The Journal of Supercomputing (2021) 77:2979–2997
https://doi.org/10.1007/s11227-020-03383-y

Scalable design and algorithm for science DMZ by considering the nature of research traffic

Chankyun Lee[1] · **Minseok Jang**[1] · **Minki Noh**[1] · **Woojin Seok**[1]

Published online: 13 July 2020

Abstract
This paper explores the nature of scientific research traffic on the Korea research environment open network. Based on these investigations, we propose a scalable design and algorithm for the science demilitarization zone (DMZ). The proposed design allows users to share a data-transfer node (DTN), which is essential but costly equipment in the Science DMZ. The proposed iterative greedy algorithm attempts to minimize the peak traffic of the shared DTNs. By considering state-of-the-art DTN and practical research traffic, the proposed design and algorithm achieve up to 79% capital expenditure (CAPEX) reduction from that of a reference design of the Science DMZ where a DTN is allocated per user. The proposed algorithm achieves a 5-order-of-magnitude reduction in computation time at the cost of acceptable CAPEX overheads compared to those of the minimum-CAPEX solutions.

Keywords Research data-transfer network · Science demilitarization zone · Data-transfer node · Network dimensioning · Scalability · Heuristic algorithm

1 Introduction

A high degree of complexity in big-science research requires close collaborations between researchers [1], where the performance of the data-transfer network strongly affects research progress. The scientific research data flows show the unique nature which is far different from that of commercial flows [2]. In the scientific research network, a limited number of researchers transfer a huge amounts of research data. The majority of research flows are terabyte- and petabyte-scale experiments and observation data, which are sensitive to networking performance statistics, such as throughput, packet loss, and delays. However, in the commercial general-purpose network, kilobyte- and megabyte-scale data of diverse applications from public

✉ Chankyun Lee
chankyunlee@kisti.re.kr

[1] Advance KREONET Center, Korea Institute of Science and Technology Information, 245 Daehakro, Yuseong, Daejeon, South Korea

users comprise the major flow. To serve such commercial flows efficiently, generality is regarded as one of the most important virtues in the general-purpose network. Complex deployments of diverse networking devices in the general-purpose network introduce this generality at the cost of functional overhead and redundancy, which can cause performance degradation. An unprecedented increase in traffic in the big-science research communities and aforementioned differences between commercial and research flows raise a fundamental question: Can the current general-purpose networking technology be optimized for the nature of big-science research data?

The Energy Sciences Network (ESnet) addresses this question by developing the science demilitarization zone (DMZ) [3, 4]. The Science DMZ overcomes physical limitations in the general-purpose network by providing a friction-free network path for scientific research data transfer. The use of the Science DMZ is limited to authorized researchers. To address the nature of research flow, the Science DMZ is physically separated from the general-purpose network and commercial applications are prohibited in the Science DMZ. Its authorized users and physical isolation enable the Science DMZ to form a trustworthy closed network, thus allowing firewalls to be avoided, as they are major obstacles to achieving a high throughput in the network [2]. The access control list (ACL) [5] provides security in the Science DMZ. To do so, the IP address of the authorized researchers should be registered with the ACL in advance. Similarly, to avoid any performance degradation of the networking equipment, it is recommended that the amount of networking equipment is minimized through the end-to-end path in the Science DMZ [2]. The Science DMZ deploys data-transfer nodes (DTNs) as end devices, which directly affect the performance of the end-to-end data transfer. The role of a DTN is restricted to research data transfer only, so all the computing resources concentrate on data transfer [4]. The transport control protocol (TCP) is the most widely used protocol on the Internet, where the upper bound of a throughput is expressed as MSS/RTT $\sqrt{L}$, where MSS, RTT, and L are the maximum segment size, round trip time, and packet loss, respectively [6]. The friction-free path in the Science DMZ effectively minimizes L and RTT. To maximize the upper bound of TCP throughput, a large MSS is preferred in the Science DMZ, such as Ethernet jumbo frame [7]. Burst flows require a large size of buffer in networking equipment, which can cause a buffer-bloat problem [8]. However, since the well-optimized Science DMZ guarantees throughput close to its physical capacity [2], we regard the buffer-bloat problem in the Science DMZ negligible.

1.1 Related work

The authors of [4] germinate a paradigm of the Science DMZ in terms of architecture, system configuration, monitoring systems, and security issues. The authors of [2] provide a tutorial for Science DMZ technology by comparing it to an enterprise network in terms of network flows, switch features, transport layer protocols, security issues, data-transfer applications, monitoring applications, and virtualization technologies. The author of [9] addresses firewall issues and security options of the Science DMZ. A software-defined network (SDN)-based

programmable policy engine and security functions for the Science DMZ are developed in [10] and [11], respectively. The authors of [12] present an SDN-enabled quality of service (QoS)-guaranteed big-data-transfer network. Science DMZ implementation experiences to support e-science in Brazil are shared in [13]. Multipath TCP (MPTCP) is a promising transport technology that can introduce higher throughput and robustness than single-path TCP [14, 15]. The Science DMZ can take advantage of MPTCP in the case of path failure, at the cost of resource reservation. The capacity of end devices is a bottleneck in big-data transmission over the present Science DMZ [16]. If future technological development of silicon electronics dramatically improves the capacity-per-cost value of end devices in the Science DMZ, we can expect further throughput gain from MPTCP in the Science DMZ environment.

Scalability is an important performance measure, especially in collaborative research networks. The authors of [17] analyze the scalability of a FiWi network using the network capital expenditure (CAPEX). The scalability of network control is evaluated by the computation time of the network-dimensioning algorithms in [18]. However, scalability has been overlooked in previous studies on Science DMZ. A scalability of Science DMZ strongly depends on the DTNs, since a DTN is a costly resource and requires consistent efforts for managing and monitoring. Therefore, this paper measures a scalability of Science DMZ by calculating total CAPEX for DTNs in the Science DMZ. A commercially available computing machine is recommended for a DTN, unless bandwidth of the Science DMZ infrastructure exceeds capacity of the machine. Price and performance comparisons of up-to-date computing machines are illustrated in Fig. 1. We plot the values from [19] and create an exponential fitting function. CAPEX evaluations in this paper are carried out assuming the fitting function in Fig. 1 for a DTN. Please note that the absolute values in Fig. 1 can vary with electronic technology

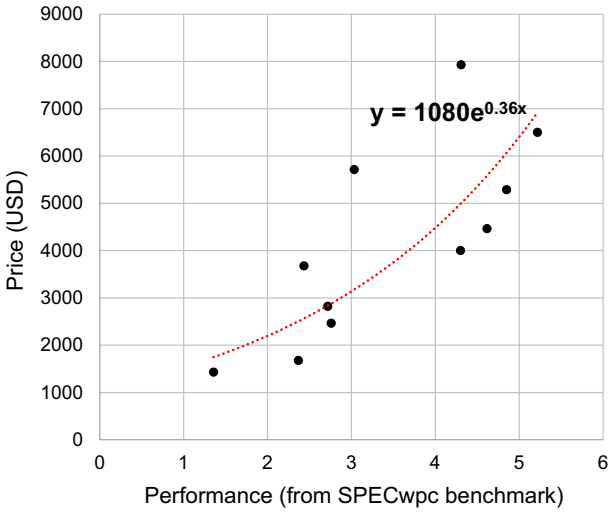

Fig. 1 Price–performance comparison of state-of-the-art computing machines

development and the market environment. However, as long as the fitting function maintains its exponential trend, the methodologies and contributions of this paper remain valid.

The rest of this paper is organized as follows: Sect. 2 observes practical research traffic in Korea research environment open network (KREONET) and investigates the nature of the research traffic, such as a heavy-tail distribution property. Based on the extensive studies of research traffic, Sect. 3 proposes a scalable design of Science DMZ with shared DTN and defines a network dimensioning problem associated with the design. Section 4 suggests an iterative greedy heuristic algorithm for the dimensioning problem in the proposed design of Science DMZ. Section 5 evaluates scalabilities of the proposed design and algorithm by means of CAPEX for DTNs and algorithm computation time, respectively. Finally, Sect. 6 concludes this paper.

2 Nature of research traffic

Korea Institute of Science and Technology Information has been operating KREONET since 1988. KREONET is a unique networking infrastructure specialized for practical scientific research data transfer in South Korea [20]. We have monitored the research traffic of 25 laboratories over KREONET from July 25 to September 25, 2019. The laboratories are representative research groups, who engage in collaborative research for significant big-science problems. Figure 2 depicts the inbound and outbound research traffic of six randomly selected laboratories located in different cities in South Korea. The research traffic of three astronomy laboratories is illustrated in Fig. 2a–c. Figure 2d, e, f shows climate, high energy, and supercomputing research laboratories, respectively. The amount of traffic in Fig. 2 is calculated as the cumulative amount of traffic (bits) over a 5-min interval, divided by 300 s and plotted in 5-min intervals. Note that the traffic data in Fig. 2 are from randomly

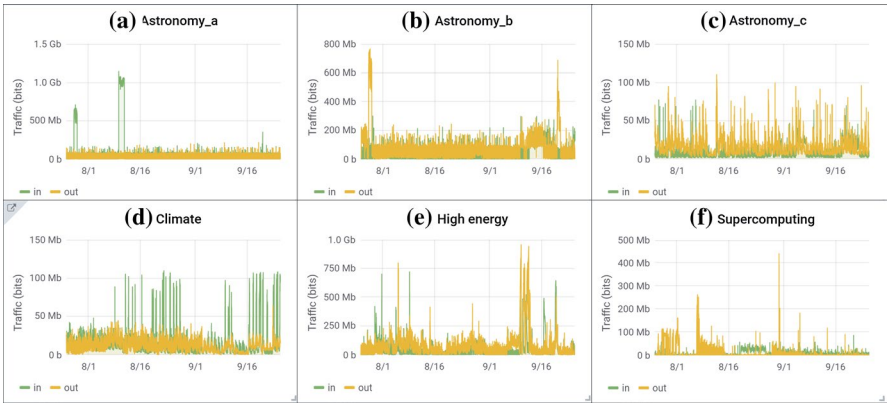

Fig. 2 Time series of amount of traffic for six research laboratories located in South Korea. (Color available online)

selected research laboratories and thus do not represent the nature of traffic in any specific research field.

As shown in Fig. 2a, Astronomy_a shows two huge amounts of inbound traffic at July 28 and August 10, respectively. One can also find a huge amount of outbound traffic from Astronomy_b in Fig. 2b. Because the traffic amount and date coincide between the first inbound of Astronomy_a and outbound of Astronomy_b, we can interpret that there was a data transfer in astronomy community, from Astronomy_b to Astronomy_a at July 28. Both inbound and outbound traffic show a periodicity sawtooth-like pattern with some burstiness in Astronomy_c. As shown in Fig. 2d, outbound traffic from the climate laboratory is relatively stable, where a small amount of traffic continuously flows out from the laboratory. However, high variations in inbound traffic are observed with an order of magnitude difference between peak and average. Burstiness is observed in both the inbound and outbound traffic of the high-energy research group in Fig. 2e. The inbound traffic of a supercomputing research laboratory is relatively small, whereas its outbound traffic fluctuates highly, as illustrated in Fig. 2f. The research traffic over KREONET widely shows burstiness. Specifically, burst transactions rarely present time correlations between different research groups.

Initiated by the monumental works in [21], the burstiness of network traffic is observed [22–24] and analyzed [25–28]. The burstiness of the network traffic requires careful considerations in network dimensioning. The burstiness of network traffic over a wide range of time scale exhibits heavy-tail distribution which decreases with a hyperbolical function, namely slower than exponential function [28]. Accordingly, the tail in the heavy-tail distribution is heavier than that in the exponential distribution. Because a cumulative density function (CDF) of an exponential distribution is expressed as $1-e^{-\lambda x}$ with a random variable x and mean value λ, taking a logarithm of the complementary CDF (CCDF) of the exponential distribution produces a linear graph with a slope of $-\lambda$. In a log–log plot of CCDF, a heavy-tail distribution forms a linear graph [28]. Figure 3 describes the log-linear and log–log plots of the CCDF for incoming traffic of Astronomy_a illustrated in Fig. 2a. A log-linear plot of the CCDF of Astronomy_a incoming traffic shows slow decrease in 98% of traffic and sudden sharp decrease in the top 2% of traffic. Similarly, a combination of two linear graphs fits the log–log plot of CCDF with a slow linear slope (−0.23) for 98% of traffic and a fast-linear slope (−27.64) for

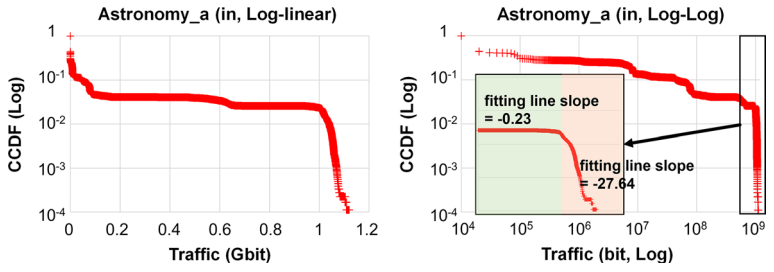

Fig. 3 Log-linear and log–log plots for CCDF of traffic in Astronomy_a

top 2% incoming traffic of Astronomy_a. "Appendix" summarizes the log-linear and log–log plots of the research traffic for all six research laboratories shown in Fig. 2.

Table 1 summarizes the analysis results of the research traffic shown in Fig. 2. We analyze the average value, maximum value, ratio of top-20% traffic to total traffic, ratio of the time for 80% of the traffic to the total time, and approximation model. The Pareto distribution is a typical example of a heavy-tail distribution, which holds an 80:20 rule; 80% of the total traffic belongs to 20% of the total time [29]. Degrees of bias of both the incoming and outgoing traffic of Astronomy_a exceed the 80:20 rule: the top 20% of time holds 99% and 96% of total traffic in the incoming and outgoing traffic, respectively. The 80:20 rule fits the incoming traffic of the high-energy laboratory well, whereas it fails for some research traffic, such as that of the climate laboratory. We fit each research traffic with diverse distribution models, namely Exponential, Pareto (shape parameter, scale parameter), Lognormal (mean, variance), and Weibull (shape parameter, scale parameter). Table 1 summarizes the best approximation model for each type of research traffic based on the goodness-of-fit of the Kolmogorov–Smirnov test [30]. From the best approximation model, for example, we can interpret that the outgoing traffic of Astronomy_a shows heavy-tail distribution, as the Weibull distribution exhibits heavy-tail property when its shape parameter is lower than 1 [31]. The fittings are determined by EasyFit Software [32].

Table 1 Summary of traffic properties for the six laboratories

		Avg. (Mbit)	Max (Mbit)	$\frac{\text{Traffic}_{\text{Top20\%}}}{\text{Traffic}_{\text{Total}}}$ (%)	$\frac{\text{Time}_{\text{80\% traffic}}}{\text{Time}_{\text{Total}}}$ (%)	Approximation model
Astronomy_a	In	42	1141	99	4	Lognormal (12.07, 10.8)
	Out	7	215	96	8	Weibull (0.33, 8.3×10^5)
Astronomy_b	In	22	299	91	15	Pareto (0.26, 1.8×10^4)
	Out	46	766	72	27	Weibull (0.61, 3.6×10^7)
Astronomy_c	In	9	77	53	48	Lognormal (15.59, 0.69)
	Out	14	110	43	57	Lognormal (16.26, 0.38)
Climate	In	14	110	57	45	Lognormal (15.93, 1.02)
	Out	10	63	41	54	Weibull (1.34, 1.1×10^7)
High energy	In	34	719	79	20	Weibull (0.48, 1.4×10^7)
	Out	59	957	68	31	Weibull (0.69, 4.2×10^7)
Supercomputing	In	6	85	72	28	Lognormal (14.71, 1.56)
	Out	8	441	94	8	Pareto (0.46, 1.01×10^5)

3 Design of science DMZ

The Science DMZ consists of a combination of network equipment, such as the DTN, switch, monitoring tool, security appliances, and interconnection of legacy networks [4]. The DTN is an essential component in the Science DMZ for high performance of end-to-end data transfer. Regarding DTN deployment scenario, one simple design of Science DMZ is that using dedicated DTNs, where a DTN is allocated per user, as shown in Fig. 4a. This paper uses the terms "lab" and "user" interchangeably. To introduce higher scalability, we propose a design of the Science DMZ with shared DTN, as shown in Fig. 4b, by tailoring the nature of research traffic. For simplicity, Fig. 4 omits the monitoring tool, security appliances, and interconnection of the legacy network, as they do not differ much in each scenario. There are various end hosts of the Science DMZ, including research laboratory, local host, and supercomputer [4].

3.1 Science DMZ with dedicated DTN

Because the role of the DTN is limited to data transfer only, we assume that the required performance of a dedicated DTN can be calculated as the maximum traffic workload of a user associated with the DTN. By considering the price-performance function $f(x)$ in Fig. 1, the CAPEX for DTNs of a Science DMZ with dedicated DTN ($C_{\text{dedicated}}$) is calculated by the summation of the price of all DTNs:

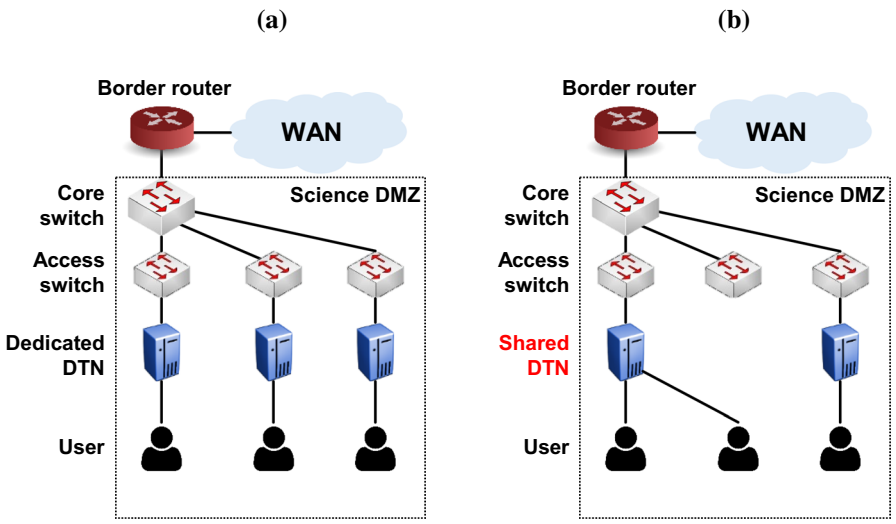

Fig. 4 Designs of Science DMZ with **a** dedicated DTN and **b** shared DTN

$$C_{\text{dedicated}} = \sum_{j=1}^{|U|} f\left(\alpha \max_{t} g_j(t)\right), \tag{1}$$

where α, U, and $g_j(t)$ represent the over-dimensioning factor, set of users, and amount of time-varying traffic of a user attached to the jth DTN, respectively. Please note that $\alpha > 1$ is generally used to deal with future traffic demand, which is uncertain. Under the burst traffic condition, the Science DMZ with dedicated DTN may result in a costly design solution to serve burst traffic for a short duration, while underutilizing the DTN computing resources during the remaining time. Moreover, an increase in $|U|$ requires a huge burden to manage and monitor the DTNs.

3.2 Science DMZ with shared DTN

For the scalable research data-transfer networking, we suggest a design where multiple users share a single DTN, as shown in Fig. 4b. We assume that a user is physically connected to one shared DTN, via a static DTN-user mapping method. In this model, a network dimensioning problem consists of the following mutually coupled problems:

1. Determining the number of shared DTNs
2. Mapping between shared DTNs and users

We denote a set of shared DTNs in the Science DMZ as D. Then, the dimensioning problem determines $|D|$ and creates $|D|$ indistinguishable partitions among $|U|$ distinguishable objects without duplicates. A partition and an object are mapped into a shared DTN and a user in the Science DMZ, respectively. We further define S_j as a set of objects belonging to the jth partition. Accordingly, the dimensioning problem is to find a solution of $(S_1, S_2, \ldots S_{|D|})$ that satisfies the constraints of $\cup_{j\in\{1,\ldots,|D|\}} S_j = U$ and $S_n \cap S_m = \phi$ for all n and m $(n \neq m)$.

For a given dimensioning solution $(S_1^*, S_2^*, \ldots S_{|D^*|}^*)$, the CAPEX for the DTNs of a Science DMZ with shared DTN (C_{shared}) is expressed by the summation of the price of each shared DTN.

$$C_{\text{shared}} = \sum_{j=1}^{|D^*|} f\left(\alpha \max_{t} \sum_{u \in S_j^*} g_u(t)\right) \tag{2}$$

The required performance of a shared DTN is calculated by the maximum of the summation of traffic workloads of users who share the DTN. Again, a fitting function from Fig. 1 can be used for $f(x)$.

One can also imagine dynamic DTN–user mapping for shared DTN where a user is physically connected with multiple DTNs and a traffic engineering algorithm dynamically assigns an appropriate DTN to the user. However, the dynamic mapping design requires additional CAPEX for link installations, which is difficult to estimate. Especially if the DTN deals with sensitive research data, the dynamic

mapping causes an authorization problem. Therefore, to provide accurate and straightforward evaluations of scalability in terms of the network CAPEX, we limit the design of shared DTN to static DTN–user mapping.

4 Dimensioning algorithm for science DMZ with shared DTN

Due to the uncertainty of network traffic, a dimensioning solution derived from the past traffic cannot guarantee the optimality for the future traffic. Although we allow an infeasible assumption that future traffic information is known in advance, finding the minimum-CAPEX solution of a dimensioning problem in the Science DMZ with shared DTN is impractical. The number of permutations of $|U|$ objects to $|D|$ indistinguishable partitions is analyzed in [33]. Because $|D|$ in our dimensioning problem can vary from 1 to $|U|$, the search space to minimize (2) span of

$$\sum_{n=1}^{|U|}\left(\sum_{|S_1|+|S_2|+\cdots+|S_n|=|U|}\frac{|U|!}{|S_1|!|S_2|!\cdots|S_n|!\left(\prod_{K}N_{(k=K)}\right)}\right), \tag{3}$$

where $N_{(k=K)}$ denotes the number of partitions whose cardinality k is equal to K.

The complexity of the network control algorithm is another important measure of its scalability in the network. To find a practical dimensioning solution, Fig. 5 suggests an iterative greedy heuristic algorithm. The algorithm decouples the DTN number decision problem from the DTN–user mapping problem and solves one by one with a greedy manner. These procedures repeat iteratively until an acceptable CAPEX solution is found. The final solution is determined by comparing CAPEXs between iterations.

4.1 DTN number decision algorithm

A trade-off between the number of shared DTNs and required performance of the DTNs manifests a DTN number decision problem important. As a heuristic approach, we propose an iteration-based DTN number decision algorithm. We define d_k as a solution of a DTN number decision problem at the kth iteration. An initial solution is important to achieve a final solution with good performance, as well as a short convergence time in the iteration approach. The initial solution d_0 is calculated as

$$d_0 = \left\lceil \arg\min_{d}\left[df\left(\frac{|U|}{d}\right)\right]\right\rceil. \tag{4}$$

A decision variable d is restricted to the set of natural numbers. Because $f(x)$ in Fig. 1 is differentiable with respect to d, d_0 can be calculated easily. The initial solution in (4) considers the worst case of DTN-user mapping result, where duty cycles of burst transactions of all users in a partition overlap each other.

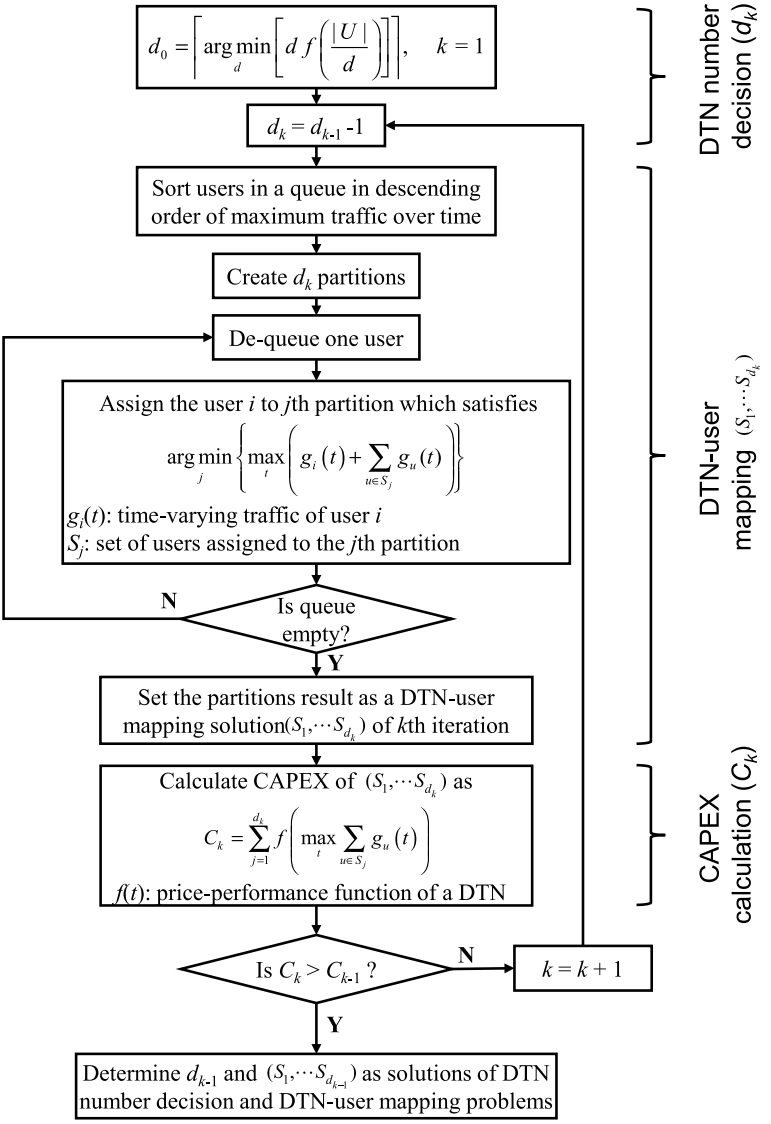

Fig. 5 Flowchart of the proposed algorithm. The algorithm consists of the DTN number decision, DTN-user mapping, and CAPEX calculation functions

Therefore, there is no need to investigate larger than d_0 for shared DTNs at kth iterations ($k > 1$). As shown in Fig. 5, the algorithm calculates total CAPEX for DTNs by decreasing the number of DTN one-by-one, as $d_k = d_{k-1} - 1$.

4.2 DTN-user mapping algorithm

Even though the proposed algorithm reduces computation complexity of the dimensioning problem by decoupling the subproblems, a minimum-CAPEX solution of a DTN–user mapping problem still requires impractical computation efforts for the large-scale networking. Assume that we fix the number of DTN as d_k and find the minimum-CAPEX solution of the DTN–user mapping problem for a given time. Then, the DTN–user mapping solution for minimizing C_{shared} is rewritten as

$$\underset{(S_1,S_2,\ldots,S_{d_k})}{\arg\min} \sum_{j=1}^{d_k} f\left(\alpha \sum_{u \in S_j} g_u\right), \tag{5}$$

subject to

$$\cup_{j\in\{1,\ldots,d_k\}} S_j = U \tag{6}$$

$$S_n \cap S_m = \phi \tag{7}$$

for all n and m ($n \neq m$). g_u is an amount of traffic of user u for a given time. The optimization problem in (5) finds the cost-minimum partitioning solution while satisfying set-partitioning constraints in (6) and (7). Therefore, the problem remaining is the set-partitioning problem [34], which is a well-known non-deterministic polynomial time (NP)-complete problem [35].

To find a practical solution, we propose a greedy heuristic algorithm for the DTN-user mapping problem. The mapping algorithm aims to reduce the peak traffic workloads of each shared DTN. For the given number of DTN (d_k), the algorithm creates d_k partitions and allocates users into the best partition one by one. A user with a higher peak traffic holds a higher priority in the greedy method. To this end, the algorithm sorts users in a temporary queue according to their peak traffic. Then, the algorithm dequeues a user and assigns the best partition for the user, which satisfies the condition

$$\arg\min_j \left\{ \max_t \left(g_i(t) + \sum_{u\in S_j} g_u(t) \right) \right\}. \tag{8}$$

This procedure is repeated until all users are assigned to the appropriate partitions. The partitioning result of kth iteration is defined as $(S_1, S_2, \ldots S_{d_k})$.

4.3 CAPEX calculation and iteration

A final solution of the dimensioning algorithm is selected by CAPEX comparisons between iterations. At the end of kth iteration, CAPEX of the DTN-user mapping solution is calculated as

$$C_k = \sum_{j=1}^{d_k} f\left(\max_t \sum_{i \in S_j} g_i(t) \right). \tag{9}$$

If C_k is less than C_{k-1}, it is highly probable that the further iteration with the smaller number of DTN in the network finds a lower CAPEX solution. Therefore, if $C_k \leq C_{k-1}$, the algorithm runs again for the $(k+1)$th iteration; otherwise, it exits and determine d_{k-1} and $(S_1, S_2, \ldots S_{d_{k-1}})$ as the final solutions for the DTN number decision and DTN–user mapping problems, respectively.

5 Performance evaluations

This section analyzes the CAPEX for DTNs and the computation time of algorithms to evaluate the scalability of the proposed Science DMZ design, as well as the dimensioning algorithm. As a benchmark solution, exhaustive search-based minimum-CAPEX algorithm for the Science DMZ with shared DTN is compared. Because the genetic algorithm is widely used for a set-partitioning problem, we also evaluate the performance of the crossover-based genetic algorithm [36]. In the genetic algorithm, an initial individual is randomly generated as a matrix form ($|D| \times |U|$) with binary elements. We assume that the ith DTN is assigned for the jth user if the element of the matrix at (i, j) is 1. Uniform crossover inside each column in the individual generates offspring where a probability of uniform crossover operator is fixed as 0.5. The fitness function is defined as 1/CAPEX. The algorithm terminates when the number of generations reaches 1000. Because the randomly generated initial solution affects the final solution in the genetic algorithm,

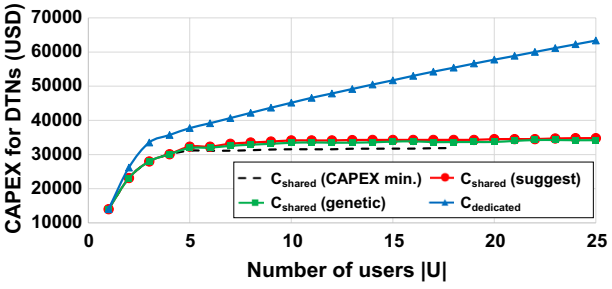

Fig. 6 Comparison of $C_{\text{dedicated}}$ and C_{shared} with respect to the number of users

the performance is averaged over 1000 runs. In this section, we use the monitored traffic in Sect. 2 as the input of the algorithms.

Figure 6 compares $C_{\text{dedicated}}$ and C_{shared} as a function of the number of users for 2 months of research traffic with $\alpha = 2$. The dimensioning problem in the Science DMZ with shared DTN is solved by minimum-CAPEX, suggested, and genetic-based algorithms. Among the 25 laboratories in Sect. 2, a set of users is selected by descending order of their peak traffic workloads. Because of this user selection rule, an increase in $|U|$ results in rapid increases in $C_{\text{dedicated}}$ and C_{shared} when $|U|$ is small. Because the shared DTN can take advantage of traffic multiplexing between users, the increase in C_{shared} is negligibly small when $|U|$ is large. However, the dedicated DTN requires $|D| = |U|$, and thus, $C_{\text{dedicated}}$ constantly increases with respect to the increase in $|U|$. The proposed algorithm requires at most 8% C_{shared} overhead from that of the minimum-CAPEX solution. The minimum-CAPEX solution becomes intractable when $|U|$ is larger than 18. The genetic and suggested algorithms show similar C_{shared} in the entire range of $|U|$. The suggested algorithm requires at most 2% overhead in C_{shared} from that of the genetic algorithm when $|U|$ is 10. For the genetic algorithm, the number of DTNs is determined by the DTN number decision algorithm in Sect. 4.1.

Figure 7 compares computation time of each algorithms for the Science DMZ with shared DTN when $|U| = 15$. By decoupling problems as well as adopting iteration and greedy approaches, the suggested algorithm consumes 0.02 s, whereas the genetic-based heuristic and exhaustive search-based minimum-CAPEX algorithms take 7 and 1200 s, respectively. The suggested algorithm reduces the computation time by 2 and 5 orders of magnitude from those of genetic and the minimum algorithm, respectively, at the cost of negligible CAPEX overhead. The simulation result manifests a high scalability of the shared DTN with suggested dimensioning algorithm both in network design and control complexity. All simulations are performed in MATLAB on a laptop with a 1.99-GHz quad-core CPU and 8-GB memory.

Table 2 summarizes the calculation results of $|D|$ of the suggested and minimum-CAPEX algorithms. Determining α is an important problem which highly affects both network performance and CAPEX. A small value of $|D|$ is preferred as long as α lies in a small value. As shown in Fig. 1, a price of DTN scales exponentially with respect to α, and thus, the suggested algorithm tries to increase

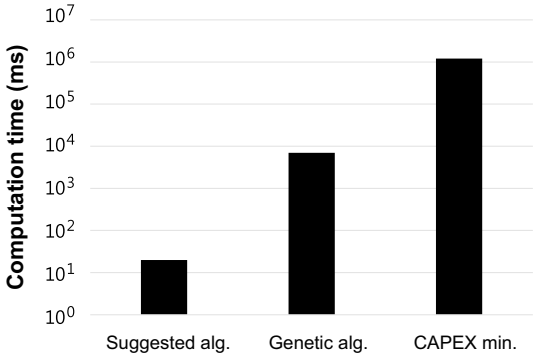

Fig. 7 Computation time comparison between algorithms

Table 2 Calculation results of $|D|$ of suggested and minimum-CAPEX algorithms

$\|U\|$	Algorithm	Dimensioning period	α				
			1	1.5	2	2.5	3
15	Suggested	July 25–August 10, 2019	2	2	2	3	3
		July 25–August 25, 2019	2	2	2	3	3
		July 25–September 10, 2019	1	2	3	4	4
	Minimum-CAPEX	July 25–August 10, 2019	2	2	3	3	3
		July 25–August 25, 2019	2	2	3	3	3
		July 25–September 10, 2019	2	3	3	3	4
25	Suggested	July 25–August 10, 2019	2	2	2	3	5
		July 25–August 25, 2019	2	2	2	3	5
		July 25–September 10, 2019	2	2	4	4	6

$|D|$ when α becomes large. To evaluate the performance of dimensioning algorithms with respect to the period of the input traffic, we consider three periods of input traffic, which are subsets of the 2 months of monitored traffic in Sect. 2. The period is defined as a dimensioning period. As shown in Table 2, the algorithms find slightly different values for $|D|$ under some conditions. We find that the different $|D|$ is the the main reason behind C_{shared} differences between the minimum solution and heuristic solutions (suggested and genetic) in Fig. 6. We leave this for future work. Because the amount of traffic increases with large $|U|$, the suggested algorithm finds equal or larger $|D|$ for larger $|U|$.

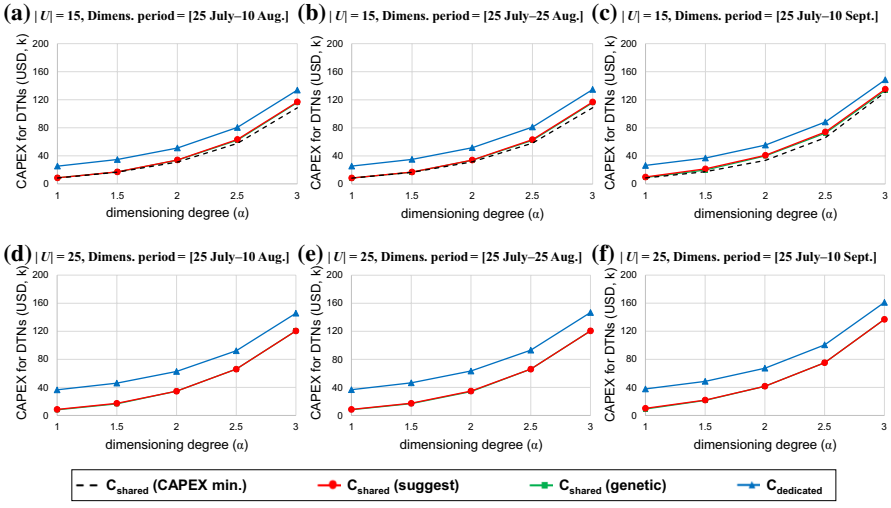

Fig. 8 CAPEX evaluations of Science DMZ designs and algorithms with respect to diverse $|U|$, dimensioning periods, and α

As shown in Fig. 8, $C_{\text{dedicated}}$ and C_{shared} exponentially increase as a function of α because of an exponential fitting function between price and performance of a computing machine in Fig. 1. In the low-α regime, a DTN-user mapping result is critical in neither $C_{\text{dedicated}}$ nor C_{shared}, because a small value is determined for $|D|$. Therefore, C_{shared} of the suggested algorithm is almost equal to that of the minimum-CAPEX algorithm in the low-α regime. An increase in α results in a large value of $|D|$; thus, $C_{\text{dedicated}}$ and C_{shared} strongly depend on the DTN–user mapping result in the high-α regime. Therefore, the difference in C_{shared} between the suggested and minimum-CAPEX algorithms appears in the high-α regime. In all conditions, the differences in C_{shared} between the suggested and minimum-CAPEX algorithms lie between 15% ($\alpha=2$ in Fig. 8c) and 1% ($\alpha=1$ in Fig. 8b). The genetic-based algorithm shows similar (at least 98%) C_{shared} of those of the proposed algorithm, while requiring a 2-order-of-magnitude longer computation time.

Because traffic in the $|U|=15$ case is a subset of that of $|U|=25$ case, $C_{\text{dedicated}}$ in Fig. 8a, b, c are smaller than those in Fig. 8d, e, f, respectively. The suggested algorithm finds an equal or larger $|D|$ for the case of $|U|=25$ than that for $|U|=15$. The large number of $|D|$ introduces a high degree of freedom for distributing users into DTNs. The combination of a high degree of freedom and effective DTN-user mapping algorithm can also lower C_{shared}. Therefore, a shared DTN design with the suggested algorithm effectively suppresses the degree of increment of C_{shared} with respect to the increment of $|U|$, whereas the dedicated DTN fails. The increases in $C_{\text{dedicated}}$ and C_{shared} from $|U|=15$–25 reach up to 45% ($\alpha=1$ in Fig. 8a, d and 3% ($\alpha=3$ in Fig. 8b, e) in dedicated DTN and shared DTN with suggested algorithm, respectively. In Fig. 6, different periods of dimensioning traffic rarely affect trends of CAPEX difference between each design and algorithm.

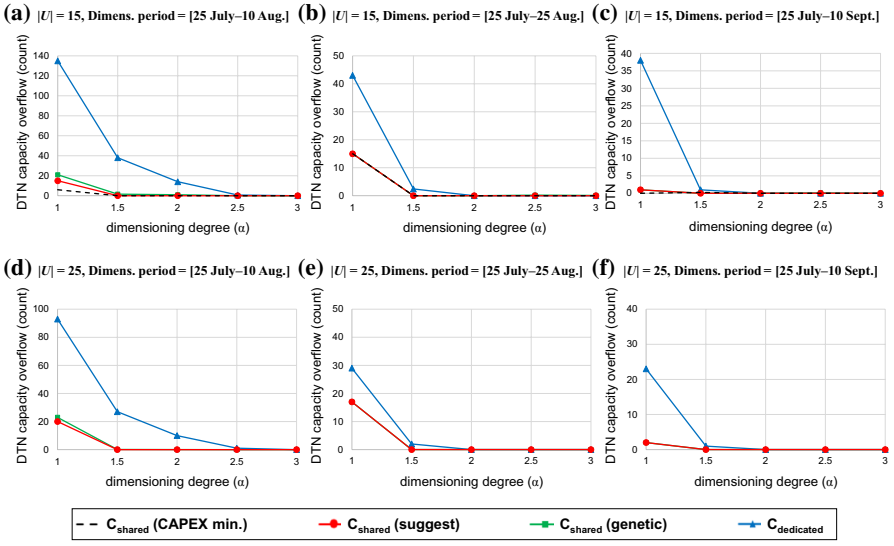

Fig. 9 DTN capacity overflow evaluations of Science DMZ designs and algorithms with respect to diverse $|U|$, dimensioning periods, and α

Owing to the uncertainty of network traffic, it is impossible to forecast the exact amount of future traffic. To effectively provision future traffic, the network dimensioning requires a careful consideration of α. We define a DTN capacity overflow for the case when the amount of traffic workload to a DTN exceeds the capacity of the DTN. To evaluate the DTN capacity overflow, we consider a provisioning scenario in which a dimensioning solution is calculated by the dimensioning period traffic. Then, the remaining traffic out of the 2 months of monitored traffic is used as the provisioning traffic to evaluate the DTN capacity overflow. For example, Fig. 9a shows the number of DTN capacity overflows of the provisioning traffic from August 11 to September 25, under the given dimensioning solution determined by dimensioning traffic from 25 July to 10 August.

As shown in Fig. 9, a higher α dramatically decreases the DTN capacity overflow counts at the cost of CAPEX overhead. Compared with the shared DTN, the dedicated DTN significantly suffers from the overflow counts in the same α condition. Similarly, the required value of α for a zero overflow in the shared DTN is much lower than that of the dedicated DTN. This observation is due to the low time-correlation of burst traffic between users who share a shared DTN. In Fig. 9a, the required values of α for zero overflow are calculated as 1.5 and 2.5 for a shared DTN with the suggested algorithm and dedicated DTN, which correspond to 17,123 and 80,437 USD in Fig. 8a, respectively. Therefore, to satisfy an overflow-free requirement during provisioning, the shared DTN with suggested algorithm reduces 79% CAPEX from that of the distributed DTN. We observe that a few randomly generated initial populations in the genetic algorithm exaggerate the average value of DTN capacity overflows, as shown in Fig. 9a, d.

6 Conclusions

Science DMZ is an optimized networking technology for research data transfer, where a DTN is an essential component to take full advantage of the plentiful network bandwidth provided by the Science DMZ infrastructure. The design of the Science DMZ regarding DTN deployment is an important dimensioning problem in the scalability of scientific research networks, as the DTN is a costly networking component and requires considerable efforts for monitoring and management. This paper proposes a new design of Science DMZ with shared DTN as a promising solution for highly scalable research data-transfer networking. In-depth explorations of the 2-month-long real research traffic of 25 laboratories over KREONET provided interesting observations in the research traffic, including burstiness, heavy-tail distribution, fitness of the 80:20 rule, and low time correlations of burst transactions between different research communities. Based on extensive studies of research traffic, this paper proposes an iterative greedy heuristic algorithm for the shared DTN design, which aims to minimize the peak traffic on the shared DTNs. The proposed algorithm achieves a short computation time by decoupling problems and adopting iterative and greedy approaches. The simulation studies on practical research traffic manifest

the high scalability of the shared DTN using the suggested algorithm, which achieves 2- and 5-order-of-magnitude computation time reductions at costs of acceptable CAPEX overheads from those of genetic-based and minimum-CAPEX solutions, respectively. Specifically, the shared DTN with suggested algorithm saves at most 79% of CAPEX from that of dedicated DTN design under the strict provisioning criteria. This paper leaves studies on the nature of research traffic with respect to time scale, including the self-similarity and scale-free properties, for future studies.

Acknowledgements This research was supported by Korea Institute of Science and Technology Information (KISTI).

Appendix

Log–linear and log–log plots of CCDFs of incoming and outgoing traffic at six laboratories are depicted in Figs. 10 and 11, respectively. Each graph corresponds to the raw traffic shown in Fig. 2.

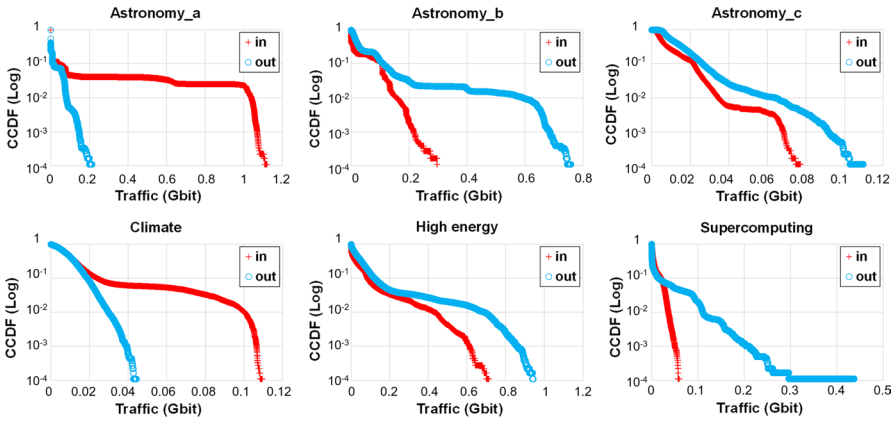

Fig. 10 Log–linear plots for CCDF of research traffic at six laboratories

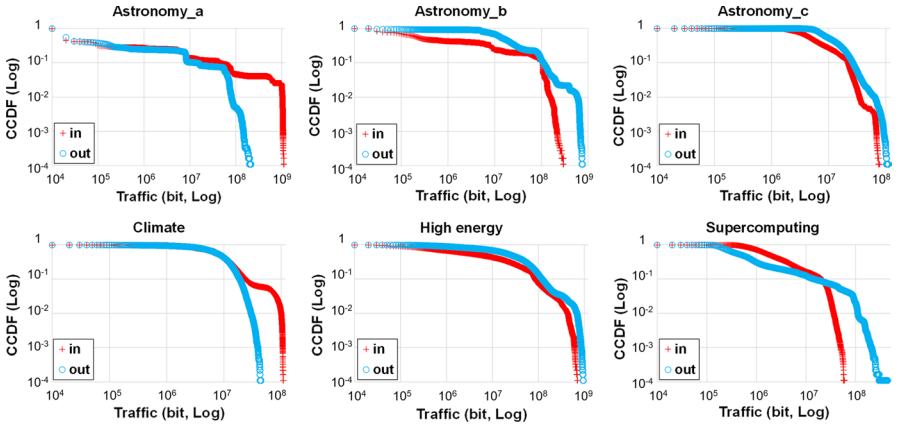

Fig. 11 Log–log plots for CCDF of research traffic at six laboratories

References

1. The EHT Collaboration (2019) First M87 event horizon telescope results. IV. Imaging the central supermassive black hole. Astrophys J Lett 875(1):1–52
2. Crichigno J, Harb E, Ghani N (2019) A comprehensive tutorial on science DMZ. IEEE Commun Surv Tutor 21(2):2041–2078
3. Dart E, Rotman L, Tierney B, Hester M, Zurawski J (2014) The science DMZ: a network design pattern for data-intensive science. Sci Program 22(2):173–185
4. The Energy Sciences Network. https://www.es.net. Accessed 10 Mar 2020
5. Qian J, Hinrichs S, Nahrstedt K (2001) ACLA: a framework for access control list (ACL) analysis and optimization. In: Steinmetz R, Dittman J, Steinebach M (eds) Communications and Multimedia Security Issues of the New Century. IFIP—the International Federation for Information Processing, vol 64. Springer, Boston
6. Mathis M, Semke J, Mahdavi J, Ott T (1997) The macroscopic behavior of the TCP congestion avoidance algorithm. Comput Commun Rev 27(3):67–82
7. Ethernet Alliance (2007) Ethernet Jumbo Frames. http://goo.gl/i6ktnh. Accessed 10 Mar 2020
8. Gettys J (2011) Bufferbloat: dark buffers in the internet. IEEE Internet Comput 15(3):95–96
9. Dart E (2013) Science-DMZ security. Winter Joint Techs, Honolulu, HI
10. Xu C, Li P, Luo Y (2018) A programmable policy engine to facilitate time-efficient science DMZ management. Elsevier Future Gener Comput Syst 89:515–524
11. Chowdhary A, Dixit V, Tiwari N, Kyung S, Huang D, Ahn G (2017) Science DMZ: SDN based secured cloud testbed. In: IEEE Conference on Network Function Virtualization and Software Defined Networks
12. Shah S, Wu W, Lu Q, Zhang L, Sasidharan S, DeMar P, Guok C, Macauley J, Pouyoul E, Kim J, Noh S (2018) AmoebaNet: an SDN-enabled network service for big data science. J Netw Comput Appl 119:70–82
13. Magri D, Carvalho T, Redigolo F, Rojas M, Junior M, Ciuffo L, Dias G, Moura A, Vetter F (2014) Science DMZ: support for e-science in Brazil. IEEE Int Conf e-Sci 2:75–78
14. Barre S, Paasch C, Bonaventure O (2011) MultiPath TCP: from theory to practice. Int Conf Res Netw 444:457
15. Baccarelli E, Scarpiniti M, Momenzadeh A (2018) Fog-supported delay-constrained energy-saving live migration of VMs over multipath TCP/IP 5G connections. IEEE Access 8:42327–42354
16. Science DMZ architecture. https://fasterdata.es.net/science-dmz/science-dmz-architecture/. Accessed 12 June 2020

17. Peralta A, Inga E, Hincapie R (2017) Optimal scalability of FiWi networks based on multistage stochastic programming and policies. IEEE/OSA J Opt Commun Netw 9(12):1172–1183
18. Lee C, Cao X, Yoshikane N, Tsuritani T, Rhee J (2015) Scalable software-defined optical networking with high-performance routing and wavelength assignment algorithms. OSA Opt Express 23(21):27354–27360
19. Digital Engineering. https://www.digitalengineering247.com/article/xi-mtower-pcie-workstation-an-overclocked-performance-champ/. Accessed 10 Mar 2020
20. Korea Research Environment Open Network. https://www.kreonet.net/eng/ Accessed 10 Mar 2020
21. Leland W, Taqqu M, Willinger Wilson D (1994) On the self-similar nature of Ethernet traffic (extended version). IEEE/ACM Trans Netw 2(1):1–15
22. Paxson V, Floyd S (1995) Wide-area traffic: the failure of Poisson modeling. IEEE/ACM Trans Netw 3:226–244
23. Feldmann A, Gilbert A, Willinger W (1998) Data networks as cascades: investigating the multifractal nature of Internet WAN traffic. ACM SIGCOMM Comput Commun Rev 28:42–55
24. Beran J, Sherman R, Taqqu M, Willinger W (1995) Long-range dependence in variable-bit-rate video traffic. IEEE Trans Commun 43(2):1566–1579
25. Crovella M, Bestavros A (1995) Explaning world wide web traffic self-similarity. Tech. Rep. TR-95-015, Boston University, CS Department, Boston, MA 02215
26. Willinger W, Taqqu M, Leland W, Wilson D (1995) Self-similarity in high-speed packet traffic: analysis and modeling of Ethernet traffic measurements. Stat Sci 10(1):67–85
27. Alasmar M, Zakhleniuk N (2017) Network link diemsnioning based on statistical analysis and modeling of real Internet traffic. Cornell Univ Libr, New York
28. Crovella M, Bestavros A (1997) Self-similarity in world wide web traffic: evidence and possible causes. IEEE/ACM Trans Netw 5(6):835–846
29. Pareto V (1897) Cours D'economique politique. Macmillan, New York
30. Massey F (1951) The Kolmogorov–Smirnov test for goodness of fit. J Am Stat Assoc 46(253):68–78
31. Litjens R, Boucherie R (2003) Elastic calls in an integrated services networks: the greater the call size variability the better the QoS. Perform Eval 52:193–220
32. EasyFit. http://www.mathwave.com/. Accessed 15 June 2020
33. Milton J, Arnold J (1995) Introduction to probability and statistics: principles and applications for engineering and the computing science. McGraw-Hill, New York
34. Larsen J (2010) Set partitioning and applications. http://www2.imm.dtu.dk/courses/02735/sppintro.pdf. Accessed 22 June 2020
35. Karmarkar N, Karp R (1982) The differencing method of set partitioning. Computer Science Division, U. of California, Berkeley, CA, USA, Tech. Rep. UCB/CSD 82/113
36. Umbarkar J, Sheth D (2015) Crossover operators in genetic algorithms: a review. ICTACT J Soft Comput 6(1):1083–1092