ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

Computing List of Ordered Pairs from Disjoint Closed Interval to compute a most Probable Delay Path:NP-Hard in polynomial time

Sujit P Nale [1], **S.P Sonavane** [2]

Dept. of Computer Science, Walchand College of Engineering, Sangli, India[1,2]

ABSTRACT*:* Computing most probable delay of a given graph having node(n) and edges (m) is NP-Hard problem. The hardness of the problem doesn't impact on the restricted version problem. The main idea is to solve the problem with the help of applying some constraint on given NP-Hard problem. Two different cases are considered to solve the given NP-Hard problem. First case considers as if the constraint satisfied with the problem, then it is easy to solve given NP-Hard problem in polynomial time with the help of approximation algorithm. Second case, otherwise the computational complexity of the problem remains non-polynomial. Normally, the network operator needs to find a source to destination path such that the bandwidth of the path is no smaller than the specified bandwidth requirementand delay of the path is no longer than specified the delay bound.To solve the above problem in polynomial time it is necessary to find a pair of tight lower and upper bound which is in the form of disjoint close interval. In this paper, an algorithm is designed to calculate a set of ordered pairs which is a part of implementing polynomial time algorithm to compute the most probable delay between source to destination for a given graph.

Keywords:Disjoint close interval, order pair, probable delay, link weight.

I. Introduction

The Constrained Shortest Path problem (CSP) is to identify a minimum cost path from a source node to a destination node whose delay is within specific bound. Communication network is modelled as directed graph with delay parameter associated with each link, We consider the problem of determining most probable delay constrained path [1] from a source node to a destination node.

In this paper,Delay of the link is a random variable with known mean and known variance as well as continuous and differentiable probability density function. Using central limit theorem this problem can be formulated as a path problem[2] which involves simultaneously optimizing two additive path parameters by applying constraints on delay bound. It can be considered that the given problem is restricted version of NP-Hard problem which would be treated as NP-Complete problem and has a polynomial solution.

Basically, there are two cases arise when there is one path with a mean delay less than delay bound, An exact pseudo polynomial algorithm [1] works. In another case where the path delay meanis greater than the delay bound problem which can be solved in a simple approximation algorithm. Network multimedia applications are becoming increasingly popular and are making a good case for deployment of quality of service (QoS) [3] based Network architecture. One of the key issues in such architecture is how to determine appropriate path that fulfill QoS requirement [4].It is used in applications for video delivery over bandwidth limited networks. Assume that each link in the network has a bandwidth and a delay for a given source-destination pair.According to bandwidth requirement, It is necessary to find a set of source to destination path such that the delay of the longest path is minimized while the aggregated bandwidth from the set of path meets the bandwidth requirement.

II. Related work

The Most probable delay constrained path problem (MPDCP) is related to the traditional multiconstrained QoS routing problem which is known to be NP-Hard. In Paper [1] they made mild assumption on probability distribution of link delay which leads to formulation that requires determining an optimal path with respect to mean path delay and path variance. They considered two cases, one of which requires minimization of both mean path delay and path variance and other require minimization of mean path delay and maximization of path variance.

Boston university representative Internet Topology gEnerator(BRITE) [6] is one of the topology generator tool which generates topology in order to provide link weight input as a random variable. Version 1.0 of BRITE requires a Java 2 JDK 1.3.

Table 1:BRITE Output format: Nodes Section

Rule	Field	Meaning
1	*Node Id*	*Unique id for each node*
	Xpos	*x-axis coordinate in the plane*
3	*Ypos*	*y-axis coordinate in the plane*
4	*Indegree*	*Indegree of the node*
5	*Outdegree*	*Outdegreeof the node*

Table 1 [7] has information about each node in graph. It consist unique id, x and y coordinate in plane total indegree, outdegree determines exact position of node in plane used in simulation.

Table 2: BRITE Output Format: Edges Section

Rule	Field	Meaning
1	*EdgeId*	*Unique id for each edge*
2	*From*	*node id of source*
3	*To*	*node id of destination*
4	*Length*	*Euclidean length*
5	*Delay*	*propagation delay*
6	*Bandwidth*	*bandwidth (assigned by AssignBWmethod)*
7	*ASfrom*	*if hierarchical topology, AS id of source node*
8	*ASto*	*if hierarchical topology, AS id of destination node*
9	*Type*	*Type assigned to the edge by classification routine*

Table 2 [7] has information about edges. Each edge has unique id as defined in node section, information about two nodes which contaions that common edge, length of edge, link delay and bandwidth assign to network.

III. PROPOSED ALGORITHM

A. *Phasessof Algorithm:*

The algorithm has three phases (fig 1). In the first phase (Line 1) sort the given disjoint interval in non-decreasing order. In the second phase (Lines 2-8) check whether intervals are disjoint or not. If not made it disjoint by the union of these two intervals. In the third phase(Lines 9-26) complete set of ordering pair $[\delta_k, \Delta_k]$ Such that K≤mlog$(n-1)$.

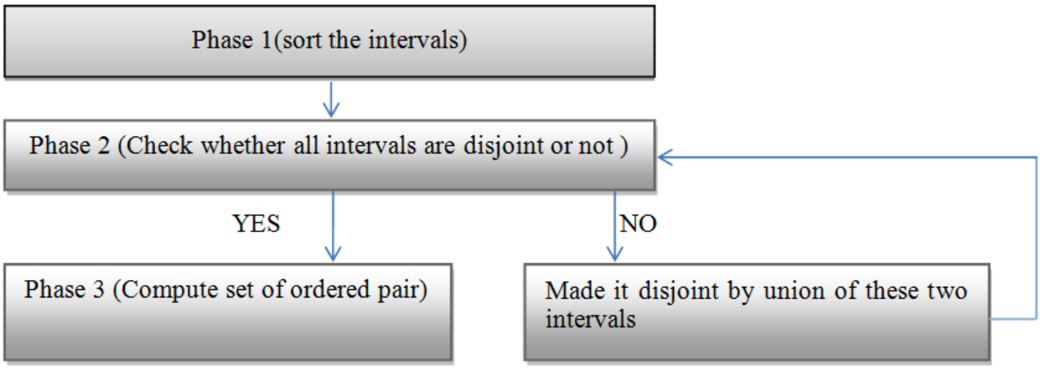
Phase 1(sort the intervals)
Phase 2 (Check whether all intervals are disjoint or not)
YES
NO
Phase 3 (Compute set of ordered pair)
Made it disjoint by union of these two intervals

Figure1:Block diagram to calculate a list of ordered pairs

B. *TERMS USED*

a) Approximation path:-a probable path whose link delay is maximum.
b)Disjoint interval:-set of ordered pair that contains lower bound and upper bound.
c) Most probable delay path:-gives maximum delay within the link from source to destination.

C. *PROCESS*

- Simulate this network using ‘BRITE’ topology.
- For all *intervals* Check whether *the interval* is disjointor not.
- Go to the next interval.
- Continue this process until all *Disjoint interval* Covers.

D.*Description of the Proposed Algorithm:*

Let G be the graph with node n=7 and m=13 according to an algorithm, various observation are listed bellow.
maximum set of closed interval J≤m*log (n-1)
i.e13log(7-1) = 34.
Consider the disjoint intervals are$[\alpha_1,\beta_1]$,$[\alpha_2,\beta_2]$,$[\alpha_3,\beta_3]$,$[\alpha_4,\beta_4]$, $[\alpha_5,\beta_5]$, $[\alpha_6,\beta_6]$.
[1,3], [5,8], [9,16], [18,20], [22,28], [30,32].
On this sorted input proposed algorithm finds set of ordered pair as follows.

According to (algorithm 1) in input $\log(\frac{3}{1}) + \log(\frac{8}{5}) + \log\left(\frac{16}{9}\right) + \log(\frac{20}{18}) + \log(\frac{28}{22}) + \log(\frac{32}{30}) \leq 34$.
i.e*4.00≤34.*
Input is valid.
Otherwise if intervals like $\log\frac{d}{a} \leq \log\frac{b}{a} + \log\frac{d}{c}$.
fore.g if [a,b] [c,d] =[9,16] [14,20] then
$\log\frac{20}{9}$ =*1.16*, $\log\frac{16}{9}$ =*0.84*, $\log\frac{20}{14}$ = *0.52*
1.16≤ (0.84+0.52)
i.e*1.16≤1.36*
The condition is true. Since intervals are joint, In order to make them disjoint replace [a,b] [c,d] as [a,d].

Make sure all the intervals are disjoint then,

1 Condition for list of ordered pairs (δ_k, Δ_k) Where k:=1, δ_1 :=1, Δ_1 :=2*δ_1=2.
2 Since 2Є {[1,3), [5,8), [9,16), [18,20), [22,28), [32,30)}
3 Set δ_2 := Δ_1:=2 and Δ_2 :=2*δ_2=4.
4 Since 4 does not Є {[5,8), [9,16), [18,20), [22,28), [32,30)}

5 setδ_3 :=5, Δ_3 :=2*δ_3=10.
6 since 10 € {[9,16), [18,20), [22,28), [32,30)}
7 set $\delta_4 := \Delta_3 := 10$
8 Δ_4 :=2*δ_4 :=20.
9 Since 20 does not € {[18,20),[22,28), [32,30)}
10 set$\delta_5 := 22$ Δ_5 :=2*δ_5 := 44.
11 Since 44 > 34 step process.
12 list of ordered pairs are *{(1,2),(2,4),(5,10),(10,20),(22,44)}*.

IV. Pseudo Code

This section presents the designed algorithm which works on how to find out list of ordered pairs by giving a set of disjoint closed interval.

Input:-

a) Graph G with ‘n’ node and ‘m’ edges having link weight μ and σ^2
b) A set of close interval [α_j ,β_j] Such as J<=m.
c) $\sum_{1\leq j\leq J} \log(\frac{\beta_j}{\alpha_j})$ ≤mlog (n-1).

```
Step 1: Sort the interval 0<$\alpha_1 \leq \beta_1 < \alpha_2 \leq \beta_2 <$...........<$\alpha_J \leq \beta_J$.
Step 2:for each j=1 to J do
    if(log$\frac{d}{a}$) ≤ log$\frac{b}{a}$ + log$\frac{d}{c}$then
    return [$\alpha_j, \beta_{j+1}$];
    else
return [$\alpha_j, \beta_j$];
end if
end for
Step 3: Each disjoint interval [$\alpha_1,\beta_1$],[$\alpha_2,\beta_2$],...........,[$\alpha_j,\beta_j$] Compute a list of ordered pairs such as ($\delta_k, \Delta_k$)
        Such as K≤mlog($n-1$).
Step 4: Set k:=1;
$\delta_k := \alpha_k$;
    $\Delta_k$ :=2*$\delta_k$;
while($\Delta_k$<$\beta_j$) do
            if$\Delta_k$€[$\alpha_j$ ,$\beta_j$ ) then
                    $\delta_{k+1} := \Delta_k$;
                $\Delta_{k+1}$ :=2*$\delta_{k+1}$;
            k:=k+1;
else
            Set  $\delta_{k+1}$ :=smallest $\alpha_j$>$\Delta_k$
        $\Delta_{k+1}$ :=2*$\delta_{k+1}$;
    k:=k+1;
    end if
    return [$\delta_{k+1}, \Delta_{k+1}$];
    end while
                    k:=K;
                    return [$\delta_k, \Delta_k$];
```

V. Simulation Results

Simulation study is carried out using Graph visualization software (Graphviz). Computing most probable delay path between giving graph with certain node and edges.Graphviz layout takes the description of graph in a simple text language and generates a diagrams in an useful format. To run the code in graphviz require special type of extension called as .dot run in linux environment.

Dot accepts input in the DOT language. The main (outermost) graph can be directed (digraph) or undirected graph. dot makes layouts of directed graphs as well as undirected graph. (A separate layout utility named as neato [7] draws undirected graphs) In the main graph, a subgraph defines a subset of nodes and edges. A node is created when its name first appears in the file. An edge is created when nodes are joined by the edge operator '→'. Running dot on this file following is the command.

a. $ dot -Tps graph1.dot -o graph1.ps (//dot Tpdf extension filename.dot -o outputfile.pdf extension)
b. $ dot -Tjpeg graph1.dot -o graph1.jpeg (//dot Tjpegentension filename.dot –o outputfile.jpeg extension)
c. $ dot -Tgif graph1.dot -o graph1.gif (//dot Tgraphics interchange format filename.dot –o outputfile.gif extension)

To implement fully polynomial time approximation theorem [1]; requires knowledge of pair of lower and upper bound. For that purpose, following algorithm suggests to compute a set of ordering pair that contains a desired pair (δ,Δ). Let d (i, j) be the random variable associated with the delay of the link (I, j) for path p then d(p)=$\sum_{(i,j\epsilon p)} d(i,j)$ And $\prod_D(p) = pr[d(p) \leq D]$ Where D = Maximum delay requirement. The most probable delay constrained path problem, ask for a path port such that $\prod_D$ p_{opt}>=I_D(p).

The lab experiments were carried out on the linux platform. Table 3 consists of a set of links (m), nodes (n), disjoint closed interval and set of ordered pairs. These set of ordered pair contains one pair that which is desired pair(δ, Δ) and helps to compute most probable delay in polynomial time.

Table 3: Sample result.

No of edges	No of nodes	Disjoint Closed Interval	Set of Ordered Pair
9	5	[1,2] [3,8] [10,13]	{(1,2),(3,6)(6,12),(12,24)}
11	6	[3,10] [12,25]	{(3,6),(6,12),(12,24),(24,48)}
15	8	[2,6] [7,15] [18,29] [31,40]	{(2,4),(4,8),(8,16),(18,36), (36,72)}
17	9	[5,10][12,18] [20,23][26,34] [35,39][42,50]	{(5,10),(12,24),(26,52)}
19	10	[2,7][9,16] [18,22][25,33] [35,44][46,53] [54,57]	{(2,4),(4,8),(9,18), (18,36),(37,74)}

VI. Conclusion and Future Work

In this paper, We discuss how to find set of ordered pairs from the disjoint close interval. More importantly, The cardinality of this set is at most mlog(n). One of the pair contains tight upper and lower bound which is in a polynomial size. The pair helps us to compute the most probable delay constrained path between given source to destination in the given graph in polynomial time. This work helps us to find most probable delay between given source to destination in polynomial time. This work is helpful to network operator who wants to know about probable factor of bandwidth. The path which belongs to probable maximum delay helps to set amount of delay bound within network. The problem is NP-Hard, however applying constraint on that problem it can solve in polynomial time which reduces computational complexity.

References.

1. Ying Xiao, KrishnaiyanThulasiraman, Xi Fang, Dejun Yang, and GuoliangXue, "Computing a Most Probable Delay Constrained Path: NP-Hardness and Approximation Schemes," IEEE TRANSACTIONS ON COMPUTERS, VOL. 61, NO. 5 MAY 2012.

2. GuoliangXue,Weiyi Zhang, Jian Tang, and KrishnaiyanThulasiraman, "Polynomial Time Approximation Algorithms for Multi-Constrained QoS Routing", IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 16, NO. 3, JUNE 2008.

3. TurgayKorkmaz, and Marwan Krunz, "Bandwidth-Delay Constrained Path Selection Under Inaccurate State Information", IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 11, NO. 3, JUNE 2003.

4. Ying Xiao, KrishnaiyanThulasiraman, "Approximation and Heuristic Algorithms for Delay Constrained Path Selection under Inaccurate State Information", Proceedings of the First International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QSHINE'04)0-7695-2233-5/2004 IEEE. (Accessed on : Aug 2013).

5. Alberto Medina, AnukoolLakhina, Ibrahim Matta, John Byers, “BRITE: An Approach to Universal Topology Generation”, BRITE is funded in part by NSF grants CAREER ANI-0096045,CAREER ANI-0093296 and ANI-9986397. (Accessed on : Aug 2013).

6. Emden R.Gansner and EleftheriosKoutsofios and Stephen North, "Drawing graphs with dot".(Accessed on :November 2, 2010).

7. Alberto Medina, AnukoolLakhina, Ibrahim Matta, John Byers,” BRITE: Universal Topology Generation from a User’s Perspective”, BRITE is in part funded by NSF grants CAREER ANI-0096045 and ANI-9986397.(Accessed on :April 12, 2001).

BIOGRAPHY

Sujit P. Nale is pursuing his Mtech in Computer Science and engineering. His current research interest includes Theory of Computation.

Dr.ShefaliSonavane has completed her M.E Ph.D in Computer Science and engineering and is working as Assistant Professor at department of CSE at WCE Sangli.(MH-India) She is associated with few research projects and has published a good number of papers at National and International leval. Her current research interest includes Theory of Computation and graph theory.