

UDC 519.1

Heuristic Approach to the Problem of Minimal Extension of a Communication Network and Its Assessment Based on a Specialized Class of Weighted Digraphs

A. Gordonov, L. Petingi

*Computer Science Department
College of Staten Island, City University of New York
2800 Victory Boulevard, 1N-215 Staten Island, NY, USA, 10314*

In the paper we are presenting a heuristic approach to solve the problem of changing network topology by minimally extending a digraph G' through adding edges from a given spanning supergraph G of G' , such that the sum of the costs of the new edges is minimum, and in the new graph the end-to-end delay between two distinguished vertices s and t meets a predefined time constraint (ME problem). We develop a heuristic based upon the Genetic-type algorithm technique. Moreover, the application of this heuristic is justified and is shown that the solution of ME problem belongs to the NP-hard computational class.

Key words and phrases: networks, Genetic Algorithm, NP-hard, delay and cost constraints.

1. Introduction

A huge number of services offered by modern communication networks to the users very often still cannot satisfy highly changeable requirements. A network topology and natural restrictions related to it (such as delays) may prevent clients from implementing new services and introducing new network functions. It is especially true when we are talking about real-time networks. In this paper we are considering a problem of changing network topology by adding additional edges to the network graph in such a way that time constraints will be met and all added edges will have a minimum possible cost. We are also going to assess the performance of the solution by using as an example a specific class of weighted digraphs (which will be defined in 3). We chose this example because of the following two reasons. First of all the structure of the graphs in the class closely corresponds to the structure of a big class of applications – real-time networks for distributed objects (such as communication and control systems of linear gas and oil pipelines). Second – it is easy to find an analytical optimal solution that can be used to evaluate the performance of the algorithm developed in the paper.

Let $G = (V, E)$ be a weighted digraph with vertex-set V , and edge-set E . Edge-set has weight functions $T : E \rightarrow R^*$ and $C : E \rightarrow R^*$ (where R^* are non-negative real numbers).

Let s and t be two distinguished vertices of G . A path $P = (\hat{V}, \hat{E})$ is a digraph with vertex-set $\hat{V} = \{v_1, v_2, \dots, v_k\} \subseteq V$, and edge-set $\hat{E} = \{(v_1, v_2), (v_2, v_3), \dots, (v_{k-1}, v_k)\} \subseteq E$. If $s = v_1$, and $t = v_k$, then we call this path a s, t -path. Moreover, given a path P , let $\tau(P)$ and $\epsilon(P)$ represent the sum of the weights of the edges of P with regard to the weight functions T and C , respectively. Let $P_{\text{short}}(G, T)$ be the shortest path between s and t in G with respect to the weight function T . Let $G' = (V, E')$ be a spanning subgraph of $G = (V, E)$ and let λ be a bound. We formulate the following optimization problem:

$$\Theta : \text{Minimize} \quad \sum_{e \in E' \subseteq (E - E')} C(e) \quad (1)$$

Such that $\tau(P_{short}(G^*, T)) \leq \lambda$, where $G^* = (V, E' \cup E'')$.

Informally, this problem is to extend a digraph G' with edges from its supergraph G such that the sum of the costs of the new edges is minimum, and in the new graph there exists a simple path between s and t that meets the λ – constraint. For future references we are going to call this problem the Minimal Extension Graph Problem or ME problem. One of the possible applications of the ME problem is an extension of a real-time network. The problem arises, for example, when a node has to deliver real-time data to another network destination node, and the transmission must meet certain delay constraints. In the case that the delay constraints are not met, we want to add additional edges such that the total cost of these edges is minimized, and the transmission time is acceptable. This problem can be shown to be at least as hard as the Constrained Shortest Path problem (CSP), known (see [1–3]) to belong to the NP-hard computational class, as the CSP problem can be reduced to the ME problem: we therefore propose an approximation algorithm using a Genetic Algorithm approach. In Section 2 we present a Genetic-type approximation algorithm to solve the ME problem. In Section 3 we evaluate this approximation technique for a class of weighted graphs where the exact optimal solutions of the ME problem are known. Finally, in Appendix, we formally prove that the ME problem is NP-hard, hence justifying the application of an approximation technique; moreover we discuss the complexity of the algorithm described in this paper.

2. A Generic-Type Algorithm for the Solution of the ME Problem

2.1. General Description of the Algorithm

As we will show in Appendix, an exact solution of the ME problem requires, in the worst-case, exponential computational time. In this section we develop a polynomial iterative heuristic algorithm for the solution of the ME problem, using a Genetic Algorithm approach.

Given a digraph $G' = (V, E')$, with terminal vertices s and t , and supergraph $G = (V, E)$ of G' . Since we are concerned only with the costs of the added edges, we define the fitness of an edge $e = (u, v)$ of G as:

$$FITNESS(e) = \begin{pmatrix} 0 : e \in E' \\ C(e) : e \in E - E' \end{pmatrix}. \quad (2)$$

Moreover, the fitness function for a s, t -path $P = (\hat{V}, \hat{E})$ of G is defined as:

$$FITNESS(P) = \sum_{e \in \hat{E}} FITNESS(e) \quad (3)$$

Given the set of all s, t -paths of G , to find a s, t -path P of G with minimum fitness, and with $\tau(P) \leq \lambda$, is equivalent to solve the ME problem, as the digraph G^* (see optimization problem in previous section) is obtained from G' by adding the edges of P that do not belong to G' i.e., $G^* = (V, \hat{E} \cup E')$. As we follow a GA algorithmic technique, each individual in the population represents a s, t -path P of G , with $\tau(P) \leq \lambda$. Its corresponding chromosome contains the list of the edges $e \in \hat{E}$, that makes up the path. Moreover, each edge e of the path will also contain the information of its associated weight $T(e)$ and fitness $FITNESS(e)$.

For the Genetic Algorithm, the implementation follows the simple pseudo-code:

Procedure : ME Algorithm
Input : Weighted digraph $G' = (V, E')$
Corresponding supergraph $G = (V, E)$
Population Size; λ – constraint; NumberIterations;
Output : Weighted digraph G^ representing the solution*
yield by the ME algorithm.
Begin
Generation = 1;
(1) *Pop = GenerateInitialPopulation(G);*
(2) *While(Generation < NumberIterations)*
(2.1) *Pop = Selection(Pop);*
(2.2) *Pop = Reproduction(Pop);*
Generation = Generation + 1;
End While
Let $P = \text{Best Individual}(Pop)$;
Let $G^ = (V, (E(P) \cup E'))$;*
Return (G^);*
End

3. Reproduction

Reproduction is accomplished by randomly selecting pairs of a population. Given a pair of s, t -paths P_1 and P_2 of G , two new offsprings P_3 and P_4 are generated. We start with the strong assumption that the end-to-end delay between vertices s and t of both P_1 and P_2 meet the constraint λ i.e., $\tau(P_1) \leq \lambda$ and $\tau(P_2) \leq \lambda$. Moreover, the offspring of P_1 and P_2 , P_3 , will also meet this constraint (i.e., $\tau(P_3) \leq \lambda$).

The original population is created by application of the k -shortest paths algorithm with respect to the weight function T , and where $k = \text{PopulationSize}$ (for further references see for example [4–6]). If the i -path of the original population does not meet the end-to-end constraint with respect to λ (i.e., $\tau(P_i) > \lambda$), then the solution of the ME problem is obtained from one of the previous $i - 1$ paths of this population with minimum fitness, thus we can assume that the original k -shortest paths meet the λ constraint.

Let $P = (\hat{V}, \hat{E})$ be a path of G and $u, v \in \hat{V}$; the $\text{Path}(P, u, v)$ is a subpath of P with end-vertices u and v .

Let $P_1 = (V_1, E_1)$ and $P_2 = (V_2, E_2)$ be two s, t -paths of a population. We assume also that the vertex-set $V_1 = (s = v_1, v_2, v_3, \dots, t = v_n)$ is an ordered n -tuple such that $e = (v_i, v_{i+1})$ is an edge of E_1 .

Let $V_2 = (s = u_1, u_2, u_3, \dots, t = u_m)$ be also a ordered m -tuple with respect to the edges of E_2 . In addition let $W = V_1 \cap V_2 = (s = w_1, w_2, w_3, \dots, t = w_p)$, where W is an ordered p -tuple such that if $w_i = v_c$ and $w_{i+1} = v_r$ then $c < r$ (i.e., the vertices of W are ordered with respect to the order of the vertices of V_1).

Let $P' = \text{Path}(P_2, u, v)$ such that $u = w_c$ and $v = w_r$ for $c < r$, and P' does not include any other vertices from W , then we call this subpath a forward-subpath of P_2 . For example, in Fig. 1, $P'' = \text{Path}(P_2, u_1, u_3)$ is a forward-subpath of P_2 , but $P''' = \text{Path}(P_2, u_3, u_4)$ is not.

Let P_3 be the path obtained from P_1 by replacing the subpath $P' = \text{Path}(P_1, v_i, v_j)$ of P_1 by the forward-subpath $P'' = \text{Path}(P_2, v_i, v_j)$, for some nodes v_i, v_j , and suppose that $\tau(P_3) \leq \lambda$, then we called P'' a permissible-forward-subpath. For example in Fig. 1, there are two forward-subpaths, but $P'' = \text{Path}(P_2, u_1, u_3)$ is the only

permissible-forward-subpath. In Fig. 2, P_3 is obtained by replacing the subpath $P' = Path(P_1, v_1, v_3)$ in P_1 , by the permissible-forward-subpath $P'' = Path(P_2, u_1, u_3)$ of P_2 .

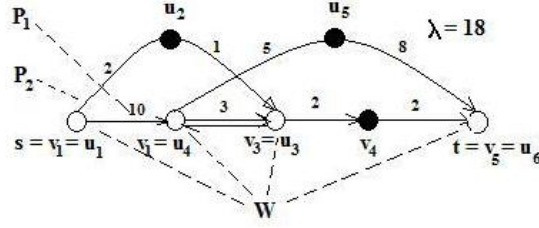


Figure 1. Two paths P_1 and P_2 and the corresponding T -weights of their edges

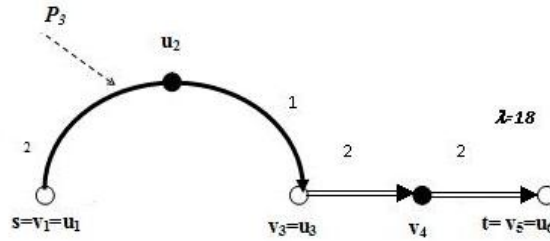


Figure 2. P_3 is the offspring of P_1 and P_2 . Moreover, $\tau(P_3) \leq \lambda$

From all the permissible-forward-subpaths (if any) of P_2 , the reproduction algorithm randomly chooses one, to generate P_3 . The second offspring, P_4 , is obtained in the same manner, but by allowing P_2 to be the first path in the pair (P_1, P_2) . Since there are $k/2$ pairs of paths in a population and each pair produces two offsprings, the reproduction procedure generates k new s, t -paths of G .

4. Selection

Given a population Pop of ks, t -paths, as we mentioned in the previous section, we assume that each path P of Pop has $\tau(P) \leq \lambda$. Recall from the Section 2.1 that the fitness of a path P is defined as in (2) and (3).

In the Selection process, we determine how many copies of a particular path will be passed to the next generation by ranking each path by an index function, involving the Fitness of a path.

The average fitness of a population Pop composed of k paths $P_1, P_2, P_3, \dots, P_k$ is defined as

$$AverageFitness(Pop) = \frac{1}{k} \left(\sum_{i=1}^k FITNESS(P_i) \right).$$

We also define the maximal fitness of a population Pop as

$$MaxFitness(Pop) = \max_{P \in Pop} (FITNESS(P)).$$

The numbers of copies of a path P that will be passed to the next generation is determined by the following index:

$$NumCopies(P) = \left\lfloor \frac{MaxFitness(Pop) - FITNESS(P)}{MaxFitness(Pop) - AverageFitness(Pop)} \right\rfloor.$$

Since the solution of the ME problem is obtained by adding to G' the edges of the path with minimum $FITNESS$ in G , the larger the difference between $MaxFitness(Pop)$ and $FITNESS(P)$, the more copies of P will be made. Also the index $umCopies$ is used to rank the paths of Pop in a non-ascending order, and the paths with higher priority will be included first. This procedure will continue until the size of the next generation is k .

5. Empirical Results

In this section we will assess the performance of the ME algorithm by comparing the solution given by the algorithm with the optimal solution analytically obtained for a specific class of weighted digraphs.

Consider the following weighted-digraph $G = (V, E)$ with node-set $V = (s = v_1, v_2, v_3, \dots, t = v_n)$, with corresponding weight functions on the edges (T and C). The edge-set E is partitioned in three subsets E_1, E_2 , and E_3 .

Let

$$\begin{aligned}
 E_1 &= \{(u, z), \text{ where } u = v_i, z = v_{(i+1)}, 1 \leq i \leq n - 1\}, \\
 E_2 &= \{(u, z), \text{ where } u = v_i, z = v_{(i+2)}, 1 \leq i \leq n - 2\}, \\
 \text{and } E_3 &= \{(u, z), \text{ where } u = v_i, z = v_{(i+3)}, 1 \leq i \leq n - 3\}.
 \end{aligned}$$

Regarding the weight function T , we have $T(e) = 1$, for $e \in E_1, T(e) = 6$, for $e \in E_2$, and $T(e) = 12$, for $e \in E_3$. With respect to the weight function C , we have $C(e) = i$, for $e \in E_1, e = (v_i, v_{(i+1)})$, and $C(e) = 0$, for $e \in E_2, \text{ or } e \in E_3$ (see Fig. 3). Let $G' = (V, E' = E_2 \cup E_3)$, that is, another way to see this problem is that we extend G' with edges from E_1 thus that the sum of the costs of the new edges from E_1 is minimized and the new graph meet the end-to-end delay constrained between s and t . Equivalently, as we mentioned previously, the original population is composed of s, t -paths of G that meet the λ -constraint, and, by the application of the ME algorithm, we will try to minimize the fitness of the paths under consideration.

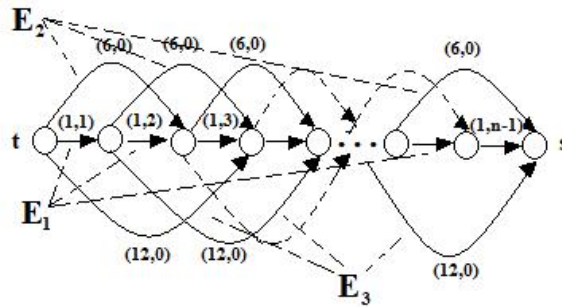


Figure 3. Class of weighted-digraphs G

For each edge, the ordered pair (a, b) represents the t -weight and cost associated with that edge. For this class of graphs, let n be the number of nodes of G . If we let $\lambda = 2n$, the analytical solution for the ME problem is represented by the path P with edge-set $E(P) = E^* \cup E^{**}$, where $E^* = \{(u, z) \text{ such as } u = v_i, z = v_{(i+1)}, 1 \leq i \leq n/2 - 1\}$, and $E^{**} = \{(u, z) \text{ such as } u = v_i, z = v_{(i+2)}, n/2 \leq i \leq n - 2\}$ (see Fig. 4). The optimal solution is a path P with $FITNESS(P) = (n - 1)(n + 1)/8$. In Table 1 we compare the optimal solution obtained analytically and the ones obtained by the ME algorithm for the case where $\lambda = 2n$, for different values of n . Moreover, we let $PopulationSize = 100$, and $NumberIterations = 100$. As Table 1 shows, the error deviation is within nine percent.

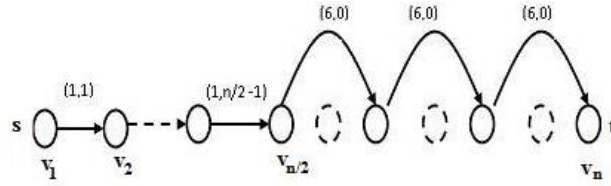


Figure 4. Optimal solution for the ME problem for the class of weighted-graphs G , and for $\lambda = 2n$

Experimental and Analytical Results

Table 1

Number of Vertices	Optimal Solution	ME Algorithm Solution	Deviation
13	21	21	0
25	78	78	0
37	171	180	2
49	300	312	4
61	465	480	4
73	666	694	4
85	903	937	4
97	1176	1231	4.5
109	1485	1527	2.8
121	1830	1874	2.4
241	7260	7863	7
361	16290	17334	6
481	28290	31254	7
601	45150	47107	4
721	64980	70988	9
841	88410	94413	6
961	115440	119215	6
1081	146070	155687	6.5

6. Appendix

First we would like to show that the minimal extension of a communication network to meet an end-to-end delay constraint problem belongs to the NP-hard computational class. As the ME problem, the Constrained Shortest Path problem (CSP) is to find a minimum cost path P of G between s and t (see (1)) such that $\tau(P) > \lambda$ (i.e., find among all s, t -paths of G that meet the time constraint λ , the one that has minimum cost). If we restrict $G' = (V, \emptyset)$ to be the subgraph of $G = (V, E)$ (i.e., G' is a spanning subgraph of G with no edges), then to solve the ME problem for an instance conformed of weighted digraphs G' and G , is equivalent to solve the CSP for the instance G . Thus the ME problem is at least as hard as the CSP problem. Since in [1] it was shown that a CSP problem is NP-hard, then by restriction the ME problem also belong to this computational class.

Let consider next the computational complexity of the heuristic presented in this paper. Given a digraph $G = (V, E)$, an algorithm was introduced in [4] to find the k -shortest path between two vertices s and t in a digraph $G = (V, E)$ in time

$O(kn(e + n \log n))$, where n number of nodes in V and e number of edges in E . Let $k = PopulationSize$ and $i = NumberIterations$. The original population Pop (step 1 of the algorithm presented in Section 2.1) for the digraph $G = (V, E)$ was created by using the k -shortest path Algorithm (see [4]) with respect to the weight function T of G . Given two s, t -paths P_1, P_2 , assuming both have lengths n , the complexity of generating their offspring P_3 is of order $O(n)$. For a population of size k , since there are $k/2$ pairs of paths and since each pair produces two offsprings, P_3 and P_4 , the reproduction (step 2.2) will take $O(kn)$. In the Selection (step 2.1) to find the *MaxFitness* of Pop , as well as to determine the *AverageFitness* of Pop will take $O(kn)$. Step 2 (*while* structure) will take $O(ikn)$, thus the complexity of the algorithm is $O(ikn + kn(e + n \log n))$, thus the complexity of the ME algorithm is the same as the one to generate the k -shortest paths of G .

References

1. *Handler G. Y., Zang I.* A Dual Algorithm for the Constrained Shortest Path Problem // *Networks*. — Vol. 10. — 1980. — Pp. 293–310.
2. *Gellermann T., Sellmann M., Wright R.* Shortest Path Constraints for the Resource Constrained Shortest Path Problem // *Lecture Notes in Computer Science*. — 2005. — Vol. 3524/2005. — Pp. 201–216.
3. *Sigurd M., Zachariasen M.* Constrained Shortest Path // *Proceeding of Algorithms-ESEA 2004, 12-th Annual European Symposium*. — Bergen, Norway: 2004. — P. 802.
4. *Eppstein D.* Finding the k -Shortest Paths // *SIAM J. Computing*. — Vol. 28(2). — 1998. — Pp. 652–673.
5. *J. Hersberger M. M., Suri S.* Finding the k — Shortest Simple Paths: A New Algorithm and its Implementation // *Proceeding of the Fifth Workshop on Algorithm Engineering and Experiments. ALENEX 2003*. — Baltimore, MD, USA: 2003. — Pp. 26–36.
6. *Lawler E. L.* A Procedure for Computing the K Best Solutions to Discrete Optimization Problems and its Application to the Shortest Path Problem // *Management Science*. — Vol. 18. — 1972. — Pp. 401–405.

УДК 519.1

Эвристический подход к проблеме минимального расширения коммуникационной сети и его оценка, основанная на использовании специального класса графов

А. Гордонов, Л. Петинжи

Кафедра вычислительной техники

*Колледж Стейтен Айленда, Нью-Йоркский городской университет
2800 Бульвар победы, Статен-Айленд, Нью-Йорк, США, 10314*

В статье представлен эвристический подход к проблеме изменения топологии сети путём минимального расширения ориентированного графа G' с помощью добавления рёбер из суперграфа G графа G' таким образом, что сумма стоимостей новых рёбер минимальна и общая задержка между двумя выделенными узлами s и t удовлетворяет заранее определённым ограничениям. Для решения этой проблемы авторами разработан алгоритм генетического типа. Более того, проведена оценка эвристического подхода с использованием специального класса ориентированных графов, и показано, что решение проблемы минимального расширения коммуникационной сети с ограничениями на задержку принадлежит к классу NP-полных вычислительных задач.