Traversal Marker Placement Problems are NP - Complete *

Shachindra Maheshwari

CU-CS-092-76

University of Colorado at Boulder
DEPARTMENT OF COMPUTER SCIENCE

* This work was supported by NSF Grant DCR 75-09972.

ABSTRACT

This note discusses the problem of placing software monitors in programs to aid testing. It is shown that the optimal as well as the ε-approximate traversal marker placement problems are NP-complete.

Introduction.

In validating software through systematic testing evaluation of a test requires that the output be monitored in some way. Ramamoorthy et. al. [1] has proposed monitors for analysing test paths during a run. Traversal markers are an example of such montitors that use the information recorded during the test to determine the test path.

The problem considered in this note is that of optimal placement of monitors viewing the optimization process as determining the minimal number of locations needed to place monitors to identify the test path traversed. We show that the optimal placement problem for traversal markers is NP-complete. Briefly, NP-complete problems belong to that wide class of problems for which the existence of an efficient (polynomial time) solution for any problem in the class implies the existence of such solutions for all other problems in NP [2]. A problem X is NP-complete if

(i) some NP-complete problem A "reduces" to X, i.e. the existence of an efficient solution to X implies such a solution to A, and

(ii) X can be solved non-deterministically in polynomial time, i.e., $X \in NP$.

In view of the NP-completeness of the optimal placement problem, we analyze the complexity of algorithms that give approximate solutions. Following Johnson [3] we define an algorithm to be ε-approximate for a minimization problem if and only if $\left| \frac{F^* - \tilde{F}}{F^*} \right| \leq \varepsilon, \varepsilon > 0$, where F^* is the optimal solution and $\tilde{F}$ is the approximate solution obtained. We show that the corresponding approximate problem for traversal marker placement problem is also NP-complete. This implies that any polynomial time approximation algorithm for the above problem will produce arbitrarily bad outputs on some inputs.

Proofs of NP-Completeness.

We view programs as flow graphs, i.e., directed graphs with a single source and sink denoted by s and t respectively. In what follows we consider acyclic flow graphs only.

Consider the flow graph represented by the directed graph $G = (V,E)$, where V denotes the set of vertices and E a relation on V denotes the set of edges. $P(G)$ is the set of the paths from s to t, and for $p \varepsilon P(G)$, $A(p)$ represents the set of edges constituting the path p. A set of edges $E_1, E_1 \subseteq E$, defined to be a placement for traversal markers if $E_1 \cap A(p_i) \neq E_1 \cap A(p_j)$ for all $p_i, p_j \varepsilon P(G)$, i.e., each path from s to t in G covers a unique subset of edges in E_1. Clearly the set of edges in a flow graph is a placement for traversal markers. One of the properties of traversal markers is that deletion of traversal marker edges from the digraph results in a subgraph that has at most one path between any pair of vertices, i.e.,a uniconnected subgraph. Ramamoorthy et. al. [1] has shown that for acyclic flow graphs maximal uniconnected subgraph problem (i.e., determining the subset of edges of smallest cardinality deletion of which results in a uniconnected subgraph) and the optimal traversal marker placement problem are equivalent, i.e., the solution to both problems is the same.

For the problems considered here it is quite straight forward to show that they can be solved non-deterministically in polynomial time. We shall therefore give proofs of "reductions" only.

Lemma 1. The vertex cover problem can be transformed polynomially to the traversal marker placement problem.

Proof. For an undirected graph $G=(N,A)$† and a postive integer ℓ the vertex cover problem is defined as determining the existence of a set $R \subseteq N$ such that $|R| \leq \ell$ and every edge in A is incident with some vertex in R.

Consider the following transformation on the graph $G=(N,A)$:

$$
\begin{aligned}
V' &= \{s,t,u_0,u_1\} \cup A \cup N \ , \\
E' &= \{(s,u_0),(s,u_1)\} \cup \{(u_1,u) \mid u \varepsilon N\} \cup \\
&\quad \{(u,e),(v,e) \mid e \varepsilon A \text{ and } u,v \text{ are the vertices on which} \\
&\qquad e \text{ is incident}\} \cup \\
&\quad \{(u_0,e),(e,t) \mid e \varepsilon A\} \ , \\
k &= \ell + |A| \ .
\end{aligned}
$$

For the digraph $G'=(V',E')$ let $E^*, E^* \subseteq E'$, be such that $G^*=(V',E'-E^*)$ is uniconnected. Suppose $|E^*| \leq k$. Since G^* is uniconnected and in G' there are $|A|$ disjoint path[illegible] least $|A|-1$ edges emanating from u_0 or [illegible] In fact all of these $|A|-1$ edges must be incident at t in G' so as to render the subgraph subtended by $N \cup A \cup \{t\}$ uniconnected. One of the edges emanating from s will be in E^* so as to have at most one path from s to any other vertex. The only other multiple paths in G' are in the subgraph subtended by vertices $\{u_1\} \cup A \cup N$, denoted by $G''=(V'',E'')$. In G'' the edges emanating from u_1 correspond to the vertices in $G=(N,A)$ and the sinks of G'', i.e., the vertices in A, correspond to the edges of G. Also, in G'' there are exactly two paths from u_1 to every vertex in A. If the edge e in $G=(V,E)$ is incident on vertices u and v then the two paths from u_1 to e pass through edges (u_1,u) and (u_1,v) respectively.

---

† N and A denote vertices and edges of undirected graphs.

Hence removal of the edge (u_1,v) in G" leaves only one path between u_1 and those sinks in G" which correspond to the edges in G that are incident on v. The proof, therefore, follows. □

Lemma 2. The vertex cover problem can be transformed polynomially to the ε-approximate traversal marker placement problem.

Proof. Given an undirected graph G=(N,A) and an integer k, let G*=(V*,E*) be the digraph produced using the following transformation:

$$\begin{aligned} V^* = & \{s,t,u_0,u_1,u_2,\ldots,u_m\} \cup A \\ & \cup N \times \{1,2,\ldots,m\} \quad , \\ E^* = & \{(s,u_i) \mid 0 \le i \le m\} \\ & \cup \{(u_i,(v,i)) \mid v \in N \text{ and } 1 \le i \le m\} \\ & \cup \{((v,i),e) \mid e \in A, v \in N \ni e \text{ is incident on } v \text{ in } G, \\ & \qquad 1 \le i \le m\} \\ & \cup \{(u_0,e),(e,t) \mid e \in A\} \ . \end{aligned}$$

The digraph G*=(V*,E*) differs from G'=(V',E') of Lemma 1 in that it consists of m copies of G"=(V",E"), as defined in Lemma 1, each connected to the source s and having A as the common set of vertices. Since all the m copies of G" are connected in "parallel" any optimal or approximate algorithm can be trivially modified such that in the solution all the m copies are represented by the same number of edges. Therefore, by reasoning similar to Lemma 1 the optimal edge cover for the graph G=(N,A) will have cardinality k if and only if the optimal solution to the traversal marker placement problem has cardinality $|E|$+m.k. Since the closest non-optimal solution, assuming that the optimal solution has cardinality $|E|$+m.k, will have at least $|E|$+m.(k+1) edges, it follows that any approximate algorithm for the traversal

marker problem which on digraph G^* results in a solution of cardinality $<|E|+m.(k+1)$ solves, in effect, the optimal problem. Given any ε, we can always choose m such that $\left|\frac{F^*-\tilde{F}}{F^*}\right| = \left|\frac{m}{|E|+mk}\right| \leq \varepsilon$.
Hence ε-approximate solution to $G^*=(V^*,E^*)$ will exist if and only if the optimal solution to the vertex cover problem has cardinality $\leq k$. $\square$

Since vertex cover problem is NP-complete [2] we have that optimal traversal marker and the ε-approximate traversal marker problems are NP-complete.

References

[1] Ramamoorthy, C. F., Kim, K.M. and Chen, W. T., "Optimal Placement of Software Monitros Aiding Systematic Testing", IEEE Transactions on Software Engineering, SE-1, pp. 403-411.

[2] Karp, R. M., "Reducibility Among Combinatorial Problems", in Complexity of Computer Computations, R. E. Miller and J. W. Thatcher, eds., Plenum, New York, 1972, pp. 85-103.

[3] Johnson, D., "Approximation Algorithms for Combinatorial Problems", Fifth ACM Symposium on Theory of Computing, 1973, pp. 38-49.