constrained hamilton cycle.

$c_1 + c_2 + hpjk_i(i, W - w_1 - w_2)$;
$c_1 + c_3 + hpjl_i(i, W - w_1 - w_3)$;
$c_2 + c_3 + hplk_i(i, W - w_2 - w_3)$;

2.4 Pseudo Polynomial Time Algorithm

We summarize the method discuss above in the following algorithm:

Algorithm 1 Pseudo Polynomial Time Algorithm for WCTSP.

Step 1: Let $H_0 = T_0$ U C_0, M be the bound for the weight. Set k := 0, F_i be a fan of H_k with centre i and 1, 2, ..., r be the vertices of F_i I C_k in clockwise order. Define the quantities $hpjl(s, W)$, $hpjk(s, W)$, $hpkl(s, W)$ as in the initialization step in section 2, for s := 1, 2, ..., r and W := 0, 1,..., M;

Step 2: Use the recurrence rules of section 2 to get $hpjl_i(s, W)$, $hpjk_i(s, W)$, $hpkl_i(s, W)$ for s := 1, 2, ..., r and W:= 0, 1, ..., M;

Step 3: Setting $hpjl(i, W) = hpjl_i(r, W)$, $hpjk(i,W) = hpjk_i(r, W)$, $hpkl(i, W) = hpkl_i(r, W)$ for W := 0, 1, ..., M when F_i is contracted into a vertex i and H_k is transformed into H_{k+1}. Set k := k + 1. If H_k consist of only two vertices with three parallel edges, goto step4; Otherwise, take a fan F_i of H_k and vertices 1, 2, ..., r of F_i I H_k in clockwise order and goto step2:

Step 4: Find the minimum cost hamilton cycle HC with a weight bound W, for every W = 0, 1, 2, ..., M.

Theorem 2.1. *Algorithm 1 correctly determines the hamilton cycle with a weight no more than a given bound M and a negative weight indicating the non-existence of Weight Constrained hamilton cycle. Further more the time complexity of algorithm1 is $O(n(M + 1)^2)$, where n is the number of vertices in the graph.*

Proof. In order to prove the recurrence rules we simply need to list all relevant cases. We prove the rule for $hpjl(i, M)$, and the others can be verified in similar a way.

$G_i(1)$ consists of $G(1)$, vertex i and edge $(i, 1)$. Then $HPjl_i(1, W)$ consists of edges j , l and $HPjl(1, W - w(j) - w(l))$. Thus, $hpjl_i(1, W) = c(j) + c(l) + hpjl(1, W - w(j) - w(l))$. This prove the correctness of definition of $hpjl_i(1, W)$.

For s ≥ 2, $G_i(s)$ consists of $G_i(q)$, $G(s)$ and edges $e_1 = (q_l, s_f)$ and $e_2 = (i, s)$. We have to consider the following three cases for $HPjl(s, W)$

1. $HPjl_i(s, W)$ contains edge e_1 but not e_2;
2. $HPjl_i(s, W)$ contains edge e_2 but not e_1;
3. $HPjl_{i(}s,W)$ contains edges both e_1 and e_2;

These three cases exhaust all possible configurations of $HPjl_i(s,W)$ and corresponding three sets used for computing $hpjl_i(s, W)$.

For computing the minimum cost c(HC,W), we simply need to exhaust all possibilities of the three edges (i_f , j), (i, j), (i_l, j). The three cases just correspond to the three numbers used for computing c(HC,W). Since there are $O(M + 1)$ choices for each of the numbers W, W_1, W_2 the update of the quantities can be accomplished in $O((M + 1)^2)$ times for each reduction. There are at most $O(n)$ reductions since the number of vertices is n requiring at most n fan contraction . This completes the proof of the theorem.

3. A Fully Polynomial Time Approximation Scheme (FPTAS) For WCTSP

Algorithm1 in section 2 solves WCTSP in pseudo polynomial time. Now to obtain a Fully Polynomial Time Approximation Scheme (FPTAS) for WCTSP, using standard techniques of scaling and rounding ([6], [3], [4]) for the costs. We will use algorithm 1 with costs replaced by the weights. Hence corresponding Cost Constrained Travelling Salesman Problem(CCTSP) is given as:

$$\min \text{å}_{ij\text{Î} E} \; w_{ij}x_{ij} \quad \text{such that} \quad \text{å}_{ij\text{Î} E} \; c_{ij}x_{ij} \text{ £ x and the constraints}$$

$$\begin{array}{l} x_{ij} \text{ £ " } ij \text{ Î } E \\ x \text{ Î } R^E : \text{å} \; (x_{ij} : ij \text{ Î d}(v)) = 2 \text{ " } v \text{ Î } V \\ \text{å} \; (x_{ij} : ij \text{ Î G}) = 2 \text{ for every 3-edge cut-set G of } H. \end{array}$$

Where x is the cost constant.

Thus, in this section algorithm1 will now find minimum weight Hamilton cycle with cost at most K, K = 1, 2, ..., x .

Let c(HC,W) be the minimum cost hamilton cycle with a weight no more than W. Using standard technique of scaling and rounding, we can decide, in fully polynomial time, whether $c(HC,W) > C$ or $c(HC,W) < (1$ + Î $)C$ for any constant Î > 0. This technique will play an important role in our FPTSP for computing a cost constrained minimum weight hamilton cycle in a Halin graph. We describe this technique in Algorithm2 as TEST.

Algorithm 2: *TEST*(*C*, Î)

Step 1: set θ := $\frac{n}{C \text{ Î r}}$; Let c_θ be scaled cost edge function such that $c_\text{q}(e) = c(e)$´ q for $e \in$ E and set x = C´ q;

Step 2: Apply Algorithm 1 to *CCTSP* using the scaled edge cost function c_θ instead of original cost function c; if the weight of the cost constrained hamilton cycle is no more than W then
output YES;
else
output NO;
endif

Theorem 3.1. *Let us be given the weight constant W, the positive real numbers C and ∫. If TEST* (*C*,ò) *= NO then c(HC, W) > C. If TEST* (*C*,ò) *= YES then c(HC, W) <* (1+ òð´ *C . In addition, the worst case time complexity of TEST* (*C*,ò) *is* $O\text{æ}\frac{n^3}{\text{ò}^2}\text{ö}$.

Proof. Let HC be the hamilton cycle in H. Let $c(HC)$ = $\text{å}_{e\text{Î} HC}\; c(e)$ and $c_\text{q}(HC)$ = $\text{å}_{e\text{Î} HC}\; c_\text{q}(e)$. θ and ξ are as defined in Algorithm 2.

Assume that *TEST* (*C*,ò) = NO, which implies every hamilton cycle HC with a cost less than ξ has a weight more than W. Conversely we must have hamilton cycle HC with weight no more than W having cost greater than ξ. Thus $c_\text{q}(HC) >$ x i.e.

$\text{å}_{e\text{Î} HC}\; c_\text{q}(e) > \text{x Þ q}\text{å}_{e\text{Î} HC}\; c(e) > C\text{qÞ}$

$\text{å}_{e\text{Î} HC}\; c(e) > C$, which implies that $c(HC,W) > C$.

Now assume that *TEST* (*C*,ò) = YES, which implies hamiloton cycle HC with a weight no more than W has cost less than or equal to ξ i.e. $w(HC) \leq W$ and $c_\text{q}(HC) \leq$ ξ.

Note that $c_q(HC) \le ξ$ implies $å_{\text{eî HC}}\, c_q(e) £ x$ and thus $c(HC) < C(1+ò)$ because $q\, å_{\text{eî HC}}\, c(e) £ Cq\, Þ\, q\, å_{\text{eî HC}}\, c(e) £ C(1+ò)q$. Since $Còq = n$ and there are n-edges in HC and $c(e)θ$ is a fraction. That is, $c(HC,W) < C(1+ò)$.

The time complexity $O\left(\frac{n^3}{ò^2}\right)$ of TEST follows from theorem 3, which says that Algorithm 1 has time complexity $O\left(n(M+1)^2\right)$; $O\left(n(M)^2\right)$. We get from Algorithm 2 $M := C´\, q$ and $q := \frac{n}{C,ò}$ so $O\left(n(M)^2\right) = O\left(\frac{n^3}{ò^2}\right)$ since $n(M)^2 = n(C\times\theta)^2 = n\left(C\times\frac{n}{Cò}\right)^2 = n\left(\frac{n}{ò}\right)^2 = \left(\frac{n^3}{ò^2}\right)$.

We shall use L and U to denote the lower and upper bounds on $c(HC, W)$. Our FPTSP starts with efficiently computable values of L and U and uses bisection to reduce the ratio $\frac{U}{L}$.

For a Halin graph with n vertices, we have at most 2n-2 edges. Let $c_1 < c_2 < ... < c_k$ be distinct edge cost values. We note that $k < 2n - 2$. Now we obtain a modified graph G_j by changing the weight of the edges as follows : If the edge e is more than c_j then w(e) = BIG. We can compute a minimum weight hamilton cycle in G_j in O(n) time for every j = 1, 2, ..., k. Let $J = min\{j\}$ such that the weight of the minimum hamilton cycle in G_j is no more than W. From the definition of G_j, no edge with weight BIG can be included in the minimum weight hamilton cycle of G_j , unless the weight of the tour is more than BIG. We may use c_J as the initial value of L and $n \times c_J$ as the initial value of U.

Let B be some chosen real number which is greater than 1 + ∫/3. We will apply bisection to drive the ratio $\frac{U}{L}$ down to some number below B. Suppose that our lower bound L and upper bound U are such that $U > L \times B > L \times (1+ò/3)$. Let

$$C = \sqrt{\frac{U´\, L}{1+ò/3}}.$$

If TEST $(C,ò/3)$ = NO, then C is a lower bound for c(HC).

Define $L^{[k]} = \sqrt{\frac{U^{[k-1]}´\, L^{[k-1]}}{(1+ò/3)}}, U^{[k]} = U^{[k-1]}$

If $TEST$ $(C,ò/3)$ = YES, then $(1+ò/3) \times C$ is also an upper bound for c(HC), i.e.

$$U^{[k]} = (1+ò/3)´\sqrt{\frac{U^{[k-1]}´\, L^{[k-1]}}{(1+ò/3)}}, L^{[k]} = L^{[k-1]}$$

(Case i). TEST $(C,ò/3) = NO$.

$$\frac{L^{[k]}}{L^{[k-1]}} = \sqrt{\frac{U^{[k-1]}´\, L^{[k-1]}}{(1+ò/3)}} / L^{[k-1]} = \sqrt{\frac{U^{[k-1]}}{L^{[k-1]}´\, (1+ò/3)}} > 1$$

Since $U^{[k-1]} > L^{[k-1]}´\, (1+ò/3)$.

Thus $L^{[k]} > L^{[k-1]}$. Hence $L^{[k]}$ is increasing.

(Case ii). TEST $(C,ò/3)$ = YES.

$$\frac{U^{[k]}}{U^{[k-1]}} = (1+ò/3)\sqrt{\frac{U^{[k-1]}´\, L^{[k-1]}}{(1+ò/3)}} / U^{[k-1]}$$

$$= \sqrt{\frac{L^{[k-1]}´\, (1+ò/3)}{U^{[k-1]}}} < 1$$

Since $U^{[k-1]} > L^{[k-1]}´\, (1+ò/3)$

i.e. $U^{[k]} < U^{[k-1]}$. Hence $U^{[k]}$ is decreasing.

Since the value of U is decreasing at each iteration while L is fixed and L is increasing while U is fixed, the ratio $\frac{U}{L}$ is decreasing. Therefore we find $\sqrt{\frac{U}{L}´\, (1+ò/3)}$ decreasing.

Let us call above the process an iteration. Note that such an iteration can be accomplished in fully polynomial time (by theorem 4, time complexity of $TEST$ $(C,ò/3)$ is $O\left(\frac{n^3}{ò^2}\right)$). Furthermore, the ratio of the upper bound over the lower bound can be reduced to a number below B in polynomial number of iterations(polynomial in the input size of the given instance and $\frac{1}{ò}$). This analysis leads to a FPTSP scheme which is given in algorithm 3. We summarize the discussion in theorem 3.2.

Algorithm 3

FPTSP for weight constrained minimum cost Hamilton cycle on a Halim graph.

```
step 1
  set $B := (1+ò/3)´\,(1+ò/3)$; set $L = c_J$ and $U = n´\, c_J$
so that $\frac{U}{L} = n$
step 2
  if $U £ B´\, L$ then go to step -3;
else
  let $C := \sqrt{\frac{U´\, L}{1+ò/3}}$;
  if TEST $(C,ò/3)$ = NO, set L = C;
  if TEST $(C,ò/3)$ = YES, set $U = (1+ò/3)´\, C$;
  go to step-2;
endif
step 3
  set $q := \frac{n}{L´\, ò/3}$; $x := q´\, U$;
set $c_q(e) := \left\lfloor q´\, c(e)\right\rfloor$ for every $eÎ\, E$;
apply algorithm 1 to compute a cost constrained minimum weight hamilton cycle using the scaled cost function $c_q$.
```

Theorem 3.2. *Algorithm 3 finds a weight constrained hamilton cycle HC such that w(HC) ≤ W and c(HC) ≤ (1 + ∫) × c(HC, W). Furthermore the time complexity of Algorithm 3 is* $O\left(\frac{n^3}{ò^2}´\left(logn + log\frac{1}{ò}\right)\right)$.

Proof. If there is no hamilton cycle satisfying the weight constraint, we will find this out during our computation of the initial value of L and U. Since U is within $(1+\epsilon)^2$ of L, $c(HC,W) \in (L,U) \Rightarrow c(T)$ is within $(1+\epsilon)$ of U is also within $(1+\epsilon)$ of $c(HC,W)$. The complexity part is as follows.

Step2 in algorithm 3, can be done in *O*(*logn* + *log*1/∫) time because we need to carry out till the ratio of upper bound over the lower bound can be reduced to a number below B as discussed below.

$$\frac{U^{[k]}}{L^{[k]}} = (1+\epsilon/3)^{1/2}\frac{U^{[k-1]}}{L^{[k-1]}}$$

Taking log on both sides, we get

$$log\frac{U^{[k]}}{L^{[k]}} = \frac{1}{2}log(1+\epsilon/3) + \frac{1}{2}log\frac{U^{[k-1]}}{L^{[k-1]}}$$

$$= \frac{1}{2}log(1+\epsilon/3) + \frac{1}{2}\left(\frac{1}{2}log(1+\epsilon/3) + \frac{1}{2}log\frac{U^{[k-2]}}{L^{[k-2]}}\right)$$

$$= \left(\frac{1}{2} + \frac{1}{2^2} + \ldots + \frac{1}{2^k}\right)log(1+\epsilon/3) + \frac{1}{2^k}log\frac{U^{[0]}}{L^{[0]}}$$

$$= \left(\frac{1}{2} + \frac{1}{2^2} + \ldots + \frac{1}{2^k}\right)log(1+\epsilon/3) + \frac{1}{2^k}logn$$

$$\leq 1 + \frac{1}{2^k}logn$$

Since we need $\frac{U^{[k]}}{L^{[k]}} \leq (1+\epsilon/3)^2$

$$\Rightarrow 2log(1+\epsilon/3) \leq 1 + \frac{1}{2^k}logn$$

This holds if $(1+2\epsilon/3) < 1 + \frac{1}{2^k}logn$

$$\Rightarrow 2^k < \frac{logn}{log(1+\epsilon/3)}$$

$$\Rightarrow k < log\ logn - log(1+\epsilon/3)$$

$$\Rightarrow k < log\ logn + log\ log\frac{1}{(1+\epsilon/3)}$$

$$\Rightarrow k < log\ logn + log\ log\frac{1}{\epsilon}$$

In step 2, TEST $(C,\epsilon/3)$ can be checked in $O\left(\frac{n^3}{\epsilon^2}\right)$ and algorithm 2 is used once in step 3 having time $O\left(n(x+1)^2\right)$.

Thus the time complexity of algorithm 3 is

$$O\left(\frac{n^3}{\epsilon^2} \times \left(logn + log\frac{1}{\epsilon}\right) + n(x+1)^2\right)$$

$$= O\left(\frac{n^3}{\epsilon^2} \times \left(logn + log\frac{1}{\epsilon}\right) + n\left(\frac{n}{\epsilon}\right)^2\right)$$

$$= O\left(\frac{n^3}{\epsilon^2} \times \left(logn + log\frac{1}{\epsilon}\right)\right).$$

4. Conclusions

In this paper we have studied the weight constrained minimum cost hamilton tour on a very important class of graphs - Halin graphs. It is shown that the weight constrained travelling salesman problem on Halin graphs is NP-hard. We present a pseudo-polynomial time algorithm for computing a weight constrained minimum cost Hamilton tour in a Halin graph, and also present a fully polynomial time approximation scheme for this problem. The more challenging problem is the WCTSP on a general graph. It would be worth investigating if it is possible to reduce the WCTSP on a general graph to WCTSP on a Halin graph and obtain good approximate solutions to WCTSP on a general graph.

References

[1] G. Cornnuejols, D. Naddef, and W.R. Pulleyblank, *Halin Graphs and Travelling Salesman Problem, Mathematical Programming*, Mathematical Programming, 26(1983), 287-294.

[2] G.T.Chen and G.L. Xue, *K-pair delay constrained minimum cost routing in undirected networks*, SODA 2001-ACM -SIAM Symposium on Discrete Algorithms, pp 230-231.

[3] R. Hussain, *Approximation scheme for the restricted shortest path problem*, Mathematics of Operation Research,Vol.17(1992), pp. 36-42.

[4] O. Ibarra and C. Kim, *Fast Approximation Algorithm for the Knapsack and the Sum of Subsets Problems*, Journal of ACM, vol. 22, pp. 463-468.

[5] G. Chen and R. E. Burkard, *Constrained Steiner trees in Halin graphs*, RAIRO Oper. Res. 37 (2003) 179-194 DOI: 10.1051/ro:2003020.

[6] G. Chen and G. Xue, *An FPTAS for Weight-Constrained Steiner Trees in Series Parallel Graphs*, Theoretical Computer Science, Volume 304, Number 1, 28 July 2003, pp. 237-247.

[7] J.A Wald and C.J. Colbourn, *Steiner Trees, Partial 2-trees and minimum IFI networks*, Networks, Vol. 13(1983), pp. 159-167.

[8] Reinelt G.(1994), *The Travelling Salesman Problem Computational Solutions for TSP Applications*, Lecture notes in Computer Science, Volume page 37-66 840/1994 Springer-Verlag, Heidelberg, Berlin.

[9] Gilmore P.C., Lawler E.L., and Shmoys D.B.,(1985) Well solved special cases, in E.L. Lawler et. al. (eds), *The Travelling Salesman Problem: A guided tour of combinatorial optimization*, Wiely, N.Y.

[10] Glover, F., and Punnen, A.P., (1997) *The Travelling Salesman Problem: New Solvable Cases and Linkage with the Development of the Approximation Algorithms*, Journal of Operational Research Society, 48 (1997) 502-510.

[11] Lawler E.L., Lenstra J.K., and Shmoys D.B.,(1985) *The Travelling Salesman Problem*, Wiely- Inter science Series in Discrete Mathematics, and Sons, New York.

Author Profile

Dr. Dharmananda Gahir received the Ph.D. degree in Mathematics from Indian Institute of Technology Kanpur, India in 2009.