A Proposal Algorithm to Solve Delay Constraint Least Cost Optimization Problem

Waleed A. Mahmoud **Dheyaa J. Kadhim**
Electrical Engineering Department
College of Engineering / University of Baghdad
Email: deya_naw@yahoo.com

ABSTRACT

Traditionally, path selection within routing is formulated as a shortest path optimization problem. The objective function for optimization could be any one variety of parameters such as number of hops, delay, cost...etc. The problem of least cost delay constraint routing is studied in this paper since delay constraint is very common requirement of many multimedia applications and cost minimization captures the need to distribute the network. So an iterative algorithm is proposed in this paper to solve this problem. It is appeared from the results of applying this algorithm that it gave the optimal path (optimal solution) from among multiple feasible paths (feasible solutions).

Keywords: Routing, Routing Algorithms, QoS, Optimization and Feasible Problem.

الخلاصة

بشكل عام أختيار المسار خلال عملية تحديد المسار يمكن معاملتها كمشكلة أقصر مسار أمثل. أن الدالة الموضوعية للامثلية ممكن ان تكون اي متغير من المتغيرات الاتية: عدد القفزات، زمن التاخير، الكلفة... الخ. مشكلة تحديد المسار باقل كلفة بزمن تاخير محدد درست في هذا البحث لان تحديد زمن التاخير هو متطلب مهم من متطلبات تطبيقات وسائط النقل وتقليل الكلفة لتسهيل النقل وتوزيع البيانات في الشبكة. لذلك تم اقتراح خوارزمية تكرارية في هذا البحث لحل هذه المشكلة. وقد ظهر من نتائج التطبيق هذه الخوارزمية انها تعطي المسار الافضل (الحل الامثل) من بين عدة مسارات ملائمة (حلول ملائمة).

1. INTRODUCTION:

The Delay-Constrained Least-Cost (DCLC) path problem is searching for a path that has the minimum cost and a delay not exceeding a given upper bound [G.Feng2001].

The single mixed weight idea proposed in this paper can be briefly described as follows; "Given a network with a delay and a cost associated with each link, it can first obtain a single mixed weight for each link by combining its delay and cost in terms of one parameter, and then use Dijkstra's Algorithm to find the corresponding shortest path". It can be theoretically proved that as long as the parameter is appropriately chosen the obtained shortest path must be a feasible solution with a cost no greater than that of the least delay (LD) path. Based on this result, a heuristic algorithm is used that can produce good solutions by executing Dijkstra's shortest path algorithm at most three times. To further improve the quality of the solution, then two iterative algorithms are proposed that can generate a series of parameters gradually improving the corresponding solutions. A large number of numerical experiments are carried out and the results of them are compared [R.Sriram1998].

2. THE DCLC PATH PROBLEM [G.FENG]:

Any network can be represented by a directed graph G (V, E), where V is the set of nodes, and E is the set of links. Assume that N= [V], and M= [E].

A weight w define a nonnegative real number w(e) associated with each link, i.e., W: E→R_0^+. In particular, weight d: E→R_0^+ is called delay, while c: E→ R_0^+ is called cost. A path is a finite sequence of non –repeated nodes p=(v_1 ,$v_{2,\dots,}$ v_k) such that for 0≤i<k, there exists a link from v_1 to v_i $+1$, i.e.. (v_i ,v_i $+1$) ∈ E. A link e ∈ p means that p passes through link e. A weight w, like delay or cost, additive if the weight of a path p is equal to the summation of the weights of all links along that path,

w $(\rho) = \sum_{e\in p} w(e)$ (1)

In particular, the delay and cost of a path p are given by two equations below:

d $(\rho) = \sum_{e\in p} d(e)$ (2)

c $(\rho) = \sum_{e\in p} c(e)$ (3)

In general sense, the delay of a link is the average transmission time on that link, while the cost of a link may be the fee charged to transmitting a message on that link. However, delay and cost may be redefined as other metrics such as jitter, loss etc., as long as they are additive.

Now a formal definition for the DCLC (Delay Constraint Least Cost) problem using the above notation is illustrated below.

Definition 1:

Given a network G (V, E), a source s ∈ V and a destination node d ∈V, a delay and a cost for each link, and a delay constraintC_d, the delay –constrained least-cost (DCLC) path problem is to find a path p from s to d, such that .

(i) d (p) ≤ C_d,

(ii) c (p) ≤ c(q) for any path q from s to d that satisfies d (p) ≤ C_d,

(iii) There doesn't exit a path q from s to d, for which c(p)=c(q),

While d (p)> d (q).

It should be noted though that the third requirement is not included in a standard definition for the DCLC problem; it is introduced here to reflect our preference for a path resulting in better performance if there exist more than one solution for the standard problem. For convenience, a path that at least satisfies the first requirement in the above definition is called a ***feasible solution*** (or feasible path); a path that satisfies all the three requirements is called an ***optimal solution*** (or optimal path).

The following definition and notations are needed to describe the algorithms to be proposed in this chapter.

Definition 2:

Given two additive weights $w_1$and w_2, a mixed weight w= $w_1 + \alpha\, w_2$ means that for any link e,

w (e)= $w_1(e) + \alpha\, w_2(e)$ (4)

Where ∈ R_0^-. Apparently, a mixed weight of two additive weights is also additive.

Definition 3:

Given a source node s , a destination node d and an additive weight w. It defines a function (or procedure) Dijk (w) that returns the shortest path w from s to d found using Dijkstra's algorithm. In particular it is equivalent to let p_d=Dijk (d) be the least delay (LD) path, and p_c=Dijk (c) the least cost (LC) path between s and d. Note that that relations d (p_d)≤ d (p_c) and c (p_d) ≥ c (p_c) always hold.

Another function to be used in our algorithms is ModiDijk (c,d). If there exist multiple LC paths with different delays from s to t, function ModiDijk (c,d) will return the one that the minimum delay. This can be done using a modified Dijkstra's algorithm.

3. THE SINGLE MIXED WEIGHT IDEA:

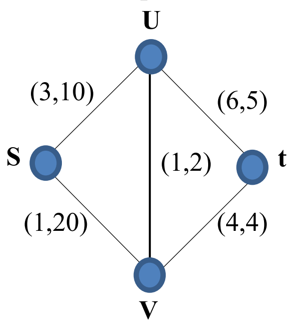
The basic idea of the algorithms proposed in the next section is to solve the DCLC problem by first combining the delay and cost into a single mixed weight, and then using Dijkstra"s algorithm to find a feasible path. In this section, the basic idea is illustrated through a simple network model.

Consider the problem defined in Fig.1 (a), in which we need to find a DCLC path from s to t with a delay bound of 8. Now, solving this problem manually required checking all four paths between s and t. even though this network model is very simple. However, it is easy to find that the LC path is s-u-t, which has a delay of 9 and is infeasible. The LD path is s-v-t, which has a delay of 5 and a cost of 24. Although the LD path is feasible, it is not the optimal solution.

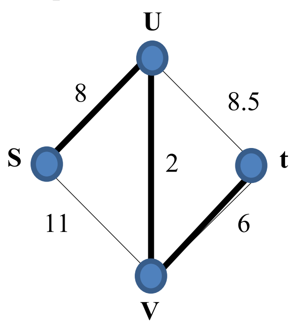
Now let us construct a mixed weight w=d+αc. If we let α=0.5.then the weight w associated with each link is shown in Fig.1 (b).by using Dijkstra"s algorithm, the shortest path from s to t, s-u-v-t, can be easily found. This path has a delay of 8 and a cost of 16, and it turns out to be the optimal path.

This example indicates that selecting an appropriate parameter to construct a mixed weight, reducing the DCLC problem to the shortest path problem, this can be readily solved using Dijkstra"s algorithm.

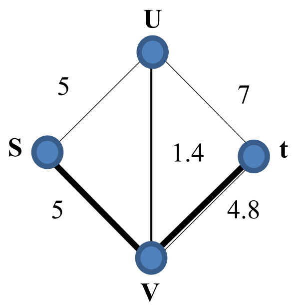
The key issue for this idea is how to choose the parameter for constructing the mixed weight. A randomly selected value for α may result in a disappointing solution. For instance, in Fig.1 (c) let α=0.2 hence the shortest path w becomes the LD path. While in Fig.1 (d), let α=2 hence the shortest path becomes the LC path.



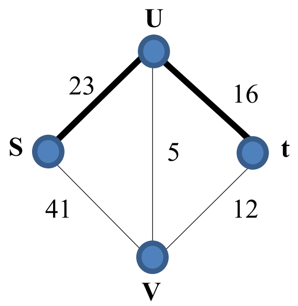
Path from s to t with delay bound=8
Figure 1 (a) A DCLC Problem



Shortest Path s-u-v-t :delay=8, cost=16
Figure 1 (b) Mixed weights when α=0.5



Shortest Path s-v-t :delay=5, cost=24
Figure 1 (c) Mixed weights when α=0.2



Shortest Path s-u-t :delay=9, cost=15
Figure 1 (d) Mixed weights when α=2

4. HEURISTIC ALGORITHMS FOR THE DCLC PROBLEM:

In the DCLC problem, the attention will be focused on finding good feasible solutions. This section will describe a basic algorithm which the other two proposed iterative algorithms based on it.

As indicated in the previous section of paramount importance in the single- mixed – weight idea is how to choose the4 parameter. the following observation shows the basic relationship between the parameter and the cost and delay of the resulting shortest path .

Theory [S.Chen and K.Nahrstedt 1998]:

If p=Dijk (d+αc) , q =Dijk (d+βc), α≥β, α€R_0^+,β€R_0^+ ,then;

c(p)≤ c(p), d(p)≥d(q).

Proof :

Since p is the shortest path weight d+ αc found by Dijkstra's algorithms, then:

d(p)+αc(p)≤ d(q)+αc(q) (5)

Similarly the following inequality holds:

d(q)+βc(q)≤ d(q)+βc(p) (6)

By combining the two inequalities, result in

αc(p)≤ d(q)+αc(q)-d(p)

= d(q)+βc(q)-βc(q)+ αc(q)-d(p)

≤ d(p)+βc(p)-βc(q)+ αc(q)-d(p)

=βc(p)+ (α- β) c(q). (7)

By moving the first term in the right – hand side to the left – hand side in equation (4.7), result in the relation c(p) ≤ c(q). Furthermore, from inequality (4.2) .it can founded that

$$d(p)\geq d(q)+\beta\{c(q)-c(p)\}\geq d(q) \qquad (8)$$

The following conclusions can be obtained from the above theory:
1) Given α € R_0^+ ,if p= Dijk (d+αc) is feasible , then for any nonnegative real number β<α , q=Dijk (d+βc) is feasible.
2) Given α € R_0^+, if p=Dijk (d+ αc) is infeasible , then for any nonnegative real number β>α, q=Dijk(d+βc) is infeasible .
3) if p =Dijk (d+αc) , where 0<α<+∞ , then

c(pd) ≥ c(p) ≥ c(pc), d(pd) ≤ d(p) ≤d(pc) .

Previous theory implies that the larger the parameter, the smaller the cost of the resulting shortest path, while the larger the delay of the path .as long as the delay constraint is not violated , a larger parameter will absolutely result in a better solution . Therefore, our remaining work is to find this goal is achieved by (Lagrange relaxation technique) that calculates lower bounds and finds good solutions for DCLC problem. This technique produces the following equation to get the value
of α that is given by:

$$\alpha \leftarrow \frac{C_d-D(p)}{C(p)-C(q)} \qquad (9)$$

The above description for solving DCLC problem is programmed here in (Visual Basic) language, the procedure of this algorithm (DCLC) is shown below:

Algorithm DCLC (G,s,t,c,d,Cd)

Step 0: Start.
Step 1: if there is more than one path with minimum cost, find the least cost path between s and d with minimum delay, and let this path named q.
q← ModiDijk(c,d)
Step 2: check the delay of the selected path in (step1) is less than or equal the delay constrained.
If (d(q)≤Cd) then
Step 3: if the result of (step) is true then the suitable path is q.
Return q
Step 4: if the result of step2 is false then the let p be the shortest path in corresponding to delay (i.e. the path that has minimum delay).
p← Dijk(d)
Step 5: check if the delay of the path p if greater than delay constrained
If (d(p)>Cd)then
Step 6: if the result of (step5) is true then there is no feasible path.
Return NULL
Step 7: if the result of step5 is false then calculate the parameter of constructing the mixed weight (α), where:

$$\alpha=\frac{delay\ constrained-delay\ of\ path(p)}{cost\ of\ path(p)-cost\ of\ path(q)}$$

$$\alpha\leftarrow\frac{Cd-d(p)}{c(p)-c(q)}$$

Step 8: Convert each path to one parameter (w), where:
w=delay+α*cost for each path
Then let p is the minimum path corresponding to (w) value.
p←Dijk(d+αc)
Step 9: p is the algorithm choice of most suitable path.
Return p
Step 10: END

5. THE PROPOSED ITERATIVE ALGORITHM:

Algorithm DCLC is very simple and fast-it only need to execute Dijkstra's algorithm at most three times.in the following section ,a proposed iterative algorithm will be described than can improve the quality of solution.
After obtaining a feasible path which is better than LD path, another parameter can be computed through the equation

$$\frac{Cd-d(pd)}{c(pd)-c(Pc}\alpha=$$

By replacing c(pd)and d(pd),respectively.as long as the new parameter is longer than the previous one, a better solution possibly can be found.
The above description for the proposed iterative algorithm named as (DCLC-A) is programmed here in (Visual Basic) language; the procedure of this algorithm is shown below:

Algorithm DCLC (G,s,t,c,d,Cd)

Step 0: Start.
Step 1: if there is more than one path with minimum cost, find the least cost path between s and d with minimum delay, and let this path named q.

q← ModiDijk(c,d)

Step 2: check the delay of the selected path in (step1) is less than or equal the delay constrained.

If (d(q)≤Cd) then

Step 3: if the result of (step) is true then the suitable path is q.

Return q

Step 4: if the result of step2 is false then the let p be the shortest path in corresponding to delay (i.e. the path that has minimum delay).

p← Dijk(d)

Step 5: check if the delay of the path p if greater than delay constrained

If (d(p)>Cd)then

Step 6: if the result of (step5) is true then there is no feasible path.

Return NULL

Step 7: Create a flag variable named continue with Boolean type having initial value= True

Continue=TURE

Step 8: Building a close loop with a condition continue=true and exit the loop just when continue=false

While continue do

Step 9: if the result of step5 is false then calculate the parameter of constructing the mixed weight (α), where:

$$\alpha = \frac{delay\ constrained - delay of\ path(p)}{cost\ of\ path(p) - cost\ of\ path(q)}$$

$$\alpha \leftarrow \frac{Cd - d(p)}{c(p) - c(q)}$$

Step 10: Convert each path to one parameter (w), where:

w=delay+α*cost for each path

Then let r is the minimum path corresponding to (w) value.

r←Dijk(d+αc)

Step 11: Check: If cost of the path(r) equal the cost of path(q) or the cost of the path P.

If (c(r) = c(q) or c(r) = c(p) then

Step 12: If the result of the condition in (step 11) is true then let continue =false to exit loop (step 15)

Continue=False

Step 13: If the result of the condition in (step 11) is false then (step 14)

Else

Step 14: Let P=the minimum path(r)

p←r

Step 15: p is the algorithm choice of most suitable path.

Return p

Step 16: END

6. DEMONSTRATION EXAMPLE:

In the following sections, demonstration examples are submitted to describe the DCLC algorithm and the proposed iterative algorithm DCLC-A and how these algorithms proceed to find a solution.

6.1 Procedure of DCLC Algorithm:

Fig. 2 shows a DCLC problem, in which it is required to find a DCLC path from node 1 to node 6 with a delay upper bound of 12. It is easy to find that the LD path is Pd=[1-3-6], while the LC path is Pc=[1-4-6]. Thus it have d(Pd)=2, c(Pd)=28, d(Pc)=18 and c(Pc)=2. Using an exact algorithm, the optimal solution to this problem is 1-4-5-6, it has a delay of 10 and a cost of 6.

When the basic algorithm DCLC is used to find a solution, it first finds the LC path and plus it in q. Since the LC path is infeasible, it continues to find the LD path and puts it in P. Since the LD path is feasible, it proceeds to compute the parameter α from equation (9), hence α=5/13. This obtains a mixed weight for each link, as shown in Fig. 3. The shortest path from node 1 to node 6, which is also the final solution of algorithm DCLC. Hence the path is 1-3-5-6. This solution, which has a delay of 3 and a cost of 16, is not optimal; however it is better than the LD path.

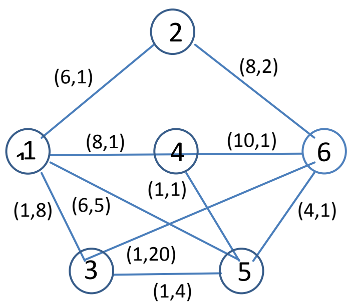


Figure 2: Optimal path from node 1 to node 6 with delay bound=12

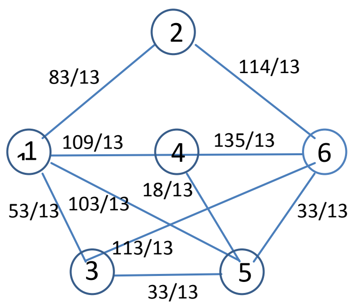


Figure 3: Shortest Path 1-3-5-6 found by DCLC and the same it can obtain by DCLC-A; after 1st iteration

6.2 Procedure of DCLC-A Algorithm:

The first iterative algorithm DCLC-A is based on algorithm DCLC. After finding the LC path q and the LD path p, the algorithm enters the iterative procedure.

In each iteration, a parameter α is first computed by equation (10) for constructing the mixed weight, and then the corresponding shortest path r is found. If r has a lower cost than p, then p is replaced by r. in the first iteration, the parameter is the same as the one used in DCLC, and correspondingly path r=[1-3-5-6] is returned. Since r is better than p, p is replaced by r. in the second iteration, the parameter is reevaluated by $\alpha=(12-3)/(16-2)=9/14$.

The corresponding mixed weight is shown in Fig. 4(a), and the shortest path r=[1-5-6] is returned. Again, p is replaced by r. in the third iteration, the parameter becomes $\alpha=(12/7)/(9-2)=(5/7)$, and the corresponding mixed weight is shown in Fig. 4(b).

Since the retuned shortest path is the same as the one obtained in the previous iteration, this algorithm stops. The final solution is 1-5-6, which has a delay of 7 and a cost of 9. It is better than the solution of algorithm DCLC, but it is still not optimal.

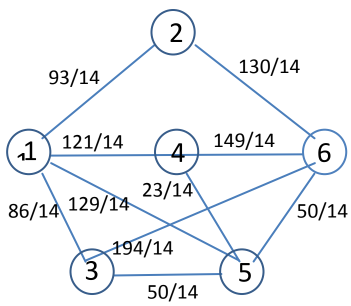


Figure 4(a): Shortest Path 1-5-6: Delay=7: Cost=9
Solution obtained after 2nd iteration

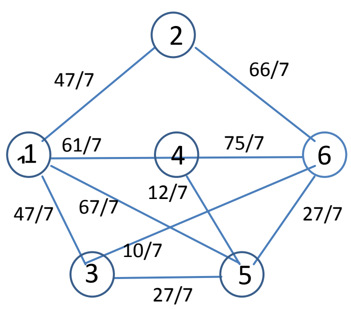


Figure 4(b): Shortest Path 1-5-6: Delay=7: Cost=9
Solution obtained after 3rd iteration

7. CONCLUSIONS:

The path selection algorithm of a routing protocol in high-speed networks must be very responsive in order to achieve a low set-up time. It must also be capable of finding solutions of high quality to ensure the most efficient utilization of network resources.

A single-mixed weight idea was proposed to solve QoS unicast routing problems was successfully used to develop heuristic algorithms for the DCLC problems. The proposed heuristics algorithm was demonstrated to be able to find solutions of high quality yet practically have very low time complexities.

8. REFERENCES:

G.Feng, "Neural network and algorithmic methods for solving routing problems in high speed network", PHD Thesis, University of Miami, Department of Electronics and Communications, 2001.

R.Sriram, "Delay Constrainted Routing in Connection-Oriented Network", Project Report, Department of Computer Science and Engineering, Indian Institute of Technology, Madras, May 1998.

S.Chen and K.Nahrstedt, "On Finding Multi-Constrained Paths", ICC'98, pp. 874-879, Atlanta, Georgia, June 1998.