Path-Aware Multipath NUM

Duan Jingpu U3002034

***Abstract*—The multipath rate control problem, also known as the multipath utility maximization problem, has been extensively studied in previous literatures. It can be formulated as an optimization problem and solved using decomposition method in a decentralized fashion. However, a major drawback of this model is that paths used by each source are statically allocated as a known priori. The emerging software defined network increases the flexibility of the network. The paths used by each flow in the SDN network could be dynamically allocated and adjusted by a centralized controller. This new networking paradigm requires a new multipath rate control model, which captures both rate allocation and path selection process. In this paper, we propose such a new model and give an initial solution to this model. We also show the applicability of this model in practical situations.**

I. Introduction

A. Network Utility Maximization

The network utility maximization problem has been extensively studied in literatures. It can be used to mathematically model the network rate control process[1]. The basic single source NUM model tries to maximize the total utility $\sum_{s \in S} U(x_s)$ summed over all the transmitting sources in the network, without violating the capacity constraint on each link $\sum_{s \in S} R_{s,l} x_s \leq c_l$. This model could be solved in a decentralized fashion using dual decomposition, where each source update its sending rate according to the path price in order to maximize the over all utility.

The single path NUM model can be extended to the multiple paths case, in which each source can use multiple paths. However, most of the literatures[2][3] that study the multipath NUM problem assume that the routing matrix is statically allocated and each source must use all the paths determined by the routing matrix. However, today's network is usually provisioned with redundant paths in order to provide network performance guarantee. Taking the datacenter network with a k-ary fat-tree topology[4] as an example, there are $(k/2)^2$ shortest paths between any two different hosts on different pods. With such a network topology, it is impossible to use all the available paths when solving the multipath NUM problem. Usually the paths used by each source are predetermined by the network administrator which consist only a small subset of all the available paths. It is preferable to calculate a global optimum when each source could dynamically choose usable paths from all the available paths in the multipath NUM problem. This is, however, hard to implement in the traditional network as routing information is static and can not be dynamically modified.

B. Multipath TCP

Previously, the study on multipath NUM model is based on a pure theoretical approach. But the recent development of the multipath TCP linux kernel stack[5] gives chances to turn the theoretical multipath NUM model into reality. While using multiple paths for a single TCP connection can exploit the path diversity and increases the total throughput, the packets that traverse different paths will arrive in the receiver out of order, causing increased CPU overhead to re-order the received packets in the receiver. This performance tradeoff limits the number of the paths that a single multipath TCP connection can use. The current implementation set this number to be 8.

C. New Trend in Computer Network

A new trend in recent network research is software defined networking[6]. Software defined network separates the data plane from the control plane. It uses a centralized controller to control different network devices and the controller can make flexible routing and scheduling decisions based on the global network information. Compared with traditional network, software defined network is a dynamic network where network configuration information could be dynamically applied to the network devices.

With software defined network, the traditional NUM problem also needs updating. Just as mentioned above, the routing matrix in the NUM problem is statically allocated. When there are redundant paths in the network, network administrator may randomly allocate a subset of paths to the routing matrix. Then Each source have to use all the paths indicated by the routing matrix to obtain a maximized utility, but the obtained utility may not be a global maximum if each source could uses other paths which are not present in the routing matrix.

Let's consider a simple example. Suppose we have a simple network with one source s and one destination d. There are three links l_1, l_2 and l_3 connecting s and d with one unit bandwidth. There're three flows f_1, f_2 and f_3 sending from s to d. If all three flows are transmitted on the same link, then each flow f_i obtains $1/3$ unit of bandwidth, corresponding to an overall utility of $3 \times U(1/3)$. But if each flow uses a different link, then each flow f_i obtains exactly one unit of bandwidth, corresponding to an aggregate utility of $3 \times U(1)$, which is larger than $3 \times U(1/3)$. We can see from this simple example that choosing different paths for each sending source will affect the overall obtained utility. In the traditional NUM model, as the paths used by each source is fixed, the obtained utility is only a maximizer to the current set of paths. Using different set of paths may result in a larger total utility.

In traditional networks, as routing information is statically allocated by the network administrator or by some static routing protocol. Each source can only use fixed paths. But in the software defined network, paths used by each source can be dynamically determined when the source starts sending data and can be dynamically updated during the data transmission process. This flexibility in routing makes dynamically choosing the set of paths used by each source feasible, thus shed new light on how to update the traditional NUM model so that it can obtain a global maximum under different paths configuration.

D. A New Multipath NUM Approach

Based on the above analysis, we want to extend the existing multipath NUM model so that this model could fit into the new software defined networking trend. The original model only deals with the rate allocation to different sources when giving fixed paths. In this new model, we'd like to explore the global optimum rate allocation when paths that the source uses are bounded and adjustable. We give our detailed model in the following sections.

II. Related Work

First, let's discuss the existing works on the multipath NUM problem and see its limitations when being applied to the current path-redundant and flexible software-defined networks.

In [3] and [2], the multipath NUM problem is studied and different distributed algorithms are derived. The basic idea is to use the decomposition method to transform the problem into source algorithm and link algorithm, such that sources and links in the network can coordinately work with each other to achieve the maximized utility given the routing path. It is mathematically modeled as an optimization problem:

$$\max \sum_{i=1}^{I} U_i(\sum_{j=1}^{J(i)} x_{ij})$$

$$\text{subject to: } \sum_{i=1}^{I}\sum_{j=1}^{J(i)} R_{ij}^l x_{ij} \le C^l, \forall l = 1, \ldots, L.$$

In this model, there are I sources. Source i has $J(i)$ different paths towards its destination. x_{ij} denotes the rate of source i on its jth path. Let $y_i = \sum_{j=1}^{J(i)} x_{ij}$ be the total source rate for source i. It is a hidden variable not explicitly expressed in the model. Function U_i is the utility function associated with each source i which is a non-decreasing concave function. By choosing the utility function used in the model, fairness among different competing sources can be guaranteed while satisfying the throughput requirement. The biggest problem when solving the multipath NUM model is that even though the objective function **??** is strictly concave for y_i, it is not concave for x_{ij}. Even though we can find the optimal total source rate y_i^*, the optimal rate x_{ij}^* on each specific path is not unique.

A dual based decomposition algorithm could be used to derive the solution to the multipath NUM problem. By introducing the Lagrangian multiplier $q^l, \forall l = 1, \ldots, L$, we can derive the Lagrangian function:

$$\begin{aligned} L(x,q) &= \sum_{i=1}^{I} U_i(\sum_{j=1}^{J(i)} x_{ij}) + \sum_{l=1}^{L} q^l (C^l - \sum_{i=1}^{I}\sum_{j=1}^{J(i)} R_{ij}^l x_{ij}) \\ &= \sum_{i=1}^{I} [U_i(\sum_{j=1}^{J(i)} x_{ij}) - \sum_{j=1}^{J(i)} x_{ij} q_{ij}] + \sum_{l=1}^{L} C^l q^l \\ &\text{where } q_{ij} = \sum_{l=1}^{L} R_{ij}^l q^l \end{aligned}$$

The dual problem of the multipath NUM problem is:

$$\begin{aligned} D(q) &= \max_x L(x,q) \\ &= \sum_{i=1}^{I} \max_x [U_i(\sum_{j=1}^{J(i)} x_{ij}) - \sum_{j=1}^{J(i)} x_{ij} q_{ij}] + \sum_{l=1}^{L} C^l q^l \end{aligned}$$

Since the original problem is not strictly concave, the dual problem $D(q)$ is not differentiable at every feasible point. A subgradient method can be used to iteratively compute the optimal solution to the dual problem. The subgradient of $D(q)$ is given by:

$$\partial D = C^l - \sum_{i=1}^{I}\sum_{j=1}^{J(I)} R_{ij}^l x_{ij}^*$$

where x_{ij}^* is the solution to the problem:

$$B_i(q) = \max_{x_i} [U_i(\sum_{j=1}^{J(i)} x_{ij}) - \sum_{j=1}^{J(i)} x_{ij} q_{ij}] \tag{1}$$

and the link price p^l can be updated using the following formula:

$$q^l(t+1) = [q^l(t) - \alpha^l (C^l - \sum_{i=1}^{I}\sum_{j=1}^{J(I)} R_{ij}^l x_{ij}^*)]^+$$

The key problem in solving the multipath NUM problem lies in solving the source problem 1. According to the KKT condition, when optimality for the source problem 1 is reached, only the path with the smallest path price will carry a positive flow. This means that $x_{ij}^* > 0$ if $q_i j = \min_{\forall j \in J(i)} q_{ij}$ and $x_{ij} = 0$ otherwise. And the optimal total source rate is given by:

$$U_i^{'}(\sum_{j=1}^{J(i)} x_{ij}) - \min_{\forall j \in J(i)} q_{ij} = 0$$

Different papers use different method to solve this problem. In [3], a subgradient based method is proposed that increases the path rate on one of the least priced path while decrease the path rate on all the other paths. In [7], a iterative gradient method is proposed but needs a long convergence time. In [2], the objective function of the original multipath NUM problem is modified by decreasing a quadratic term. This transform the objective function into a strict convex function and then the optimal path rate for each source can be derived using a proximal optimization algorithm.

We can see from the above analysis that all the existing works that try to solve the multipath NUM problem focus on the fact that the multipath routing matrix R_{ij}^l is given and fixed. This means that each source i will be assigned a fixed $J(i)$ paths and will uses all the $J(i)$ paths during the intermediate or final stage when solving the NUM problem. This scenario fits perfectly into the traditional MPLS multipath network, which has a topology with limited number of paths between each node pair. As the network topology becomes more and more complicated, this scenario is no longer applicable. Consider the typical datacenter network where thousands of shortest paths may exist between two hosts, it is not realistic to spread traffic on that many paths considering that the current MPTCP kernel implementation only supports a maximized 8 subflows for a single connection. Current feasible solution is to randomly choose usable but fixed paths for the source. As the sampled paths may vary for each sampling, the result of the multipath NUM problem may be different. This means that the result of the multipath NUM problem with sampled paths may not be the global optimal one, considering the possibility of an increased over all utility if data is transmitted on the unused paths. This calls for the development of a new path-aware multipath NUM model, that can incorporates practical considerations and the latest networking trends. We will present our new model in the next section.

III. System Model

In this section, we present the initial result of our new model and analyze some basic properties of this model. We call our newly proposed model path-aware multipath NUM model. Our model has two basic assumptions:

Fist, we assume that a global controller knows the exact topology of the whole network. It knows all the paths for each node pair and can assign a subset of the paths as the usable paths to each node pair. This is called the software-defined-network assumption.

Second, we assume that the usable paths for each source is bounded by a constant M. M is the maximized number of paths that each source can use to transmit its data to the destination. It should be reasonably large, *e.g.* 8 in practice. This is called the practical-multipath-TCP assumption.

Based on these two assumptions, our newly proposed multipath NUM model is described by the following optimization problem:

$$\max_{x_{ij} \geq 0} \sum_{i=1}^{I} U_i(\sum_{j=1}^{J(i)} x_{ij} u_{ij}) \tag{2}$$

$$\text{subject to: } \sum_{i=1}^{I} \sum_{j=1}^{J(i)} R_{ij}^l x_{ij} u_{ij} \leq C^l, \forall l = 1, \ldots, L. \tag{3}$$

$$1 \leq \sum_{j=1}^{J(i)} u_{ij} \leq M, \forall i = 1, \ldots, I. \tag{4}$$

$$\forall u_{ij} \in (0, 1) \tag{5}$$

This model differs from the original NUM problem in the following two aspects. *First*, in this model, $J(i)$ denotes all the available paths in the network topology that controller allows the source to transmit data. *Second*, a binary decision variable u_{ij} is introduced for each x_{ij}. If $u_{ij} = 0$, then source i will not use its j-th path to transmit data. And the total number of the paths used by a source will not exceed a constant number M.

This model, however, is extremely hard to solve because it is a non-convex mixed integer non-linear programming(MINLP). Most of the literatures[8] solve this problem through a deterministic branch-and-bound method, which is a pure analytical approach. But this theoretical approach is computationally inefficient and can not incur any engineering insight as the dual decomposition method of the tranditional multipath NUM problem incurs on the design of the transport layer protocol. What we need is an computationally efficient approximation solution that can be used to guide both the design of transport layer protocol and the design of flow routing scheme for the centralized controller. The solution should iteratively adjust the paths used by each source through the centralized controller and wait for the source transport layer algorithm to converge. We will leave the detailed discussion of this approximation solution in our following research work. In this paper, we just give an algorithm for solving the relaxed version of this problem.

IV. Algorithm for Solving the Relaxed Problem

In this section we present an algorithm for solving the relaxed version of the proposed path-aware multipath NUM model. If we directly relax u_{ij} in the original problem, then even though the optimal objective value could be derived, but the maximizers x_{ij}^* and u_{ij}^* are not unique because they are coupled together as $x_{ij} u_{ij}$ and x_{ij} is not bounded in the original model.

In order to derive a unique maximizer, we make some modification to the original model. We relax the utility function U_i, add a new set of constraint for the source rate summed over all the available paths and bound x_{ij}. After relaxation and modification to the original model, the model becomes:

$$\max \sum_{i=1}^{I} \sum_{j=1}^{J(i)} x_{ij} u_{ij} \tag{6}$$

$$\text{subject to: } \sum_{i=1}^{I}\sum_{j=1}^{J(i)} R_{ij}^l x_{ij} u_{ij} \le C^l, \forall l = 1, \dots, L. \tag{7}$$

$$1 \le \sum_{j=1}^{J(i)} u_{ij} \le M, \forall i = 1, \dots, I. \tag{8}$$

$$\theta S_i \le \sum_{j=1}^{J(i)} x_{ij}, \forall i = 1, \dots, I. \tag{9}$$

$$0 \le u_{ij} \le 1, 0 \le x_{ij} \le \min_{l \in L \& R_{ij}^l = 1} C^l \tag{10}$$

Since we relax the concave utility function to a linear function, we may no longer guarantee the fairness. So we introduce constraint 9 to compensate for the loss of fairness. S_i can be interpreted as the total source rate solved by the original NUM problem with sampled paths. θ is a constant between 0 and 1. We also set x_{ij} to be not larger than the minimum link capacity on its path so that the optimal x_{ij} is indeed feasible. Now the relaxed model can be reformulated as the following convex optimization problem in x and u:

$$\min -x^T I u \tag{11}$$

$$\text{subject to:} x^T R^l u \le C^l, R^l = \boldsymbol{diag}[R_{11}^l, \dots, R_{ij}^l], \forall l = 1, \dots, L. \tag{12}$$

$$Ax \le B \tag{13}$$

$$Du \le E \tag{14}$$

where I is an unit matrix, matrix A and B capture the constraints for x_{ij}, matrix D and E capture constraints for u_{ij}. We can use a dual based decomposition approach to solve this problem.

First, we derive the partial Lagrangian by relaxing the coupled inequality constraint 12 into the objective function:

$$\min_{x,u} L(x, u, \lambda) = \min_{x,u} -x^T I u + \sum_{l=1}^{L} \lambda^l (x^T R^l u - C^l) \tag{15}$$

$$\text{subject to: } Ax \le B$$
$$Du \le E$$

This is a convex program on variable x and u and the optimality could be reached for a given $\lambda_{(k)}$. Suppose that the dual problem is given by $g(\lambda_{(k)}) = -{x^*}^T I u^* + \sum_{l=1}^{L} \lambda_{(k)}^l ({x^*}^T R^l u^* - C^l)$, then the dual optimal solution could be solved using a gradient method using the following updating equation:

$$\lambda_{(k+1)}^l = \lambda_{(k)}^l + \alpha({x^*}^T R^l u^* - C^l), \forall l = 1, \dots, L, \alpha > 0. \tag{16}$$

Second, for a given $\lambda_{(k)}$, the optimal solution to the problem 15 can be solved using a standard technique. Problem 15 has two variables x and u and is convex in (x, u). It can be transformed into the following standard form for a given $\lambda_{(k)}$:

$$\min_x L_1(x) \tag{17}$$

$$\text{subject to: } Ax \le B$$

where:

$$L_1(x) = \min_u -x^T I u + \sum_{l=1}^{L} \lambda_{(k)}^l (x^T R^l u - C^l)$$

$$\text{subject to: } Du \le E$$

Problem 17 can be solved using a subgradient algorithm and the subgradient of $\partial L_1(x)$ at $x_{(k')}$ is given by:

$$\partial L_1(x_{(k')}) = -Iu^* + \sum_{l=1}^{L} \lambda_{(k)}^l R^l u^*$$

where u^* is the solution to the following linear problem:

$$\min_u (-{x_{(k')}}^T I + \sum_{l=1}^{L} \lambda_{(k)}^l {x_{(k')}}^T R^l) u - \sum_{l=1}^{L} \lambda^l C^l$$

$$\text{subject to: } Du \le E$$

By iteratively solving the two above mentioned step, we can reach the optimal x^* and u^* to the relaxed problem 6.

V. Simulation Result

In this section, we give a simple simulation result for our proposed algorithm to the relaxed problem. We use a simple network topology where there are two nodes and three links with capacity 1,2 and 1. There are two sources whose source node and available paths are the same. The maximum number of paths that the source can use is 2. We assume that the source rate for each source should be larger than 1. We also choose all the step size used in the algorithm to be 0.3. The simulation result for the algorithm is shown in 1:

There's indeed some limitations with this simulation. First, the iteration number is only 100 because of the long execution time of the algorithm. However, this iteration number may

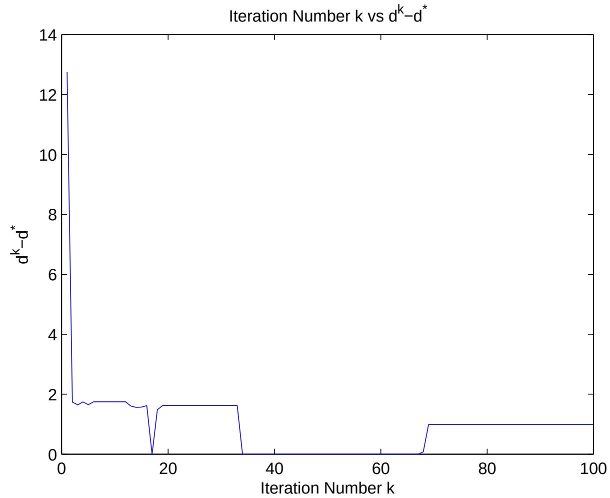

Fig. 1. The best dual value verses the dual value in each iteration.

not guarantee a good convergence result. When the simulation terminates, the optimal source rate for source 1 is $x_1 = [0.2936, 1.9991, 0.7398]$, $u_1 = [0, 1, 0]$. The optimal source rate for source 2 is $x_2 = [0.2607, 1.8174, 0.8520]$, $u_2 = [0, 1, 0]$. We can see that we are still not achieving a good convergence result. So we need more iterations, perhaps thousands of iterations to get a satisfactory optimal value.

The proposed algorithm for solving the relaxed problem is not efficient. So an efficient approximation algorithm is really needed in practical situation.

VI. Conclusion

In this paper, we describe a new model which is called path-aware multipath NUM problem, that extends the traditional multipath NUM problem to new trend in the network development. The proposed model is hard to solve using a deterministic method, so an approximation method is needed to solve the newly proposed model. And this approximation method will be carefully studied in the future research. In this paper, we concentrate on solving the relaxed version of the proposed model, using a partial dual based decomposition method. In order for the relaxed problem to have a unique optimal solution, we modify the proposed model and give detailed algorithm for solving this model.

References

[1] S. H. Low and D. E. Lapsley, "Optimization flow controli: basic algorithm and convergence," *IEEE/ACM Transactions on Networking (TON)*, vol. 7, no. 6, pp. 861–874, 1999.

[2] X. Lin and N. B. Shroff, "Utility maximization for communication networks with multipath routing," *Automatic Control, IEEE Transactions on*, vol. 51, no. 5, pp. 766–781, 2006.

[3] W.-H. Wang, M. Palaniswami, and S. H. Low, "Optimal flow control and routing in multi-path networks," *Performance Evaluation*, vol. 52, no. 2, pp. 119–132, 2003.

[4] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," in *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 4. ACM, 2008, pp. 63–74.

[5] D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley, "Design, implementation and evaluation of congestion control for multipath tcp," in *Usenix NSDI*, 2011.

[6] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.

[7] W. Guo, Z. Wang, and Y. Zhang, "Delay-based congestion control for multipath tcp," 2013.

[8] S. Leyffer, "Deterministic methods for mixed integer nonlinear programming," *Doktorarbeit, University of Dundee*, 1993.