Optimizing the Dynamic Distribution of Data-stream for High Speed Communications

ZW. Zhao, I-Ming Chen

Singapore-MIT Alliance program, Nanyang Technological University, Singapore 639798

***Abstract*—The performances of high-speed network communications frequently rests with the distribution of data-stream. In this paper, a dynamic data-stream balancing architecture based on link information is introduced and discussed firstly. Then an algorithm for simultaneously obtaining the passing nodes and links of a path between any two source-destination nodes rapidly, as well as a dynamic data-stream distribution planning is proposed. Some related topics such as data fragment disposal, fair service, etc. are further studied and discussed. Besides, the performance and efficiency of proposed algorithms, especially for fair service and convergence, are evaluated through a demonstration with regard to the rate of bandwidth utilization. Hoping the discussion presented here can be helpful to application developers in selecting an effective strategy for the distribution of data-stream.**

***Index Terms* —Data-stream balancing, Adjacency matrix, Service metric, Link information, Non-directional graph, Fair service.**

I. INTRODUCTION

PRESENTLY, it is well known that data-stream balancing is taking a pivotal role in the course of data-stream planning and control in networks. With the deeply study and widely application of communication networks technology, the capability of data-stream balancing has been being improved and advanced. Some valuable achievements have been achieved and moved forward, e.g. MPLS-TE (Multiple Protocol Label Switching-Traffic Engineering), which can provide a preferable support to TE [25,…,29]. Awduche, E. Dinan, etc. [1] proposed a framework for dynamic data-stream partition in MPLS such that input data packets can be mapped onto multiple parallel paths. However, the data-stream model in supposed conditions is too ideal. So it is difficult to describe the performances of the model accurately. For all that, the simulation result can indicate that dynamic data-stream balancing shows more advantages than static data-stream balancing mode. Widjaja, et al presented the conception of adaptive traffic engineering, in which the main object is to measure and analyze transmitted data-stream, based on the congestion of paths [2], in order to balance the data-stream for avoiding congestion. Nevertheless, the approach is only created according to the viewpoint of congestion, so as not to desirably meet the basic notion of data-stream planning that is to optimize the utilization of network resources. Furthermore, M. Heusse [3] proposed a WFR (Weighted Fair Routing) mode basing on the measurement of real-time data-stream. With this method, the data-stream balancing is achieved by taking the changes of network loads and connected links for every data-stream admission into consideration, according as hop-by-hop mechanism. Also the algorithm can be combined with OSPF protocol in IP network and PNNI (Private Network-to-Node Interface) protocol in ATM network. However, from the simulation result, the stability of the algorithm is not preferable because an intensive jitter for path switching is easy produced as high-speed routing update. Lagoa, et al [4] try to deal with data-stream balancing by way of distributed optimization method in multiple paths through the so-called sliding model control. Hence, many challenges still exist in the study and applications of data-stream planning. Remaining sections of this paper are organized as follows. In section 2, a framework of a dynamic data-stream planning is discussed. An algorithm of simultaneously collecting the passing nodes and links of a path between any two source-destination nodes is studied in section 3. Furthermore, a strategy for dynamic data-stream balancing, and related topics are analyzed and evaluated in section 4 and section 5. Finally, section 6 concludes the paper.

Manuscript received Nov. 19, 2004.

ZW. Zhao is a Research Fellow of the SMA under IMST (Innovation in Manufacturing Systems & Technology) program and member of technical committee of SMA (corresponding author, e-mail: zwzhao@ntu.edu.sg phone: +65-67904273; fax: +65-68627215).

I-Ming Chen is a SMA (Singapore-MIT Alliance) Fellow under IMST Program, and also an associate professor of Nanyang Technological University. He is now serving on the editorial board of IEEE/ASME Transactions on Mechatronics, and being member of IEEE and ASME, Chairman of IEEE Robotics and Automation Society, Singapore Chapter (e-mail: michen@ntu.edu.sg).

II. BASIC ARCHITECTURE OF DYNAMIC DATA-STREAM PLANNING

In terms of implementary mechanism, there are three approaches for data-stream planning, i.e. time-dependent (TD), state-dependent (SD) and event-dependent (ED). TD mode is a long-term (e.g., diurnal) behavior optimization for whole networks. That is, to schedule and regulate the distribution of data-stream in a macro-scale range, according to long-term observation and statistic analysis for variable data-stream. Therefore, TD mode cannot adapt to instant traffic variety as well as the short-time diversification of network status. In comparison with TD mode, SD mode belongs to a dynamic adaptive data-stream control basing on immediate network status, so it can overcome the shortcomings of TD mode. In SD mode, some real-time parameters must be gotten in time, whereas the burden of network computation is heavier than that of in TD mode. Originally, ED mode is an adaptive and distributed approach, and typically it uses a learning model to find out better paths for a certain data-stream in networks, without the requirement of ALB (available-link-bandwidth) flooding. Nonetheless, an accurate learning model may bring on extra complexity. Accordingly, taking and the advantages of ED mode and SD mode, along with the variable state of whole network which is decided by the variable state of every link. A model for data-stream planning (Fig.1) is proposed utilizing the real-time feedback information of links in this section. The pretreatment is included of some data processing units, such as classification, filter, labeling, queuing, measurement etc.

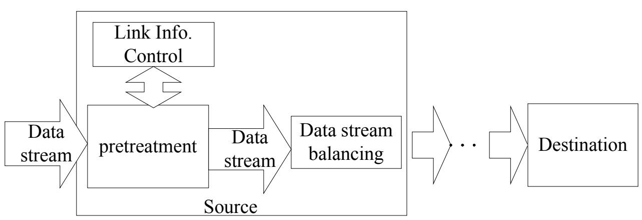

Fig. 1. Dynamic data-stream planning model based on link status

- Link information control modular: Control the propagation and admission of link information, resorting to some signaling as the flooding transmission of link information for optimized paths [9], EIGP [14], etc. Moreover, in order to build multiple paths with higher service metrics, select routing paths, update local data, and the implementation of data-stream balancing between any two nodes, a temporary traffic database is needed here.
- Dynamic data-stream balancing modular: Mapping all admission packets onto multiple paths according to proposed strategy and related algorithms in this paper.

III. A Fast Algorithm for Getting the Nodes and Links included IN a Path

For building multiple paths, local topology update, as well as routing selection, etc., it is necessary for any source-destination nodes to build a temporary traffic database based on network topology. An essential algorithm is proposed in this section.

A network topology can be emulated by non-directional graph G or directional graph D. The vertices and edges of a graph can respectively describe the nodes and links of a network. The theorem Menger [5] indicates that the connectivity of networks is an important metric. In general, the higher connectivity corresponds with the better reliability of networks. However, it is impossible to utilize a higher connectivity to obtain a better reliability in an actual network due to well-known factors, whereas a networks topology should be as simple as possible because of not only cost but also the implementary complexity. So the reliability and others characteristics of a network can be ensured by way of the advanced ability to transmit and manage network data-stream.

Supposing the topological graph of a network can be expressed by （V, E, φ）, and the graph may be a non-directional graph G or a directional graph D, where, the set of vertices is $V = \{x_1, x_2, \ldots, x_v\}$, the set of edge is $E = \{a_1, a_2, \ldots, a_e\}$. The adjacency matrix of G or D can be formed according to $A = (a_{ij}), a_{ij} = \mu(x_i, x_j)$, where, $\mu(x_i, x_j)$ denotes either the number of edge in a directional graph D whose starting vertices is x_i and end vertices is x_j, or the number of edge between x_i and x_j in a non-directional graph G.

Theorem 1 [5]: Assume A=A (D) is the adjacency matrix of a directional graph D, then within A^k the element corresponding to (i, j) represents the number of path inclusive of K directional edges, between x_i and x_j.

According to theorem 1 and an adjacency matrix calculated by above formula, the number of path inclusive of K directional edges between any two nodes in a network topology graph can be confirmed.

Example : Assume that the topology of a directional graph D is as Fig. 2, then the set of vertices is $V = \{A, B, C, D\}$, the set of edge is $E = \{a1, a2, a3, a4, a5\}$. The adjacency matrix of D is as $A(D) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$, and its corresponding table format is as

	A	B	C	D
A	0	1	1	0
B	0	0	1	1
C	0	0	0	1
D	0	0	0	0

Here, if we want to obtain the number of a path including two directional edges (links), that is to say, let K=2, then go ahead following computations:

$$A^2(G)=\begin{bmatrix}0&1&1&0\\0&0&1&1\\0&0&0&1\\0&0&0&0\end{bmatrix}^2=\begin{bmatrix}0&0&1&2\\0&0&0&1\\0&0&0&0\\0&0&0&0\end{bmatrix}.$$

The corresponding table format is

	A	B	C	D
A	0	0	1	2
B	0	0	0	1
C	0	0	0	0
D	0	0	0	0

It is obvious that the number of path including two edges between A and C, A and D, as well as B and D is 1, 2 and 1 respectively. Similarly, let K=3, we have

$$A^3(G)=\begin{bmatrix}0&1&1&0\\0&0&1&1\\0&0&0&1\\0&0&0&0\end{bmatrix}^3=\begin{bmatrix}0&0&0&1\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}.$$

It is obvious that only one path exists, which passes through 3 directional edges between A and D.

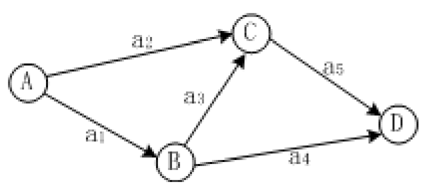

Fig. 2. Topology of directional graph D

Therefore, once a network topology is decided, the number of path including the given amount of directional links between any two nodes, can be confirmed basing on above theorem. However, the links and nodes included in a path are also needed for optimizing routing path selection and other algorithms. However, a problem still exists: The links of network is usually bi-directional, so the topology should be described with a directional graph. Nevertheless, the complexity is higher if to form a directional topological graph between any two nodes. Therefore, we prefer a non-directional topology graph to a directional one. Consequently, the problem of how to decide the number of directional links between any two nodes utilizing a non-directional topology graph can be met. Here, according to the definition of non-directional graph, it can be found that the topological matrix of non-directional graph is symmetrical. Moreover, the adjacency matrix of a directional graph always can be adjusted to be an upper triangular matrix or a lower triangular matrix, by regulating the definition of links' direction, that is the direction a source node points is the same as the direction of links. And the non-zero elements in the regulated triangular matrix are the same as those in the non-directional graph. In this way, the matrix of a non-directional graph can be transformed into the matrix of a directional graph. Further, for getting the nodes and the number of directional links, passed by every path, go ahead following processes.

Supposing the topological matrix of a network is the adjacency matrix of directional graph D as $A(D)=(a_{ij}),\quad (i,j=1,2,\ldots,n)$. i and j respectively denote the number of row and column of the matrix, and $A(D)$ is considered to be an upper triangular matrix by link regulation. Moreover, the elements of the matrix are only either 1 or 0 (any elements of a matrix can be transformed into either 1 or 0 by elementary transform). Whereby the rule can be expressed by

(1) When i=1, and if the row element of the matrix is 1, that is to say, a_{1j}=1. It is obvious that the nodes within column j (corresponding a_{1j}=1) are the adjacency nodes of starting node.

(2) Let $i=j$, then if the elements within row i are 1, the nodes in the corresponding columns are respectively the adjacency nodes of starting node in order.

(3) Independently collect each node, which is found out in step 2 and corresponding to column j. Repeat step 2 until the element, which belongs to both a selected row and the column corresponding to the end node, is 1 (we define the course as successful search). That is to say, the element within row i ($i=j$) and in the meantime within the column corresponding to the end node, is 1.

Hereby, the number of directional link between starting node and end node is equal to the times of repeating search (successful search) plus 1. In the process of each successful search, the selected nodes are just the nodes that are passed through by each path. And the number of directional links between starting node and end node are equal to the number of node between starting node and end node (excluding starting node and end node) plus 1.

IV. DYNAMIC DATA-STREAM BALANCING ALGORITHMS

The calculation of forwarding table in IGP (Interior Gateway Protocol) is based on topology-driven as well as a simple metric inclusive of hops or other values. No information like the availability of bandwidth and the features of traffics, etc. are released. It means that network loads needn't be considered for the calculation. Therefore, it is possible to result in the unbalance distribution of network traffics so that network resources cannot be well utilized. Hereby, in this section a data-stream balancing strategy is proposed so as to enable network traffics to be sound distributed, as reasonable as possible by way of real-time adjustment for data-stream distribution.

Currently, three main algorithms for traffic partition have been applied in networks. They are as: the mode based on round robin, the mode of OSPFv2-ECMP (Equal Cost Multi-Path) [7], and the mode of OMP (Optimal Multi-Path) [10]. The main idea of ECMP is to balance traffics by way of even traffic partition among links. In OMP, to partition traffics is according as a hash function applied to resource-destination nodes. However, the first algorithm is only applied to the situation of same link delay, because severe disorder data-stream may occur when a small variety of link-delay is produced in networks. The other algorithms cannot adapt to the dynamic diversification of network state owing to static topology-driven to which they belong, although traffics can be arranged to different links also.

A. Dynamic data-stream partition under concave service metric

Concave service metrics are inclusive of available bandwidth, reserved bandwidth, packet throughput, dissemination speed, etc., and normally refer to bandwidth. In this case, the service metric of routing path *P* should be decided by

$$W(P) = \min\{W(i,j), \quad W(i,k), \quad \ldots, \quad W(x,y)\} \quad .$$

Hereby, the service metric of bandwidth of a path is determined by the links with minimal bandwidth, along this path between two nodes.

Here, firstly suppose that the concave service metric information of links between resource-destination nodes has been obtained by the mode described in literature [9] or other mechanisms. So many routing paths can be formed according to the information for admission data-stream. After that, traffic distribution within multiple paths can be implemented according as below algorithms.

Assume that $W_{l(i,j,k)}$ represents the concave service metric provided for node k by node j in path i. Firstly to sort all the concave service metrics respectively for each path by order. So a minimal service metric can be respectively achieved for each path, which is denoted by

$$\min\{W_{l(1,1,k11)}, W_{l(1,2,k12)}, \ldots, W_{l(1,j,k1n)}\} \overset{\text{令}}{=} W_{l1}$$

$$\min\{W_{l(2,1,k21)}, W_{l(2,2,k22)}, \ldots, W_{l(2,j,k2n)}\} \overset{\text{令}}{=} W_{l2}$$

$$\vdots$$

$$\min\{W_{l(i,1,ki1)}, W_{l(i,2,ki2)}, \ldots, W_{l(i,j,kin)}\} \overset{\text{令}}{=} W_{li} .$$

Then the minimal service metrics are compared each other, so that current sort order for all paths' metrics can be gotten as

$$sort\{W_{l1}, W_{l2}, \ldots, W_{li},\}$$
$$= \{W_{lm1}, W_{lm2}, \ldots, W_{lmi}\}, \quad W_{lm1} \geq W_{lm2} \geq \ldots \geq W_{lmi}$$

The corresponding average value is

$$\overline{W_l} = \sum\nolimits_{j=1}^{i} W_{lmj} \Big/ i \quad . \tag{1}$$

Supposing the on demand service metric of a data-stream is W_s, well then to arrange the data-stream according with below assignment rules.

Case 1: If $W_s \geq \overline{W_l}$, N service metrics are continuously chosen beginning from maximum value in the sort order (normally N<=5) and satisfy

$$\sum_{j=1}^{N} W_{lmj} > W_s , \quad N \leq i . \tag{2}$$

In the meantime, also should satisfy

$$\frac{W_s}{\sum_{j=1}^{q} W_{lmj}} \leq (50 + k\Delta \pm \Delta)\% \leq 1, \quad \Delta > 0, q \leq N \tag{3}$$

where, K is a positive integer, [(50+KΔ-Δ)%, (50+KΔ+Δ)%] is as a fuzzy comparison range. The reason of such assignment is for preventing the traffic distribution from burst, so as to avoid unfair service. The fiducial value of iterative comparison is 50% such that a reasonable quantity of paths can be obtained for data-stream balancing. The Δ is a threshold of fuzzy comparison range (normally equal to 1 or a larger number), which can confirm the fuzzy comparison range. K is an increment proportional coefficient and its initial value may be equal to 0.

If one or several suitable service metrics for formula (3) cannot be obtained, when K is with initial value, and q changes from its initial value 0 to a maximum value (e.g. 5). Then, repeat the calculation after K=K+1, till (50+KΔ-Δ)%≥1. If a satisfied service metrics still cannot be found, it indicates that the on demand service cannot be provided at this time. So the on demand data-stream can be disposed to be as waiting status or discarded. However, if formula (3) can be satisfied, the traffic distribution algorithm should go ahead as

$$b_1 = \frac{W_{lm1}}{\sum_{j=1}^{q} W_{lmj}}, b_2 \frac{W_{lm2}}{\sum_{j=1}^{q} W_{lmj}}, \ldots, b_q = \frac{W_{lmq}}{\sum_{j=1}^{q} W_{lmj}} . \tag{4}$$

It is obvious that $b_1 + b_2 + \ldots + b_q = 1$, where $b_1 \sim b_q$ are respectively the percentage of concave service metrics assigned to each selected path.

Case 2: If $W_s < \overline{W_l}$, it is indicated that the probability of current network loads is smaller, and/or the class of concave service metric for the on demand data-stream is lower. Hereby, a service metric close to W_s and greater than (or equal to) W_s among the sort order is selected to be a service provider for the data-stream at this time. Such that not only the QoS of the data-stream can be guaranteed, but also the number of data-stream assignment, and more resource (e.g. bandwidth) fragments which can decrease the utilization of network resources, can be abated effectively. In fact, as the lack of network loads to some extent, concave parameters like bandwidth, delay, etc. can be easily satisfied, and at that time it is not necessary to emphasize keeping data-stream balancing in networks**.**

Case 3: The service metrics of all routing paths beginning from an identical node are changeable, after an admission traffic is distributed to multiple paths according with above algorithms. So some operations need to be done, that is, to repeat sort order and re-calculate $\overline{W_l}$ in terms of formula

(1). And before these operations, the service metrics of multi-path should be modified. For the case 1, the method of modifying service metrics for multi-path can be coped with {Current service metric array}=

$$\{(W_{lm1} - W_s \cdot b_1), (W_{lm2} - W_s \cdot b_2), \ldots, (W_{lmq} - W_s \cdot b_q)\} \quad (5)$$

where, b_j is a percentage of concave service metrics assigned to each selected path, which is indicated in formula (4). For the case 2, the method of modifying service metrics is dealt with

{Current service metric array}=

$$\{W_{lm1} - W_s, W_{lm2}, \ldots, W_{lmi}\} \quad (6)$$

B. Dynamic data-stream partition under additive service metric

There are some additive service metrics for some kinds of traffics, e.g. delay, jitter, number of node, link cost, etc. Under this situation, the service metric of traffics is decided by $W(P) = W(i, j) + W(i, k) + \ldots + W(x, y)$. Hereby, the additive service metric of any one path is determined by the summation of the service metrics of all links along the path. Generally, the data-stream with the requirement of additive service metric also has the requirement of concave service metric simultaneously, such as real-time voice data-stream with critical demand of delay, jitter and minimal bandwidth at the same time, etc. Hence, two or more kinds of service metrics need to be taken into account for data-stream balancing. As before, firstly suppose that the additive service metric information of links between resource-destination nodes can be obtained, and whereby many routing paths can also be formed according to the information for admission data-stream. After that, traffic arrangement within multiple paths should be implemented according as following algorithms.

Assume that $W_{l(i,j,k)}$ represents additive service metrics provided for node k by node j in path i. Firstly to sum up all the additive service metrics respectively for each path, and to sort them by order. The result is given by

$$W_{A(1,1,k11)} + W_{A(1,2,k12)} + \ldots + W_{A(1,j,k1n)} \overset{令}{=} W_{A1}$$

$$W_{A(2,1,k21)} + W_{A(2,2,k22)} + \ldots + W_{A(2,j,k2n)} \overset{令}{=} W_{A2}$$

$$\vdots$$

$$W_{A(i,1,ki1)} + W_{A(i,2,ki2)} + \ldots + W_{A(i,j,kin)} \overset{令}{=} W_{Ai}.$$

Then compare the results with each other, and the ascending sort of additive service metric for given paths can be obtained as

$$sort\{W_{A1}, W_{A2}, \ldots, W_{Ai},\}$$
$$= \{W_{Am1}, W_{Am2}, \ldots, W_{Ami}\}, \quad W_{Am1} \le W_{Am2} \le \ldots \le W_{Ami}$$

Supposing $W_{Amh} \le W_A$, $h = 1,2,\ldots,i$, W_A is the additive service metric of on demand data-stream. Obviously, if the additive service metric is only be cared, the path corresponding to W_{Ami} should be selected for the service. However, if both concave service metric and additive service metric are needed by the traffic, the problem on how to assign the traffic within the given paths and simultaneously taking the two kinds of service metric into consideration can be met. Accordingly, following method can help to reach the target. First, to sort the concave service metrics of links whose additive service metrics can satisfy $W_{Amh} \le W_A$, $h = 1,2,\ldots,i$ by descending sort, and then to distribute the traffic among multiple paths according as the traffic assignment algorithms under concave service metric denoted by formula (1) ~ (6). At the same time, the modification for the two service metrics also should be coped with after data-stream assignment. The modifications on additive service metric are simpler (only plus or minus calculation) than those on concave service metric, as long as the feedback information of links for the given paths is accurate and real time. Furthermore, the basic idea of the method is also suitable for the data-stream with the requirement of more kinds of service metrics.

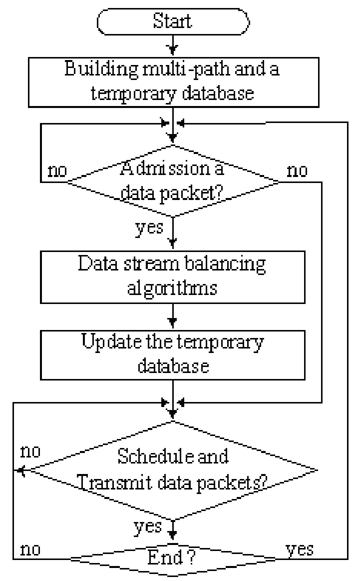

Fig. 3. Flow of dynamic data-stream balancing

C. Implementation of dynamic data-stream balancing

As for MPLS, the SPF (Shortest Path First) algorithm is still used to calculate a shortest path in link-state protocols such as IS-IS and EIGP, etc., and to be the reference of building a routing table. In fact, Data-stream balancing is not based on shortest path which is only one priority option for path selection. In actual applications, the calculation of all directional links between two nodes is heavy, so as not to ensure the real-time features of routing decision. Therefore, first to compute the passing nodes and links between any two nodes according to network topology utilizing the algorithm proposed in section 3, and build multi-path basing on the information of related links, obtained by way of some signalings [9] [14]., and then store the results into a temporary data-stream database, which can be updated by the new information of links. Hereafter, once data-stream balancing is needed for an on demand service, first to confirm several available paths, which are mapped by the temporary data-stream database, and then go ahead traffic assignment using previous algorithms according with the type of service. The concise flow of data-stream balancing is shown in Fig. 3.

D. Related topics

1) The number of valid paths between nodes

A larger storage and longer disposal time are needed if the information of total links and corresponding nodes between source-destination nodes are all included in data-stream balancing database. In fact, it is not necessary because normally several paths with higher metrics are enough and reasonable for data-stream balancing. Normally, some paths with higher service metrics may pass through more nodes, and some paths with lower metrics may be pass through fewer nodes, because traffic distribution is not well balanced in networks. Therefore, no matter how many nodes passed through by a path, more data-stream can be admitted as long as the path is with a higher service metric. Whereby, to balance the service metric of every link is a primary meaning for data-stream balancing. Usually the convergence of data-stream balancing algorithm can be speeded up as utilizing the paths with higher metrics, such that a cost-effective service may be achieved.

2) Reassign bandwidth fragment

Bandwidth fragment distributing in every path will be produced after traffic assignment, and it is also changeable dynamically. If the bandwidth fragment is omitted, much network resource may be wasted from the viewpoint of whole networks. How to use the bandwidth fragment? Here, two methods are proposed as: Reassign traffics depending on an optimization algorithm, which may produce longer delay whereby leading to a lower efficiency for data-stream planning; Allot bandwidth fragment to the kinds of "Best Effort" data-stream. We consider the second approach is better by reason of without extra delay, because there is no traffic reassignment in the approach.

3) Fairness of Services

In the course of dynamic data-stream assignment, it is the best to distribute one traffic into an identical path so as to avoid data recombination in receiver, especially for some traffic particularly like real-time data streams, etc. Furthermore, if there are multiple paths that can satisfy the demand, and among them a path whose service metric is approximate equivalent to the required service metric, should be chosen to carry the traffic, such that it can not only ensure the fairness of service but also avoid producing more bandwidth fragment. Based on this viewpoint, for concave service metric, the maximum value 1 of （50+ K$\Delta\pm\Delta$）% in formula (3) should be firstly assumed, and then finish the computation of data-stream balancing. Furthermore, because generally the traffic with the demand of additive service metric still involves the demand of concave service metric in the same time, the actual disposal should be as: First to detect if a path whose maximum additive service metric W_{Ami} and concave service metric, can both satisfy the demand of an on demand data-stream. If yes, the path will carry the traffic. Otherwise, choose $W_{Am(i-1)}$ and detect again, till W_{Am1}. Note that here $W_{Amh}, h = 1,2,\ldots,i$ can satisfy the demand of additive service metric for the traffic. If one path cannot be found to carry the service, multi-path should be selected to undertake the task. Accordingly, for the multi-path, the maximum value 1 of （50+ K$\Delta\pm\Delta$）% in formula (3) needs to be firstly assumed then start the iterative computation, till suitable paths are obtained. Afterwards, go on calculation using the algorithms on the situation of concave service metric proposed before, so as to deal with such traffic assignment. In this way, although the traffic distribution is not well proportional, the fairness of service can be guaranteed and with low complexity for data-stream balancing. So it is a recommended solution for the transmission of real-time services.

V. Performance Analysis

The performance of presented algorithms can be examined by a demonstration running on Matlab platform. Assume 5 available paths have already existed between node A and B, which is denoted by {P1, P2, P3, P4, P5}, and their service metrics are concave and stochastic. The rule of input data-stream follows the Poisson distribution. Supposing the service metric of stochastic traffic is also concave metric and its range is of (0, 1), which is the distributing area of W_s. The initial value of K equals to 0. Let Δ=3. Based on formula (1)~(6), the result of data-stream assignment can be measured according to following formula, after traffic assignment among multi-path is implemented till a stochastic input traffic cannot be distributed to the paths for service.

(Total resources – Remainder resources) / Total resources = Rate of resource utilization

Whereby, the data-stream mapping among the concave service metrics of stochastic traffic, the given paths, and the traffic assignment times is shown in fig. 4. Obviously, the proposed algorithms can ensure the fairness, rationality and validity of resource utilization among multiple paths.

VI. Conclusion

A high-level data-stream balancing represents the trend of network improvement, it is more important for ever-increasing network loads and kinds of data-stream. Although static data-stream balancing mechanism is a dominant approach in current networks, is has been known that a dynamic data-stream balancing strategy can excel static mechanisms in many aspects such as real-time features, adaptability, flexibility, etc. In this paper, a dynamic data-stream planning is firstly discussed and studied, and after that an algorithm for fast obtaining the passing nodes and links of a path between any two nodes according to network topology is proposed, being a foundational algorithm for data-stream balancing. Furthermore, a data-stream balancing strategy are proposed and analyzed based on the real-time information of links. As a result, the characteristics of proposed algorithms, such as fair service, adaptability, etc. are suitable for dynamic data-stream balancing.

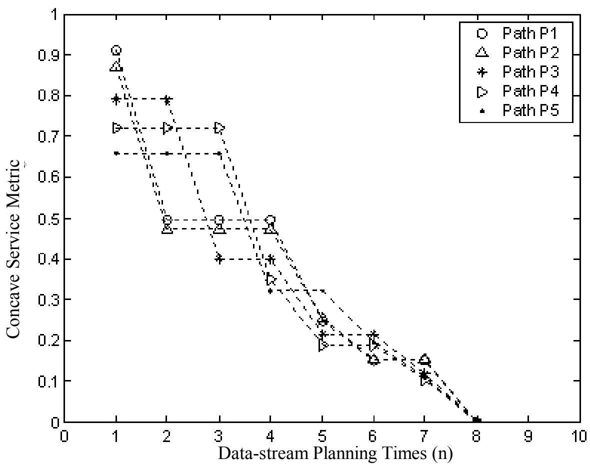

Fig. 4. Demonstration of data-stream balancing

REFERENCES

[1] E. Dinan, D.O. Awduche, B. Jabbari, "Analytical Framework for Dynamic Traffic Partitioning in MPLS Networks," IEEE/ACM Trans. Networking, June 2000, pp.1604-1608.

[2] A. Elwalid, C. Jin, S. Low, I. Widjaja, "Mate: MPLS Adaptive Traffic Engineering," IEEE/ACM Trans. Networking, March 2001, pp.1300-1309.

[3] M. Heusse and Y. Kermarrec, "A new routing policy for load balancing in communication networks," IEEE/ACM Trans. Networking, Jan. 2001, pp.267-272.

[4] C. Lagoa and H. Che, "Decentralized Optimal Traffic Engineering in the Internet," ACM SIGCOMM Computer Communication Review, Volume 30, Issue 5, October 2000, pp.39-47.

[5] J.M. Xu, Theory of Graph and its Application. HeFei, Press of University of Science and Technology, PR.China, 1998, pp.20-39.

[6] D. Awduche et al, "Requirements for Traffic Engineering Over MPLS," IETF RFC2702, September 1999.

[7] J. Moy, "OSPF Version 2," IETF RFC2328, April 1998.

[8] G. Apostolopoulos et al, "QoS Routing Mechanisms and OSPF Extensions," IETF RFC2676, August 1999.

[9] Z. Zhao, "Dynamic Planning of Network Traffic and its Applications," Ph.D. thesis, Beijing university of aero&astro, Beijing, PR.China, Chapter 5, April 2002.

[10] C. Villamizar, "OSPF Optimized Multi-path (OSPF-OMP)," Internet-draft, draft-ietf-ospf-omp-00, March 13, 1998. Available: http://www.ietf.org/proceedings/99mar/I-D/draft-ietf-ospf-omp-02.txt.

[11] T. Narten, E.Nordmark, and W. Simpson. "Neighbor Discovery for IP Version6 (IPv6)," IETF RFC1970, Aug. 1996.

[12] J. Moy, "OSPF Standardization Report," IETF RFC2329, April 1998.

[13] A. Retana et al, "OSPF Stub Router Advertisement," IETF RFC3137, June 2001.

[14] R. Albrightson et al, "EIGRP-A Fast Routing Protcol Based on Distance Vectors," Proc. Networld/Interop. 94, May 1994.

[15] P. Narvaez and K. Y. Siu, "Efficient Algorithms for Multi-Path Link State Routing," ISCOM'99, Kaohsiung, Taiwan, 1999.

[16] S. Vutukury and J.J. Garcia-Luna-Aceves, "MDVA: A Distance-Vector Multipath Routing Protocol," In Proceedings of the INFOCOM, 2001.

[17] S. Vutukury and J.J. Garcia-Luna-Aceves, "A Simple Approximation to Minimum-Delay Routing," SIGCOMM '99, September 1999.

[18] Y. Wang and Z. Wang, "Explicit routing algorithms for Internet traffic engineering," ICCCN'99, 1999.

[19] N.G. Duffield and M. Grossglauser, "Trajectory Sampling for Direct Traffic Observation," IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 9, NO. 3, June 2001, pp.280-292.

[20] C. Hopps, "Analysis of an Equal-Cost Multi-Path Algorithm," IETF RFC 2992, 2000.

[21] S. Bahk and M. Zarki, "Dynamic multi-path routing and how it compares with other dynamic routing algorithms for high speed wire area networks," ACM computer communication review, vol.22, no.4, Oct. 1992, pp.54-64.

[22] Y. Lee and Y. Choi, "An adaptive flow-level load control scheme for multipath forwarding," Springer-Verlag, July 2001, pp.771-779.

[23] S. De and S.K. Das, "Dynamic multipath routing (DMPR): an approach to improve resource utilization in networks for real-time traffic," Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2001, pp.23-30.

[24] I. Cidon, R. Rom and Y. Shavitt, "Analysis of multi-path routing," IEEE/ACM transaction on Networking, Dec. 1999, vol.7, no 6, pp.885-896.

[25] Y. Seok, Y. Lee, Y. Choi, and C. Kim, "Dynamic constrained multipath routing for MPLS networks," Computer Communications and Networks, Oct. 2001, pp.348-353.

[26] Y. Lee et al, "A constrained multipath traffic engineering scheme for MPLS networks," ICC 2002, April 2002.

[27] J.P. Wang, S. Patek, H.Y. Wang et al, "Traffic Engineering with AIMD in MPLS Networks," LNCS, May 2002.

[28] H. Saito, Y. Miyao, and M. Yoshida, "Traffic engineering using multiple multipoint-to-point LSPs," INFOCOM'2000.

[29] S.S. Lee and M. Gerla, "Fault tolerance and load balancing in QoS provisioning with multipath MPLS paths," Proc. of 9th-IWQOS, Germany, June 2001.

ZW. Zhao is currently a Research Fellow of Singapore-MIT Alliance. He received the Ph.D. degree from Beijing University of Aero&Astro, PR. China in 2002. His research interests are in space networking, self-organized communication networking, real-time power-line communication and embedded systems. He is now a member of the technical committee of Singapore-MIT Alliance.

I-Ming Chen obtained the Ph.D. degree from California Institute of Technology, Pasadena, U.S.A. in 1994. He was JSPS Visiting Scholar in Kyoto University, Japan in 1999, and currently Fellow of Singapore-MIT Alliance. His research interests are in reconfigurable automation, real-time networking and smart material based actuators. Dr. Chen has published more than 80 technical articles in refereed international journals and conferences. He is now serving on the editorial board of IEEE/ASME Transactions on Mechatronics. He was Program Co-Chairman of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) in Hawaii, USA, 2001, and Co-Chairman of Technical Program, 7th International Conference on Control, Automation, Robotics & Vision, Singapore, Dec. 2002. He is member of IEEE and ASME, Chairman of IEEE Robotics and Automation Society, Singapore Chapter, Chairman of Prototyping for Robotics and Automation Technical Committee under IEEE Robotics and Automation Society.