Hindawi Publishing Corporation
Scientific Programming
Volume 2016, Article ID 1938312, 8 pages
http://dx.doi.org/10.1155/2016/1938312

Research Article

Bulk Restoration for SDN-Based Transport Network

Yang Zhao,[1] Lei Wang,[1] Xue Chen,[1] and Futao Yang[1,2]

[1] *State Key Lab of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2] *School of Physical Science and Technology, Anshan Normal University, Anshan 114007, China*

Correspondence should be addressed to Yang Zhao; ploenin@163.com

Received 6 January 2016; Revised 6 July 2016; Accepted 25 July 2016

Academic Editor: Basilio B. Fraguela

We propose a bulk restoration scheme for software defined networking- (SDN-) based transport network. To enhance the network survivability and improve the throughput, we allow disrupted flows to be recovered synchronously in dynamic order. In addition backup paths are scheduled globally by applying the principles of load balance. We model the bulk restoration problem using a mixed integer linear programming (MILP) formulation. Then, a heuristic algorithm is devised. The proposed algorithm is verified by simulation and the results are analyzed comparing with sequential restoration schemes.

1. Introduction

The unprecedented amount of data that needs to be communicated imposes new challenges on transport networks [1, 2]. Providing survivability becomes a key issue, as a single transport network failure (such as fiber cut) may cause a huge amount of data loss which would largely degrade or even disrupt network services [3, 4].

Since there could be a lot of data flows passing through one fiber link, upon a single link failure, a number of data flows could be simultaneously disrupted and controller of network is triggered for recovering each affected flow. The bursting big amounts of recovery requests generate intense resource contention [5]. The resource contention is caused by the concurrent attempts of two or more backup path instances to use the same part of the network resource; that is, bandwidth on one or more links could be used up. In addition, since large amount of resource is occupied by restoration, one or more links could be blocked; therefore some subsequent incoming flow requests could be rejected. Therefore, it is of crucial importance to design a strategy for reducing resource contention during restoration.

Recently, a number of researches have been done about bulk restoration. In [6], the authors propose a recovery scheme that recovers concurrently in both the optical and IP layers. The main idea is to recover disrupted flows concurrently in both layers in order to reduce restoration time and avoid congestion. In [7, 8], the authors study bulk restoration for wavelength division multiplexing (WDM) network and flex grid optical network, respectively. The main idea of both the articles is to let subsequent requests reuse the light paths established for previous requests, so that resource contention could be reduced. Reference [9] studies restoration in wavelength convertible wavelength switched optical network (WSON) and wavelength converters are utilized during restoration in order to avoid restoration contention. However, most of the previous works focus on wavelength resource while, in this paper, we mainly pay attention to the bandwidth resource. We try to reduce the contention for bandwidth resource during bulk restoration. Furthermore, some articles including [7] reveal that the order of the input requests affects the results. In [7], the authors repeat the restoration process many times with requests sorted randomly and take the best result. However, the repetitive processing may be time consuming and the result depends on the number of iteration. In this paper, we design a dynamic order mechanism which is described in Section 5. The order of the request to be handled is specified dynamically based on the current network status.

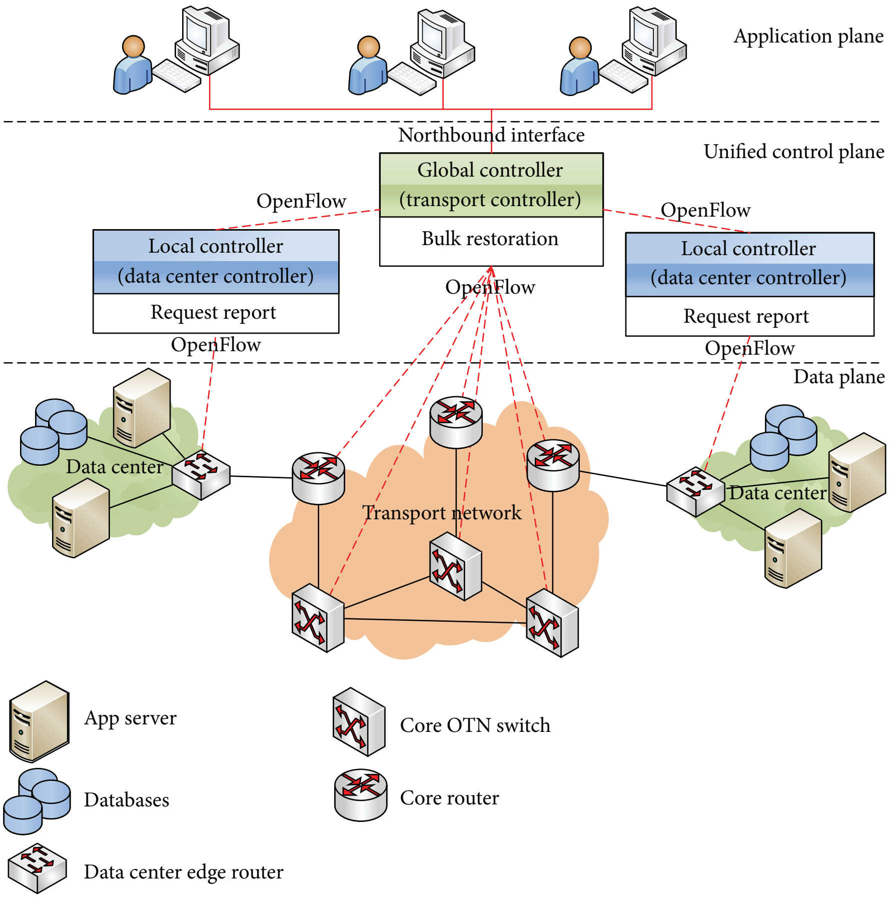


Figure 1: Network control architecture.

This paper proposes a scheme that coordinates the restoration of disrupted flows globally. The key method of this scheme is to schedule backup paths for all the affected flows concurrently, which is different from general sequential restoration schemes. In addition, the idea of load balance is introduced to avoid resource contention. On the one hand, more disrupted flows could be recovered by global scheduling; on the other hand, resource contention is reduced so that network throughput could be improved.

The remainder of this paper is organized as follows. Section 2 introduces the network control architecture and technical background. Section 3 details the principle of bulk restoration. In Sections 4 and 5, we present the mixed integer linear programming (MILP) formulation and heuristic algorithm of the bulk restoration problem, respectively. We evaluate and analyze restoration performance through numerical results in Section 6. Section 7 concludes this paper.

2. Network Control Architecture

This paper introduces a software defined networking- (SDN-) based centralized control architecture which enables a programmable network control including data plane, controller plane, and applications as shown in Figure 1 [10]. Data plane can divide into IP layer and optical layer. The optical transport network (OTN) technology is introduced. Each node in data plane is composed of an OTN switch and an IP router. Light paths are established between adjacent OTN switches, and electric regeneration is performed on each node. Data flows are carried by light paths via optical channel data unit (ODU) channels. All the transport resources are software defined by the centralized controller plane. With global view of network and simple control interface, the controller plane can provide globally optimized routing and resource assignment. Local controllers report flow requests to the global controller. The global controller handles functions including paths calculation, failures positioning, and restoration. Applications which interact with control plane via northbound interface provide customers with various operation functions.

3. Bulk Restoration

In general restoration, backup paths are computed following the first-come-first-served principle and each backup path is calculated independently [11]. However, as some backup path could be routed on the same link, resource contention is unavoidable. As a result, network survivability and throughput are reduced.

Since SDN introduces the centralized control mechanism [12, 13], recovery requests can be coordinated simultaneously by the controller. Upon a link failure causing multiple flows disrupted, all recovery requests are collected by the controller, so that backup paths could be calculated synchronously. What is more, as controller has full knowledge of all the flow requests and the network status, the disrupted flows can be deduced from failure link and configuration of network nodes.

Upon link failure occurrence, the closest optical transport node detects the failure and informs the controller with an OpenFlow PORT_STATUS message [14]. Since the controller maintains the Traffic Engineering Database (TED), the flow list impacted by this failure can be obtained. Based on the current network status, the controller calculates the backup paths by using a bulk restoration algorithm and then switches the data traffic from the work path to the backup path. This process is shown in Figure 2.

To schedule resources and backup paths for multiple recovery requests is an NP-hard problem [15]. So we use MILP bulk restoration (MILP-BR) which is based on the MILP formulation to find the optimal solution. However, the computation of the MILP solution is time consuming, while restoration is generally time sensitive [16, 17]. We propose a heuristic algorithm named contention avoiding bulk restoration (CA-BR) to calculate the near-optimum solution.

4. Bulk MILP Restoration

The bulk restoration problem is known to be NP-hard. We modeled this problem by MILP formulation.

The network topology is modeled through a graph $G(N, L)$, with N network nodes and L bidirectional links. Upon reception of all the recovery requests, controller builds and solves the following MILP formulation to compute backup paths for all the requests.

Variables are the following:

$F^{s,d,r}$ is a binary variable that is 1 if the backup path of the rth flow between source-destination pair (s, d) is set up;

$F_{i,j}^{s,d,r}$ is a binary variable that is 1 if the backup path of the rth flow between source-destination pair (s, d) is routed along link (i, j);

$p_{s,d,r}^{k}$ is the kth candidate backup path for request $d_{s,d,r}$;

$F_{i,j}^{s,d,r,k}$ is a binary variable that is 1 if $p_{s,d,r}^{k}$ is routed on link (i, j).

Constants are the following:

$D^{s,d}$ is the set of recovery requests, between pair (s, d), to be recovered;

$d^{s,d,r}$ is the rth recovery request between pair (s, d);

$b^{s,d,r}$ is the bandwidth demand of rth flow between source-destination pair (s, d);

$B_{i,j}$ is available capacity of link (i, j);

α is a multiplicative factor to be inserted in the triple-objective function;

β is the second multiplicative factor to be inserted in the triple-objective function.

Objective is as follows:

$$\text{Min} \quad \left[\alpha \times \left(\sum_{s=0}^{N}\sum_{d=0,d\neq s}^{N}\sum_{k=1}^{|D^{s,d}|} b^{s,d,r} - \sum_{s=0}^{N}\sum_{d=0,d\neq s}^{N}\sum_{k=1}^{|D^{s,d}|}\left(F^{s,d,r}\times b^{s,d,r}\right)\right) + \beta \times R_{\max} + \sum_{s=0}^{N}\sum_{d=0,d\neq s}^{N}\sum_{r=1}^{|D^{s,d}|}\sum_{(i,j)\in L} F_{i,j}^{s,d,r}\times b^{s,d,r}\right] \tag{1}$$

$$\text{subject to} \quad \sum_{r=1}^{|D^{s,d}|} F^{s,d,r} \leq \left|D^{s,d}\right|, \quad \forall s \in N,\ d \in N,\ s \neq d \tag{2}$$

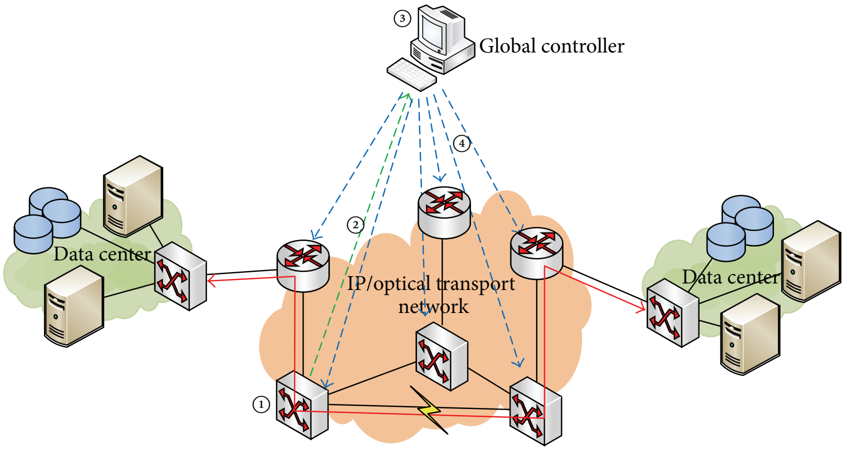
$$\sum_{j:(h,j)\in L} F_{h,j}^{s,d,r} - \sum_{i:(i,h)\in L} F_{i,h}^{s,d,r} = \begin{cases} F^{s,d,r} & h = s \\ 0 & h \neq s, d \\ -F^{s,d,r} & h = d \end{cases} \quad \forall s \in N,\ d \in N,\ s \neq d,\ h \in N,\ r \in D^{s,d} \tag{3}$$

$$\sum_{s=0}^{N}\sum_{d=0,d\neq s}^{N}\sum_{r=1}^{|D^{s,d}|} F_{i,j}^{s,d,r}\times b^{s,d,r} + \sum_{s=0}^{N}\sum_{d=0,d\neq s}^{N}\sum_{r=1}^{|D^{s,d}|} F_{j,i}^{s,d,r}\times b^{s,d,r} \leq B_{i,j} \quad \forall (i, j) \in L \tag{4}$$

$$\sum_{m:(i,m)\in L} F_{i,m}^{s,d,r} \leq 1 \quad \forall s \in N,\ d \in N,\ s \neq d,\ i \in N,\ r \in D^{s,d} \tag{5}$$

$$\sum_{m:(m,j)\in L} F_{m,j}^{s,d,r} \leq 1 \quad \forall s \in N,\ d \in N,\ s \neq d,\ j \in N,\ r \in D^{s,d} \tag{6}$$

$$\sum_{m:(m,s)\in L} F_{m,s}^{s,d,r} = 0 \quad \forall s \in N,\ d \in N,\ s \neq d,\ r \in D^{s,d} \tag{7}$$



① OTN switch detects failure
② Switch informs controller through OpenFlow message
③ Controller obtains the flow list by checking TED and calculates backup path
④ Controller configures routers and switches to exchange data traffic from work path to backup path

Figure 2: Process of bulk restoration.

$$\sum_{m:(d,m)\in L} F_{d,m}^{s,d,r} = 0 \quad \forall s \in N,\ d \in N,\ s \neq d,\ r \in D^{s,d} \tag{8}$$

$$\sum_{(i,j)\in L} F_{i,j}^{s,d,r} \geq F^{s,d,r} \quad \forall s \in N,\ d \in N,\ s \neq d, (i,j) \in L \tag{9}$$

$$R_{\max} \geq \frac{\sum_{s=0}^{N} \sum_{d=0,d\neq s}^{N} \sum_{r=1}^{|D^{s,d}|} \left(F_{i,j}^{s,d,r} \times b^{s,d,r}\right)}{B_{i,j} + \delta} \quad \forall (i,j) \in L \tag{10}$$

$$F^{s,d,r}, F_{i,j}^{s,d,r} \in [0,1] \quad \forall s \in N,\ d \in N,\ s \neq d, (i,j) \in L,\ r \in D^{s,d}. \tag{11}$$

Equation (1) represents the triple-objective function. We utilize two constants α and β to control the precedence of the three addends. The first term minimizes the total bandwidth of unrecovered flows; the second term means to allocate backup paths with the aim of reducing maximum link resource occupation, thus minimizing resource contention between subsequent requests; the third term minimizes the total resource occupation.

Equation (2) ensures the number of backup paths to be less than or equal to recovery requests. Equation (3) ensures the flow conservation for each backup path. Equation (4) ensures that on each link the resource used by backup paths is less than or equal to the available capacity. Since each link is bidirectional, the backup paths in both directions are taken into account. Equations (5) and (6) ensure that a backup path cannot pass the same node more than once. Equations (7) and (8) ensure that a backup path cannot enter the source or exit the destination. Equation (9) ensures that each recovered request has an available backup path. Equation (10) finds the maximum occupancy ratio of all the links where δ is a constant to avoid $B_{i,j}$ being 0. Equation (11) ensures that $F^{s,d,k}$ and $F_{i,j}^{s,d,k}$ are binary values only.

5. Centralized Bulk Load Balance Restoration

The algorithm can be described in 6 steps.

Step 1. Controller collects each recovery request $d^{s,d,r}$ and inserts it into a set $\{D_{s,d,r}\}_{\text{proc}}$.

Step 2. Based on the current network status, controller calculates the K-shortest backup paths of each $d^{s,d,r}$ from $\{D_{s,d,r}\}_{\text{proc}}$ by KSP algorithm [18] and then inserts these k candidate paths into a set $\{P_{s,d,r}^{k}\}$.

Step 3. For each backup path $p_{s,d,r}^{k}$ from $\{P_{s,d,r}^{k}\}$, controller calculates the weight $w_{s,d,r}^{k}$ by (12), where δ is a constant to avoid the denominator being 0. Insert $w_{s,d,r}^{k}$ into a set $\{W_{s,d,r}^{k}\}$:

$$w_{s,d,r}^{k} = \sum_{(i,j)\in L} \left(\frac{F_{i,j}^{s,d,r,k} \times b^{s,d,r}}{\left(B_{i,j} + \delta\right)} \right). \tag{12}$$

Step 4. Controller obtains the minimum one from $\{W_{s,d,r}^{k}\}$, that is, $w_{ms,md,mr}^{mk} = \min\{W_{s,d,r}^{k}\}$, and switches traffic of

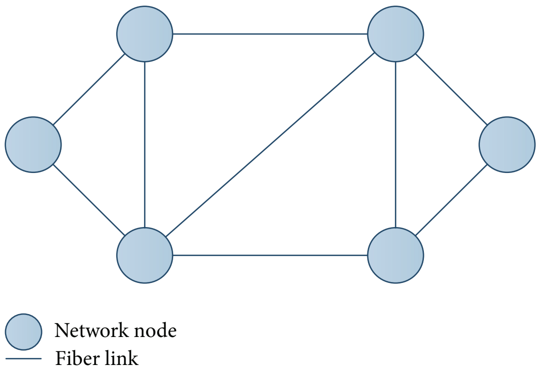


Figure 3: The 6-node 9-link test network.

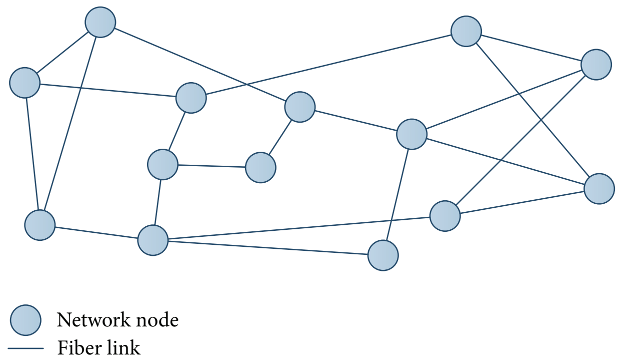


Figure 4: The 14-node 21-link NSFNET network.

$d_{ms,md,mr}$ from work path to backup path $P^{mk}_{ms,md,mr}$. Remove $d_{ms,md,mr}$ from $\{D_{s,d,r}\}_{\text{proc}}$. Update network status.

Step 5. If $\{D_{s,d,r}\}_{\text{proc}}$ is not empty turn to Step 3; otherwise turn to Step 6.

Step 6. The algorithm ends.

6. Simulation Results

We compare centralized bulk load balance restoration (CA-BR) with MILP-based bulk restoration (MILP-BR) on the 6-node 9-link network which is shown in Figure 3. The total bandwidth of each link is 60 G. We assume that flows (uniformly distributed between each node pair) arrive as a Poisson process. Flows are active for an exponentially distributed holding time with a mean of 1 (normalized) unit while 8 flows arrive per unit time. We set the constants in (1) as $\alpha = 50$ and $\beta = 20$ and the constant in (10) as $\delta = 1$, respectively; the value of k in KSP is 4.

After 1600 flows (with bandwidth demands uniformly distributed between 4 Gbps and 5 Gbps) arrival, we freeze the network state and then simulate a failure of each link in the network. Upon each fiber link failure, a set of recovery requests can be obtained. We perform MILP-BR and CA-BR by solving the MILP problem and simulation, respectively. For MILP-BR and CA-BR we record the number of disrupted flows and restoration performance including the number of recovered flows (RF), total resource occupation ratio (TRO), the maximum available resource occupancy ratio (MAO), and computation time (CT). RF is used to analyze restoration efficiency of the two schemes. TRO is defined as resource used during restoration over the total network resource. Available resource occupancy ratio is defined as resource used on one link during restoration over the remaining available resource before restoration on a specific link and MAO is the maximum one of all the links. TRO and MAO are used to analyze the level of resource contention. A big TRO or MRO indicates intense resource contention. Table 1 illustrates the performance of MILP-BR and CA-BR at a load of 64 Erlangs.

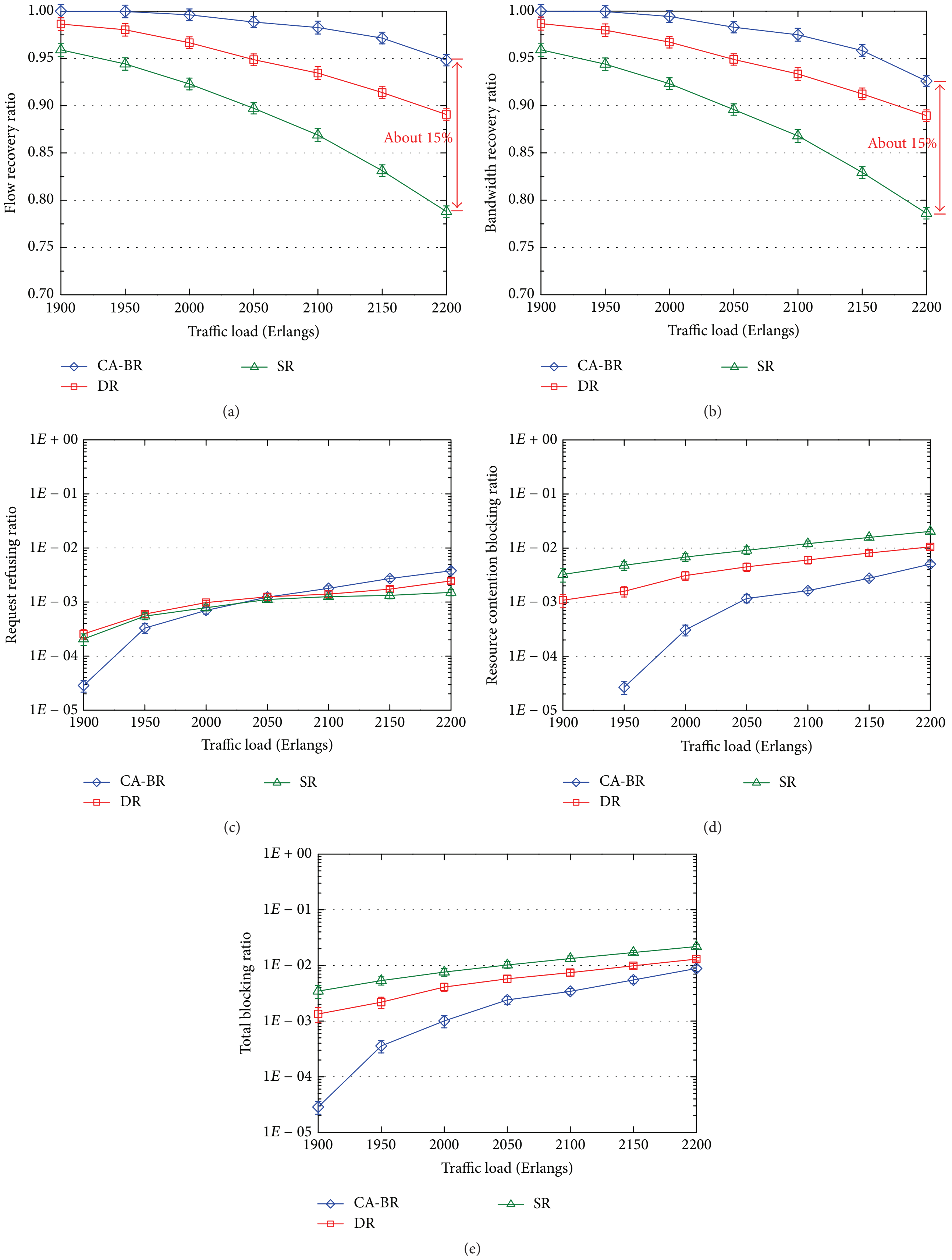
Results show that the RF values of the two schemes are similar. Furthermore, in case of the same RF, sometimes MILP-BR has slightly lower TRO and MAO than CA-BR, while sometimes TRO and MAO of the two schemes are the same. However, MILP-BR takes such a long time that it is unsuitable to be applied in industry. CA-BR performs similarly to MILP-BR and has a much shorter restoration time than MILP-BR. For the above-mentioned reasons, we use CA-BR as the bulk restoration scheme in the following section.

We compare three algorithms by simulation: the proposed scheme CA-BR, dynamic fast restoration by arriving time sequence (DR) introduced in [19], and static restoration by arriving time sequence (SR) widely used in current network restoration [20]. DR selects the shortest available backup path for each disrupted flow based on the current network status. SR uses a fixed backup path for each recovery request that is calculated in advance.

The 14-node 21-link NSFNET network shown in Figure 4 is considered as the test network. There are 10 wavelengths per fiber and the bandwidth of each wavelength is 400 Gbps; that is, the total bandwidth of each fiber link is 4000 Gbps. The add-drop ports are used to drop/add connections to/from router. We set add-drop ports bandwidth half of the OTN line-side bandwidth on each node. We assume that flows (with uniformly distributed source-destination pairs) arrive as a Poisson process and are active for an exponentially distributed holding time. 25 flows arrive per unit time. Bandwidth demands of each flow are uniformly distributed between 5 Gbps and 15 Gbps. 25000 flows are simulated in each experiment; that is, we simulate 1000 unit times. The link failure generation model is also a Poisson process. The failure holding time and arrival interval are 100 and 20 unit times, respectively. The value of k in KSP is 4.

We evaluate CA-BR, SR, and DR and Figure 5 shows flow recovery ratio, bandwidth recovery ratio, request refusing ratio, and total blocking ratio. For each simulation point, we repeat the simulation program 500 times with different random input flow requests and average the results in order to avoid errors caused by random data. Each point is plotted with the reached confidence interval at 95% confidence level.

Flow recovery ratio is defined as the number of recovered flows over the total number of arrived flows. The CA-BR scheme always has the highest flow recovery ratio because it preferentially processes the recovery request which has the least increase on resource contention so that the resource



(a)

(b)

(c)

(d)

(e)

Figure 5: Simulation results: (a) flow recovery ratio, (b) bandwidth recovery ratio, (c) request refusing ratio, (d) resource contention blocking ratio, and (e) total blocking ratio.

Table 1: Simulation results of MILP-BR and CA-BR.

Failed link	Disrupted flows	MILP-BR				CA-BR			
		RF	TRO	MAO	CT	RF	TRO	MAO	CT
(A, B)	4	4	0.166342	0.761468	50 ms	3	0.111414	0.761468	<1 ms
(A, C)	8	6	0.206632	0.965035	4 s 7 ms	5	0.204887	0.895105	<1 ms
(B, C)	10	8	0.254013	0.991561	4 m 14 s 59 ms	8	0.254013	0.991561	<1 ms
(B, D)	10	4	0.142905	0.807692	5 m 7 s 13 ms	4	0.144928	0.865385	<1 ms
(C, D)	11	6	0.18816	0.994012	17 m 28 s 7 ms	5	0.14656	0.881119	1 ms
(C, E)	11	5	0.143421	0.816514	5 m 20 s 23 ms	5	0.143421	0.816514	1 ms
(D, E)	5	4	0.134467	0.865385	53 ms	4	0.139876	0.875	<1 ms
(D, F)	11	10	0.386004	0.957784	40 s 3 ms	8	0.291324	0.836538	1 ms
(E, F)	2	2	0.0832603	0.840708	40 ms	2	0.0832603	0.840708	<1 ms

contention can be kept at a relatively lower level and subsequent requests have much higher chance to be recovered. SR has the lowest flow recovery ratio because it is a static scheme.

The bandwidth recovery ratio is the amount of bandwidth recovered over the total bandwidth of recovery requests. Results show that CA-BR always provides the best performance thanks to the resource contention avoiding mechanism. It reveals that CA-BR uses network resource more efficiently so as to improve network throughput. SR always provides the worst performance.

Request refusing ratio is the number of requests which are refused on arriving due to the insufficient resources over the total request number. The requests which are blocked by link failure are not counted in. Figure 5(c) shows that all the tested schemes provide very low and similar request refusing ratio. The CA-BR scheme provides the lowest request refusing ratio for lower loads. However, request refusing ratio of CA-BR significantly increases with the traffic load and becomes the highest under higher loads. It is because CA-BR uses extra resources to recover more requests which can be seen from Figures 5(a) and 5(b).

Resource contention blocking ratio is the number of requests blocked due to the resource contention during restoration over the total request number. Figure 5(d) shows that the CA-BR scheme always keeps very low resource contention blocking ratio. At low traffic load, this is particularly significant (no request is blocked due to resource contention at 1900 Erlangs). However, DR and SR provide higher resource contention blocking ratio, even at low traffic load. It reveals that CA-BR can signally reduce the impact of resource contention during restoration.

Total blocking ratio is the number of requests including refused requests and unrecoverable flow over the total request number; that is, the requests which are refused on arriving and the disrupted requests which can not be recovered are counted in. Figure 5(e) shows that the CA-BR scheme provides the lowest total blocking ratio. For all schemes the total blocking ratio increases with the traffic load. However, CA-BR always keeps total blocking ratio at a low level because of the resource contention avoiding. Figures 5(c), 5(d), and 5(e) reveal that CA-BR slightly reduces the requests refusing rate while increasing the network survivability and overall improves network throughput.

7. Conclusion

We propose a global bulk restoration scheme which collects all the flow recovery demands caused by a link failure and schedules backup paths synchronously. The idea of load balance is introduced while scheduling the backup paths, to alleviate the resource contention between recovery requests thus enhancing the network survivability. We propose both an MILP formulation and a heuristic algorithm to find an optimal solution and a near-optimum solution, respectively. The latter is compared against traditional sequential algorithms. Simulation results show that for the chosen NSFNET reference network the proposed scheme can significantly improve both flow recovery ratio and bandwidth recovery ratio by about 15%.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This study is supported by the National Natural Science Foundation of China (no. 61501054), the Innovative Research Fund of Beijing University of Posts and Telecommunications (2015RC16), and the Fund of State Key Lab of Information Photonics and Optical Communications (IPOC2015ZT07), China.

References

[1] Z. Xiaoxue, V. Vijay, K. Bikash, K. Valey, and H. Tad, "The prospect of inter-data-center optical networks," *IEEE Communications Magazine*, vol. 51, no. 9, pp. 32–38, 2013.

[2] S. Kipp, "Bandwidth growth and the next speed of ethernet," in *Proceedings of the North American Network Operators Group Meeting*, 2012.

[3] M. Kodialam and T. V. Lakshman, "Integrated dynamic IP and wavelength routing in IP over WDM networks," in *Proceedings of the 20th Annual Joint Conference on the IEEE Computer and Communications Societies (IEEE INFOCOM '01)*, pp. 358–366, IEEE, Anchorage, Alaska, USA, April 2001.

[4] Z. Zhou, T. Lin, K. Thulasiraman, G. Xue, and S. Sahni, "Cross-layer network survivability under multiple cross-layer metrics,"

Journal of Optical Communications and Networking, vol. 7, no. 6, pp. 540–553, 2015.

[5] J. Ahmed, C. Cavdar, P. Monti, and L. Wosinska, “Hybrid survivability schemes achieving high connection availability with a reduced amount of backup resources,” *Journal of Optical Communications & Networking*, vol. 5, no. 10, pp. A152–A161, 2013.

[6] R. Ghimire, S. Mohan, M. Leary, and T. Tidwell, “Concurrent multi-layer restoration scheme for GMPLS based WDM networks,” in *Proceedings of the 2nd International Symposium on Advanced Networks and Telecommunication Systems (ANTS ’08)*, pp. 1–3, Mumbai, India, December 2008.

[7] A. Castro, R. Martínez, R. Casellas et al., “Experimental assessment of bulk path restoration in multi-layer networks using PCE-based global concurrent optimization,” *Journal of Lightwave Technology*, vol. 32, no. 1, pp. 81–90, 2014.

[8] A. Castro, L. Velasco, J. Comellas, and G. Junyent, “Dynamic restoration in multi-layer IP/MPLS-over-flexgrid networks,” in *Proceedings of the 9th International Conference on the Design of Reliable Communication Networks (DRCN ’13)*, pp. 155–162, Budapest, Hungary, March 2013.

[9] S. Ruepp, N. Andriolli, J. Buron, L. Dittmann, and L. Ellegaard, “Restoration in all-optical GMPLS networks with limited wavelength conversion,” *Computer Networks*, vol. 52, no. 10, pp. 1951–1964, 2008.

[10] M. Jarschel, T. Zinner, T. Hoßfeld, P. Tran-Gia, and W. Kellerer, “Interfaces, attributes, and use cases: a compass for SDN,” *IEEE Communications Magazine*, vol. 52, no. 6, pp. 210–217, 2014.

[11] S. Gringeri, B. Basch, V. Shukla, R. Egorov, and T. Xia, “Flexible architectures for optical transport nodes and networks,” *IEEE Communications Magazine*, vol. 48, no. 7, pp. 40–50, 2010.

[12] H. Kim and N. Feamster, “Improving network management with software defined networking,” *IEEE Communications Magazine*, vol. 51, no. 2, pp. 114–119, 2013.

[13] T. S. Eugene and Z. Cai, *Maestro: Achieving Scalability and Coordination in Centralizaed Network Control Plane*, Dissertations & Theses-Gradworks, 2012.

[14] OpenFlow Switch Specification, http://archive.openflow.org/documents/openflow-spec-v1.1.0.pdf.

[15] X. Zhang, F. Shen, L. Wang, S. Wang, L. Li, and H. Luo, “Two-layer mesh network optimization based on inter-layer decomposition,” *Photonic Network Communications*, vol. 21, no. 3, pp. 310–320, 2011.

[16] G. Li, D. Wang, J. Yates, R. Doverspike, and C. Kalmanek, “IP over optical cross-connect architectures,” *IEEE Communications Magazine*, vol. 45, no. 2, pp. 34–39, 2007.

[17] B. Rajagopalan, D. Pendarakis, D. Saha, R. S. Ramamoorthy, and K. Bala, “IP over optical networks: architectural aspects,” *IEEE Communications Magazine*, vol. 38, no. 9, pp. 94–102, 2000.

[18] D. Eppstein, “Finding the k shortest paths,” *SIAM Journal on Computing*, vol. 28, no. 2, pp. 652–673, 1998.

[19] A. Fumagalli and L. Valcarenghi, “IP restoration vs. WDM protection: is there an optimal choice?” *IEEE Network*, vol. 14, no. 6, pp. 34–41, 2000.

[20] O. Gerstel, C. Filsfils, T. Telkamp et al., “Multi-layer capacity planning for IP-optical networks,” *IEEE Communications Magazine*, vol. 52, no. 1, pp. 44–51, 2014.

Advances in Multimedia
The Scientific World Journal
International Journal of Distributed Sensor Networks
Journal of Industrial Engineering
Applied Computational Intelligence and Soft Computing
Advances in Fuzzy Systems
Modelling & Simulation in Engineering
Journal of Computer Networks and Communications
Advances in Artificial Intelligence
Advances in Computer Engineering
International Journal of Computer Games Technology
International Journal of Biomedical Imaging
Advances in Artificial Neural Systems
Advances in Software Engineering
Journal of Robotics
Advances in Human-Computer Interaction
Computational Intelligence and Neuroscience
International Journal of Reconfigurable Computing
Journal of Electrical and Computer Engineering