Annals of Discrete Mathematics 25 (1985) 27-46

A LOCAL-RATIO THEOREM FOR APPROXIMATING THE WEIGHTED VERTEX COVER PROBLEM

R. BAR-YEHUDA and S. EVEN

Computer Science Department Technion - Israel Institute of Technology
Haifa, Israel 32000

A local-ratio theorem for approximating the weighted vertex cover problem is presented. It consists of reducing the weights of vertices in certain subgraphs and has the effect of local-approximation.

Putting together the Nemhauser-Trotter local optimization algorithm and the local-ratio theorem yields several new approximation techniques which improve known results from time complexity, simplicity and performance-ratio point of view.

The main approximation algorithm guarantees a ratio of $2-\frac{1}{\kappa}$ where κ is the smallest integer s.t. $(2\kappa-1)^{\kappa}\geq n$ (hence: ratio $\leq 2-\frac{loglogn}{2logn}$) †

This is an improvement over the currently known ratios, especially for a "practical" number of vertices (e.g. for graphs which have less than 2400, 60000, 10^{12} vertices the ratio is bounded by 1.75, 1.8, 1.9 respectively).

1. Introduction

A *Vertex Cover* of a graph is a subset of vertices such that each edge has at least one endpoint in the subset. The *Weighted Vertex Cover Problem* (WVC) is defined as follows: Given a simple graph $G(V, E)$ and a weight function $\omega: V \to R^{+}$, find a cover of minimum total weight. WVC is known to be NP–Hard, even if all weigts are 1 [16] and the graph is planar [8]. Therefore, it is natural to look for efficient approximation algorithms.

Let A be an approximation algorithm. For graph G with weight functions ω, let C_A, C^* be the cover A produces and an optimum cover,

† *All log bases, in this paper, are 2.*

respectively. Define

$R_A(G, \omega) = \dfrac{\omega(C_A)}{\omega(C^*)}$ and let the performance ratio $r_A(n)$ be

$$r_A(n) = Sup\left\{R_A(G, \omega) \mid G = (V, E) \text{ where } n = |V|\right\}.$$

Many approximation algorithms with performance ratio ≤ 2 have been suggested; see, for example Table 1. No polynomial-time approximation algorithm A is known for which $r_A(n) \leq 2 - \varepsilon$, where $\varepsilon > 0$ and fixed. Several approximation algorithms are known for which $R_A(G, \omega) \leq 2 - \varepsilon(G)$, where ε depends on G; e.g. $\varepsilon(G) = \dfrac{2}{\Delta(G)}$ where $\Delta(G)$ is the maximum degree of the vertices of G.

In Section 2 we review the Nemhauser and Trotter [18] local optimization algorithm (NT) as well as the observation of Hochbaum [11] to use it for approximating WVC.

In Section 3, a new theorem, the 'Local-Ratio Theorem' is presented and proved. It consists of reducing the weights of vertices in certain subgraphs and has the effect of local approximation. As an example of its power we present a trivial proof for the correctness of a linear time approximation algorithm $COVER1$ with $r_{COVER1}(n) \leq 2$.

In Section 4, we present two approximation algorithms in which the NT algorithm and the local-ratio theorem are shown to be useful. The first algorithm $COVER2$ satisfies $r_{COVER2}(n) \leq 2 - \dfrac{1}{\sqrt{n}}$ for general graphs, while for planar graphs $r_{COVER2}(n) \leq 1.5$ and its time complexity is $O(n^2 \log n)$. Hochbaum [12] obtained the same performance ratio, but we manage to avoid the time complexity of 4-coloring.

For our last algorithm, $COVER3$, we prove that $r_{COVER3}(n) \leq 1 - \frac{1}{k}$, where k is the least integer s.t. $(2k-1)^k \geq n$. A similar result, but only for unweighted graphs, has been obtained independently by Monien and Speckenmeyer [17].

In Section 5, the Local-Ratio Theorem is extended to the *Weighted Set-Cover Problem.*

TABLE 1–SUMMARY OF APPROXIMATION RESULTS.
(for NT, see Table 2)

References	*Performance ratio* $\leq$	*Complexity for weighted*	*Complexity for unweighted*
[9]	2	not applicable	E
[11]	2	V^3	
[1]	2	E	
[12]	$2-\frac{2}{\Delta}$	"NT"	"NT"
this paper	$2-\frac{2}{\sqrt{V}}$	"NT"	"NT"
this paper	$2-\frac{\log\log V}{2\log V}$	"NT"	EV
(For planar graphs)			
[12]	1.6	"NT"	$V^{1.5}$
[12]	1.5	"NT" + "4–COLORING"	"NT" + "4–COLORING"
this paper	1.5	"NT"	"NT"
[2]	$1\frac{2}{3}$	not applicable	V
[5]*	$1+\varepsilon$	not applicable	Vlog V

* *Although this result seems much better then the one shown 2 lines above for unweighted graphs, the algorithm is not useful for practical computations. For example for the algorithm to achieve a ratio* ≤ 2 *one may need ad many as* $2^{2^{160}}$ *vertices* [6]. *For a similar algorithm see* [2].

TABLE 2–"NT"'s Complexity (the same as MAX FLOW)		
$G=(V,E)$	*Weighted*	*Unweighted*
General	$E^{2/3}V^{5/3}$ or $EV\log V$	$E\sqrt{V}$
Planar	$V^2\log V$	$V^{1.5}$

2. The Nemhauser and Trotter local optimization algorithm

In this short section we review the local optimization algorithm of Nemhauser and Trotter [18] which is very useful for approximations of WVC [12].

Let $G(V, E)$ be a simple graph. We denote by $G(U)$ the subgraph of G induced by $U \subseteq V$ and let $U' = \{u' | u \in U\}$. Define the weights of vertices in U' by $\omega(u') = \omega(u)$.

Algorithm *NT*

Input: $G(V, E)$, ω.

Phase 1: Define a bipartite graph $B(V, V', E_B)$ where $E_B = \{(x, y') \mid (x, y) \in E\}$.

Phase 2: $C_B \leftarrow C^*(B)$.

Output: $C_0 \leftarrow \{x \mid x \in C_B \text{ } AND \text{ } x' \in C_B\}$
$V_0 \leftarrow \{x \mid x \in C_B \text{ } XOR \text{ } x' \in C_B\}$

The following theorem states results of Nemhauser and Trotter.

The NT–Theorem: The sets C_0, V_0 which Algorithm *NT* produces, satisfie the following properties:

(i) If a set $D \subseteq V_0$ covers $G(V_0)$ then $C = D \cup C_0$ covers G.

(ii) There exists an optimum cover $C^*(G)$ such that $C^*(G) \supseteq C_0$.
[(i) and (ii) are called the local optimally conditions.]

(iii) $\omega(C^*(G(V_0))) \geq 1/2\, \omega(V_0)$.

For a proof see [18]. An alternate proof is given in the APPENDIX.

The significance of the NT–Theorem, as pointed out by Hochbaum [12], is that the problem of approximating WVC can be limited to graphs $G(V_0)$ which satisfy (iii). A simple illustration of this approach was used by Hochbaum to obtain an algorithm with performace ratio ≤ 2 : Call $NT(G(V),\omega)$ to get C_0, V_0 and return $C = C_0 \cup V_0$.

Let us consider now the time-complexity of finding $C^*(B)$, which determines the time-complexity on NT.

For the unweighted case the problem can be converted into the maximum matching problem on B (see for example [3] which is of time-complexity $O(E\sqrt{V})$ (see [14]).

For the weighted case the problem can be converted into a maximum flow problem (see, for example [12]) which is of time-complexity $O(E^{2/3}\ V^{5/3})$ [10] or $O(EV\log V)$ [19]. For a summary of the results, see Table 2.

3. The local-ratio theorem

In a previous paper [2] we presented a local approximation technique for the vertex cover problem of unweighted graphs. In this section we present a local approximation technique for the vertex cover problem of weighted graphs. First, we present the following lemma:

Lemma: Let $G(V, E)$ be a graph and ω, ω_1 and ω_2 be weight functions on V, s.t. for every $v \in V$: $\omega(v) \geq \omega_1(v) + \omega_2(v)$. Let C^*, C_1^* and C_2^* be optimum covers of G with respect to ω, ω_1 and ω_2. It follows that: $\omega(C^*) \geq \omega_1(C^*) + \omega_2(C_2^*)$.

Proof: $\omega(C^*) = \sum_{v \in c^*} \omega(v)$

$$\geq \sum_{v \in c^*} (\omega_1(v)+\omega_2(v))$$

$$= \omega_1(C^*)+\omega_2(C^*)$$

$$\geq \omega_1(C^*)+\omega_2(C^*) \quad \text{[by the optimality of } C_1^* \text{ and } C_2^*\text{]}$$

Q.E.D.

Let $\bar{G}$ be an unweighted graph of $\bar{n}$ vertices, whose optimum cover contains $\bar{c}^*$ vertices. Define $\bar{r}=\bar{n}/\bar{c}^*$. Let A be an approximation algorithm for WVC and let $LOCAL_{\bar{G}}$ be the following algorithm:

Algorithm $LOCAL_{\bar{G}}$

Input: Graph $G(V, E)$, with weight function ω.

Phase 1: Choose a subgraph $\tilde{G}(\tilde{V}, \tilde{E})$ of G which is isomorphic to $\bar{G}$.

Choose $0 \leq \delta \leq Min\{\omega(x) \mid x \in \tilde{V}\}$

$$\text{Define } \omega_0(x) = \begin{cases} \omega(x)-\delta & \text{if } x \in \tilde{V} \\ \omega(x) & \textit{else.} \end{cases}$$

Phase 2: Call $A(G, \omega_0)$ to get C_0.

Output: $C \leftarrow C_0$.

The Local-Ratio Theorem: $R_{LOCAL_{\bar{G}}}(G, \omega) \leq Max\{\bar{r}, R_A(G, \omega_0)\}$

Proof: Let c^* and c_0^* be the weights of the optimum covers of G with respect to ω and ω_0, and let $r = Max\{\bar{r}, R_A(G, \omega_0)\}$, then

$$\omega(C) \quad \leq \quad \omega_0(C)+\delta \cdot \bar{n} \qquad [\text{by } |C \cap \tilde{V}| \leq n]$$

$$\leq R_A(G, \omega_0) \cdot c_0{}^* + \bar{r} \cdot \delta \cdot \bar{c}^* \qquad \text{[by definitions]}$$

$$\leq r \cdot (c_0{}^* + \delta \bar{c}^*) \qquad \text{[by } r\text{'s definition]}$$

$$\leq r \cdot c^* \qquad \text{[by the lemma]}$$

Q.E.D.

Let us consider now a corollary of the Local-Ratio Theorem. Let Γ be a finite family of graphs, and $r_\Gamma = Max\{\bar{r} \mid \bar{G} \in \Gamma\}$.

We denote by $G(U)$, $U \subseteq V$, the subgraph of $G(V, E)$ induced by U.

Algorithm $LOCAL_\Gamma$

Input: $G(V, E)$, ω.

Phase 0: **For** every $x \in V$ **do** $\omega_0(x) \leftarrow \omega(x)$ **end**

Phase 1: **For** every $\tilde{G}(\tilde{V}, \tilde{E})$, subgraph of G which is isomorphic to some $\bar{G} \in \Gamma$, **do**

$\delta \leftarrow Min\{\omega_0(x) \mid x \in \tilde{V}\}$.

For every $x \in \tilde{V}$ **do** $\omega_0(x) \leftarrow \omega_0(x) - \delta$ **end**

end

Phase 2: $C_1 \leftarrow \{x \mid \omega_0(x) = 0\}$.

$V_1 \leftarrow V - C_1$.

Call $A(G(V_1), \omega_0)$ to get C_2.

Output: $C \leftarrow C_1 \cup C_2$.

The Local-Ratio Corollary:

$$R_{LOCAL_\Gamma}(G, \omega) \leq Max\{r_\Gamma, R_A(G(V_1), \omega_0)\}.$$

Proof: By induction on i, the number of iterations of Phase 1, during which $\delta > 0$.

For $i=0$ the claim is trivial.

Suppose the claim holds for i, and for some G, ω Phase 1 runs $i+1$ iterations during which $\delta > 0$. Let $\bar{G} \in \Gamma$ be the graph used in the first iteration during which $\delta > 0$. Observe that we can view the running of $LOCAL_\Gamma$ as an application of $LOCAL_{\bar{G}}$ where A (in Phase 2 of $LOCAL_{\bar{G}}$) is replaced by the remaining part of $LOCAL_\Gamma$. The inductive step is now an immediate consequence of the Local-Ratio Theorem. .

Q.E.D.

In the applications of $LOCAL_\Gamma$ we shall refer to r_Γ as (a bound on) the local-ratio of Phase 1. Let us demonstrate a simple application of the Local-Ratio Corollary.

Algorithm *COVER* 1

Input: $G(V, E), \omega$.

Phase 1: **For** every $e \in E$ **do**

Let $\delta = Min\{\omega(x) | x \in e\}$.

For every $x \in e$ **do** $\omega(x) = \omega(x) - \delta$ **end**

end

Output: $C \leftarrow \{x \mid \omega(x) = 0\}$.

This approximation algorithm is essentially the one we described in [1][1].

Proposition: For algorithm *COVER* 1.

(1) The time complexity is $O(E)$

(2) Its performance ratio ≤ 2

Proof:

(1) The number of operations spent on each edge is bounded by a

[1] *In [1] we used a global rather then a local point of view.*

constant.

(2) Using the Local-Ratio Corollary, with Γ which is a single edge.

Q.E.D.

4. Putting together NT and the local-ratio theorem

Hochbaum [12] suggested the following approach to approximate WVC: Let $G(V, E)$, ω be the problem's input, such that $\omega(C^*(G)) \geq 1/2\ \omega(V)$. (This is achieved by the NT algorithm). Color G by k colors and let I be the "heaviest" monochromatic set of vertices. The cover produced is $C=V-I$. It follows that

$$\frac{\omega(C)}{\omega(C^*)} = \frac{\omega(V)-\omega(I)}{\omega(C^*)} \leq \frac{\omega(V)-\omega(V)/\kappa}{1/2\ \omega(V)} = 2 - \frac{2}{k}.$$

For general graphs she gets the ratio $2 - \frac{2}{\Delta}$ (Δ is the maximum degree) and for planar graphs ($k = 4$) the performance ration ≤ 1.5 in time-complexity of NT and 4-coloring.

We suggest the use of a preparatory algorithm in which all triangles are omitted (with local-ratio 1.5) and therefore, the residual graph is easier to color.

Algorithm *COVER*2

Input: $G(V, E)$, ω, k.

Phase 1: [Triangle elimination]

For every triangle $T\,(T \subseteq V)$ do

  Find $\delta = Min\,\{\omega(x) \mid x \in T\}$

  For every $x \in T$ do $\omega(x) \leftarrow \omega(x) - \delta$ end

end

$C_1 \leftarrow \{x \mid \omega(x) = 0\}$

$V_1 \leftarrow V - C_1$.

Phase 2: Call $NT(G(V_1), \omega)$ to get C_0, V_0.

Phase 3: Find a cover approximation, C_2, of $G(V_0)$, by using k-coloring (as in Hochbaum's approach).

Output: $C \leftarrow C_1 \cup C_0 \cup C_2$.

For $k \geq 4$, $R_{COVER2}(G, \omega) \leq 2 - \frac{2}{k}$, since the local-ratio of Phase 1 is 1.5 (here Γ contains of a single graph $\bar{G}$ which is a triangle), the local-ratio of Phase 2 is 1 and the local-ratio of Phase 3 is $2 - \frac{2}{k}$.

Since the graph colored in Phase 3 is triangle-free, we get the following additional results:

(1) For general graphs $r_{COVER2}(n) \leq 2 - \frac{1}{\sqrt{n}}$ by using Wigderson's approach [20] for coloring a triangle-free graph by $k = 2\sqrt{n}$ colors in linear time [21].

(2) For planar graphs $r_{COVER2}(n) \leq 1.5$, and the time-complexity of 4-Coloring a triangle-free planar graph is linear. (One uses the fact that in such graphs there is always a vertex of degree≤ 3. See, for example, [13]. This prevents the need to use a more complex 4-Coloring algorithms.
Note that the 1.5 performance ratio, for unweighted planar graphs, can be achieved by the "$1 + \varepsilon$ algorithm" of [5] (or [2]), however their algorithm is expo-exponential time w.r.t. $1/\varepsilon$ and for $\varepsilon = 0.5$ is not practical.

Before we present our main algorithm we need a few preliminaries. Let the triple $(G(V, E), \omega, k)$ (where G is a graph with weight function ω and k is a positive integer) be called *proper* if the following conditions hold:

(i) $(2k-1)^k \geq |V|$.

(ii) There are no odd circuits of length $\leq 2k-1$.

(iii) $\omega(C^*) \geq 1/2\, \omega(V)$.

In the following procedure the statements in square brackets are added for the analysis only, and variables j (integer), C_0, V_0, C_1, V_1,... (sets), are used only in the brackets.

Procedure *COVER.PROPER*

Input: proper $(G(V, E), \omega, k)$.

Phase 0: V' $V' \leftarrow V$, $C' \leftarrow \phi$ · $[j \leftarrow 0]$

Phase 1: **While** $V' \neq \phi$ **do**

Find $\nu \in V'$ s.t. $\omega(\nu) = Max\{\omega(u) \mid u \in V'\}$.

Let $A_0, A_1, \ldots, A_k$ be the first $k+1$ layers†

of a Breadth-First-Search (BFS‡) on $G(V')$ starting with $A_0 = \{\nu\}$

Define $B_{2t} = \bigcup_{i=0}^{t} A_{2i}$ and $B_{2t+1} = \bigcup_{i=0}^{t} A_{2i+1}$

(for t=0, 1, 2, ...)

$f \leftarrow Min\{s \mid \omega(B_s) \leq (2\kappa - 1) \cdot \omega(B_{s-1})\}$.

Add B_f to C'. $[C_j \leftarrow B_f]$

Remove $B_f \cup B_{f-1}$ from V'. $[V_j \leftarrow B_f \cup B_{f-1}, j \rightarrow j+1]$

end

Output: $C \leftarrow C'$.

Proposition 1: Procedure *COVER.PROPER* satisfies the following properties:

(1) In every application of Phase 1, $f \leq \kappa$.

(2) In every application of Phase 1, B_{f-1} is an independent set in $G(V')$.

(3) C covers G.

† *Starting with some* m, $A_m, A_{m+1}, \ldots, A_k$ *may be empty.*

‡ *See for example* [7].

(4) For every iteration j, $\omega(C_j) \leq (1 - \frac{1}{2k}) \cdot \omega(V_j)$.

(5) $\omega(C) \leq (1 - \frac{1}{2k}) \cdot \omega(V)$.

(6) $R_{COVER.PROPER}(G) \leq 2 - \frac{1}{k}$.

(7) The time complexity is $0(|V| \log |V| + |E|)$.

Proof:

(1) Assume the contrary.
Thus, for every $s \leq k$, $\omega(B_s) > (2k-1) \cdot \omega(B_{s-1})$. Thus,

$\omega(B_k) > (2k-1)^k \cdot \omega(B_0)$	[by the assumption]
$\geq |V| \cdot \omega(B_0)$	[by (i) of the definition of proper]
$= |V| \cdot \omega(\nu)$	[$B_0 = A_0 = \{\nu\}$]
$\geq |V'| \cdot \omega(\nu)$	[$V' \subseteq V$]
$\geq \omega(V')$	[by ν 's definition]
$\geq \omega(B_k)$	[$B_k \subseteq V'$]

which is absurd.

(2) An edge between two vertices of A_s, $s < k$, implies the existence of an odd-circuit of length $\leq 2k-1$. This contradicts condition (ii) of properness.

(3) For every iteration j, all edges which are (indirectly) deleted are covered by the current C_j, which joins C'. This follows from the properties of BFS and (2) above.

(4) For every iteration j, $\omega(B_f) \leq (2k-1) \cdot \omega(B_{f-1})$. Since $B_f = C_j$, and $B_{f-1} = V_j - C_j$, we have $\omega(C_j) \leq (2k-1) \cdot [\omega(V_j) - \omega(C_j)]$. This implies the stated inequality.

(5) By summation of the inequality of (4) for every j.

(6) By definition $R_{COVER.PROPER}(G, \omega) = \dfrac{\omega(C)}{\omega(C^*)}$. Property (5) above and condition (iii) of properness imply that

$$R_{COVER.PROPER}(G, \omega) \leq \frac{(1-\frac{1}{2k})\cdot \omega(V)}{1/2\, \omega(V)}.$$

(7) We may start the algorithm by sorting the vertices according to their weights, which requires $O(|V| log|V|)$ steps. This complexity includes, now, the total time used in Phase 1 for finding $Max\{\omega(u) \;|u \in V'\}$.
It is not necessary to continue the BFS of Phase 1, beyond layer f. Thus, each such search is linear in the number of edges to be eliminated from $G(V')$, and the total time spent in the search of Phase 1 is linear in $|E| + |V|$.
Thus (7) follows.

Q.E.D.

Now, the main algorithm.

Algorithm $COVER3$

Input: $G(V,E)$, ω

Phase 0: Find the least integer k s.t. $(2k-1)^k \geq |V|$.

Phase 1: [Elimination of short odd circuits with local-ratio $\leq 2-\frac{1}{k}$].

For every odd circuit $D \subseteq V$ s.t $|D| \leq 2k-1$ **do**

$\delta \leftarrow Min\{\omega(x)|x \in D\}$

For every $x \in D$ **do** $\omega(x) \leftarrow \omega(x) - \delta$ **end**

end

$C_1 \leftarrow \{x | \omega(x) = 0\}$.

$V_1 \leftarrow V - C_1$.

Phase 2: Call $NT(G(V_1), \omega)$ to get C_0, V_0.

Phase 3: Call $COVER.PROPER(G(V_0), \omega, k)$ to get C_2.

Output: $C \leftarrow C_1 \cup C_0 \cup C_2$.

Proposition 2: Algorithm $COVER3$ satisfies the following properties:

(1) $r_{COVER3}(n) \le 2 - \frac{1}{k}$.

(2) Its time complexity is the same as NT's. (see Table 2)

(3) for unweighted graphs its time complexity is $O(|V| \cdot |E|)$.

Proof:

(1) The combination of Phase 2 and 3 yields an algorithm with performance ratio $\le 2 - \frac{1}{k}$, since the NT algorithm performs local-optimization (ratio=1) and, for proper graphs, $COVER.PROPER$ has performance ratio $\le 2 - \frac{1}{k}$ [by Proposition 1(6)]. Let $\bar{G}_l$ be a simple circuit of length $2l-1$ thus, $\bar{n}_l = 2l-1$ and $\bar{c}_l^* = l$; therefore, $\bar{r}_l = (2l-1)/l = 2-1/l$. Consider, now, the Local-Ratio-Corollary where, $\Gamma = \{\bar{G}_l \mid l \le k\}$, thus, $r_\Gamma = Max\{\bar{r}_l \mid l \le k\} = 2 - 1/k$ and (1) follows.

(2) Let us perform Phase 1 in a slightly different way: Choose a vertex v, and build the first k layers of the BFS starting from v. If there is an edge $u - \omega$, where u and ω belong to the same layer, then an odd-length-circuit D has been detected (see for example [15]). In this case we find $\delta = Min\ \{\ \omega(x) \mid x \in D\ \}$, reduce the weights of the vertices in D by δ and the vertices whose weight is zero are eliminated from the representation of the graph (for the purpose of performing Phase 1). If no such edge (closing an odd circuit) is detected then v is eliminated from the representation of the graph, since no odd-circuit of length $\le 2k-1$ passes through it. In any case at least one vertex is eliminated for each BFS. Thus the time complexity is $O(|V| \cdot |E|)$.
In Phase 2, the best known time-complexity of the NT algorithm

(table 2) is greater then $|V| \cdot |E|$. Phase 3 requires $O(|E|+|V|log|V|)$, by Proposition 1(7).
Thus, the whole algorithm is of time complexity as the NT algorithm.

(3) For unweighted graphs, the NT algorithm can be performed in $O(|E|\sqrt{|V|})$ time and therefore, the algorithm is of time complexity $O(|V| \cdot |E|)$.

Q.E.D.

Corollary: $r_{COVER3}(n) < 2 - \frac{loglogn}{2logn}$.

Proof: Define $g(n) = \frac{logn}{loglogn}$ which is monotone increasing for $n \geq 16$. By k-th definition $(2k-3)^{k-1} < n$. Thus, $g((2k-3)^{k-1}) < g(n)$.

We want to show that $k < 2g(n)$. Thus, it suffices to show that $k < 2g((2k-3)^{k-1})$. This is an exercise in elementary mathematics.

Q.E.D.

5. Extending the local ratio theorem for the weighted set cover problem

Let HWVC be the following problem. Given a hypergraph $G=(V,E)$ with weight function $\omega: V \to R^+$, find a set $C \subseteq V$ of minimum total weight s.t. for every $e \in E$, $|e \cap C| \geq 1$. The local-Ratio Theorem and its Corollary hold also for HWVC. Algorithm $COVER1$ can be applied directly to HWVC with $r_{COVER1} \leq \Delta_E$ [where Δ_E is the maximum edge-degree (or cardinality) in G]. Its running time is linear in the length of the problem's input ($\sum |e|$). HWVC is actually the Weighted-Set-Cover Problem[2] and is an extension of WVC [$\Delta_E = 2$]. Even

[2]Chvatal [4] gets, performance-ratio $\leq \sum_{j=1}^{\Delta} \frac{1}{j} = O(log\,\Delta)$.
[where Δ is the maximum vertex degree in G].

though we get performance-ratio$\leq \Delta_E$ in linear time, we suspect that for any fixed Δ_E, there is no polynomial time approximation algorithm with a better constant performance ratio, unless P=NP [even for the unweighted case], (this is an extension of a conjecture of Hochbaum).

6. Appendix – NT theorem

Let $G(V, E)$ be a simple graph. We denote by $G(U)$ the subgraph of G induced by $U \subseteq V$ and let $U' = \{u' \mid u \in U\}$. Define the weights of vertices in U' by $\omega(u') = \omega(u)$.

Nemhauser and Trotter [18] presented the following local optimization algorithm:

Algorithm NT

Input: $G(V, E)$, ω.

Phase 1: Define a bipartite graph $B(V, V', E_B)$ where $E_B = \{(x, y') \mid (x, y) \in E\}$.

Phase 2: $C_B \leftarrow C^*(B)$.

Output: $C_0 \leftarrow \{x \mid x \in C_B \text{ } AND \text{ } x' \in C_B\}$
$V_0 \leftarrow \{x \mid x \in C_B \text{ } XOR \text{ } x' \in C_B\}$

The following theorem states results of Nemhauser and Trotter, but our proof is shorter and does not use linear programming arguments.

The NT–Theorem: The sets C_0, V_0 which Algorithm NT produces, satisfies the following properties:

(i) If a set $D \subseteq V_0$ covers $G(V_0)$ then $C = D \cup C_0$ covers G.

(ii) There exists an optimum cover $C^*(G)$ such that $C^*(G) \supseteq C_0$.

(iii) $\omega(C^*(G(V_0))) \geq 1/2\, \omega(V_0)$.

Proof: Define $I_0 = \{x \mid x \notin C_B \text{ AND } x' \notin C_B\} = V-(V_0 \cup C_0)$

Let $(x, y) \in E$. In order to prove (i) we need to show that either $x \in C$ or $y \in C$.

Case 1: $x \in I_0$, i.e. $x, x' \notin C_B$. Thus, $y, y' \in C_0$ and therefore $y \in C_0$.

Case 2: $y \in I_0$. Same as Case 1.

Case 3: $x \in C_0$ or $y \in C_0$. This case is trivial.

Case 4: $x, y \in V_0$. Thus, either $x \in D$ or $y \in D$.

In order to prove (ii), let $S = C^*(G)$. Define $S_V = S \cap V_0$, $S_C = S \cap C_0$, $S_I = S \cap I_0$ and $\bar{S}_I = I_0 - S_I$.
Let us show that:

$$C_{B_1} = (V-\bar{S}_I) \cup S'_c \text{ covers } B. \qquad (*)$$

Let $(x, y') \in E_B$. We need to show that either $x \in C_{B_1}$ or $y' \in C_{B_1}$.

Case 1: $x \notin \bar{S}_I$. Thus, $x \in V-\bar{S}_I$ $[\subseteq C_{B_1}]$

Case 2: $x \in \bar{S}_I$. Thus, $x \in I_0$, $x \notin S$ and therefore $x \notin C_0$. It follows that $y \in S$ [since S covers (x, y)] and $y \in C_0$ [by Case 1, in the proof of (i)].
Thus, $y \in S \cap C_0$ $[=S_c]$ and therefore $y' \in S'_c$ $[\subseteq C_{B_1}]$.

This proves (*). Now,

$$\begin{aligned}
\omega(V_0) + 2\omega(C_0) &= \omega(V_0 \cup C_0 \cup C'_0) \\
&= \omega(C_B) && [\text{by definitions of } V_0, C_0] \\
&\leq \omega(C_{B_1}) && [\text{by } (*) \text{ and optimality of } C_B] \\
&= \omega((V-\bar{S}_I) \cup S'c) \\
&= \omega(V_0 \cup C_0 \cup S_I \cup S'_c). \\
&= \omega(V_0) + \omega(C_0) + \omega(S_I) + (S_c).
\end{aligned}$$

It follows that $\omega(C_0) \leq \omega(S-S_V)$. Thus, $\omega(C_0 \cup S_V) \leq \omega(S)$. However, $C_0 \cup S_V$ covers G [by (i)] and therefore $C_0 \cup S_V$ is an optimum cover of G and contains C_0. This proves condition (ii).

In order to prove (iii), assume S_0 is an optimum cover of $G(V_0)$. By (i), $C_0 \cup S_0$ covers G, and by B 's definition $C_0 \cup C'_0 \cup S_0 \cup S'_0$ covers B. Thus,

$$\omega(V_0) + 2\omega(C_0) = \omega(C_B)$$

$$\leq \omega(C_0 \cup C'_0 \cup S_0 \cup S'_0)$$

[by optimality of C_B]

$$= 2\omega(C_0) + 2\omega(S_0).$$

Therefore, $\omega(V_0) \leq 2\omega(S_0)$.

Q.E.D.

References

[1] Bar-Yehuda, R., and S. Even, "A Linear Time Approximation Algorithm for the Weighted Vertex Cover Problem", *J. of Algorithms*, vol. 2, 1981, 198-203.

[2] Bar-Yehuda, R., and S. Even "On Approximating a Vertex Cover for Planar Graphs", *Proc. 14th Ann. ACM Symp. Th. Computing*, 1982, 303-309.

[3] Bondy, J. A., and U.S.R. Murty, *Graph Theory and Applications*, North Holland, 1976. Theorem 5.3, page 74.

[4] Chvatal, V., "A Greedy-Heuristic for the Set-Covering Problem", *Math. Operations Res.*, vol. 4, no. 3, 1979, 233-235.

[5] Chiba, S., T. Nishizeki, and N., Saito, "Applications of the Lipton and Tarjan's Planar Separator Theorem", *J Information Proc.*, 4, 1981, 203-207.

[6] Djidjev H. N. "On the Problem of Partitioning Planar graphs", *SIAM J. on Alg. and Disc. Meth.* 3, 2, 1982, 229-240.

[7] Even, S., *Graph Algorithms*. Comp. Sci. Press, 1979.

[8] Garey, M. R., and D. S. Johnson, *Computers an Intractability; A Guide to the Theory of NP-Completeness*; Freeman, 1979.

[9] Gavril, F. see [8] page 134.

[10] Galil, Z., "A new Algorithm for the Maximal Flow Problem", *Proc. 9th Ann. Symp. on Foundations of Computers Science*, 1978, 231-245.

[12] Hochbaum, D. S. "Approximation Algorithms for the Set Covering and Vertex Cover Problems, "*SIAM J. Comput.*, vol. 11, 3, 1982, 555-556.

[12] Hochbaum, D. S., "Efficient Bounds for the Stable Set, Vertex Cover and Set Packing Problems", *Dis. App. Math.* vol. 6, 1983, 243-254.

[13] Harary, F., *Graph theory*, Addision-Wesly, 1972 (revised).

[14] Hopcroft, J. E., and R. M. Karp, "An $n^{5/2}$ Algorithm for Maximum Matching in Bipartite Graphs," *SIAM J. Comput.*, 2, 1973, 225-231.

[15] Itai, A., and M. Rodeh, "Finding a Minimum Circuit in a Graph", *SIAM J. Comput.* vol. 7, 4, 1978, 413-423.

[16] Karp, R. M. "Reducibility among Combinatorial Problems", in R. E. Miller and J. W. Thatcher (eds.), *Complexity of Computer Computations*, Plenum Press, New York 85-103.

[17] Monien, B. and E. Speckenmeyer, "Some further Approximation Algorithms for the Vertex Cover Problem", to be presented in *the 8th Coll. on Trees in Algebra and Programming*. Univ. degli Studi de L'Aquila, Italy. March 1983.

[18] Nemhauser, G. L. and L. E. Trotter, Jr., "Vertex Packing: Structural Properties and Algorithms" *Mathematical Programming*, vol. 8, 1975, 232-248.

[19] Sleator, D. D. K., "An *0(mnlogn)* Algorithm for the Maximum Network Flow", Doctoral Dissertation, Dept. Computer Science, Stanford University, 1980.

[20] Wigderson, A., "Improving the Performance Guarantee for Approximate Graph Coloring", *Journal of Association for Computing Mechinery*, vol. 30, 4, 1983, 729-735.

[21] Wigderson, A., private communication.