On the Optimal Provider Selection for Repair in Distributed Storage System with Network Coding

Chengjin Jia¹, Jin Wang²*, Yanqin Zhu³, Xin Wang⁴, Kejie Lu⁵,⁶, Xiumin Wang⁷, and Zhengqing Wen⁸

¹,²,³,⁸Department of Computer Science and Technology, Soochow University, China
⁴School of Computer Science, Fudan University, China
⁵College of Computer Science and Technology, Shanghai University of Electronic Power, China
⁶Department of Electrical and Computer Engineering, University of Puerto Rico at Mayagüez, United States
⁷School of Computer and Information, Hefei University of Technology, China

Abstract. In large-scale distributed storage systems (DSS), reliability is provided by redundancy spread over storage servers across the Internet. Network coding (NC) has been widely studied in DSS because it can improve the reliability with low repair time. To maintain reliability, an unavailable storage server should be firstly replaced by a new server, named new comer. Then, multiple storage servers, called providers, should be selected from surviving servers and send their coded data through the Internet to the new comer for regenerating the lost data. Therefore, in a large-scale DSS, provider selection and data routing during the regeneration phase have great impact on the performance of regeneration time. In this paper, we investigate a problem of optimal provider selection and data routing for minimizing the regeneration time in the DSS with NC. Specifically, we first define the problem in the DSS with NC. For the case that the providers are given, we model the problem as a mathematical programming. Based on the mathematical programming, we then formulate the optimal provider selection and data routing problem as an integer linear programming problem and develop an efficient near-optimal algorithm based on linear programming relaxation (BLP). Finally, extensive simulation experiments have been conducted, and the results show the effectiveness of the proposed algorithm.

Keywords: network coding; distributed storage system; provider selection; routing; linear programming; LP-relaxation

1 Introduction

With the rapid development of big data, the information explosion results in the rapid development of data storage. There are about 5 Exabytes independent

* Corresponding author: wjin1985@suda.edu.cn
information created in 2015 and 8.6 Zettabytes of data center traffic by 2018 [1]. Therefore, many large-scale DSSs, e.g., Google File System [2], Azure [3], are widely used for achieving high reliability by storing the data redundantly over multiple unreliable storage servers.

Reliability is one of the basic requirements for these DSSs that users can get data anywhere anytime. The traditional methods for providing reliability in DSSs include replication and Reed-Solomon codes [4]. In 2000, NC was proposed to increase the throughput of the network, balance network load and so on [5]. It has been proved distributed storage applications can achieve good benefits with NC [6]. When using NC, it keeps the MDS property of erasure code that the original file is divided into \( k \) packets, then encoded into \( n \) coded packets [7]. Users can recover the original file by any set of \( k \) coded packets among \( n \) coded packets. Therefore, more and more researchers pay attention on NC in DSS.

Although NC can improve storage reliability, the data of distributed storage systems is prone to be damaged, such as an outage of the server, invasion by the hackers, disk damaged. To keep the same level of reliability, when a server fails or leaves the system, a new server has to join the system and accesses existing servers to regenerate the lost data, which leads to repair bandwidth consumption and regeneration time. Based on the ideas of NC, the functional minimum storage regeneration (FMSR) codes have been proposed to minimize the repair bandwidth or regeneration time in DSS [8,9].

Although FMSR code can significantly minimize repair bandwidth, it cannot ensure that the regeneration time is minimized. In order to reduce the regeneration time, Li et al. proposed a tree-structured data regeneration in the heterogeneous network [10,11]. Most of current studies focus on obtaining data from multiple surviving servers to regenerate the lost data under the condition that the bandwidth of the path between each servers and the new comer is given. However, each link in physical network may be shared by multiple paths, which means the bandwidth of each link should be shared between different paths. Therefore, in practice, the bandwidth of the routing path from each selected server, i.e., provider, to the new comer may not be achieved.

Next, we introduce an example that shows the effect for regenerating the lost data by selecting a given number of servers as the providers and routing paths from the providers to the new comer. Fig. 1(a) gives the original network topology and includes routers denoted as \( R_j \) and storage servers denoted as \( F_i \). In this example, each server \( F_i \) stores different coded packets of the same file. When \( F_4 \) is unavailable, to keep the same level of reliability, a new server should be installed to replace \( F_4 \) and acquire data packets from multiple available storage servers to regenerate the lost data. Therefore, in this example, we also denote the new comer as \( F_4 \). We assume the number of providers is 3, which is denoted as \( d \) in the rest of the paper and the size of the file is \( M = 300 \text{ Mb} \). With the minimum-storage regenerating code [12,13], each server storages \( \alpha = M/k = 150 \text{ Mb} \) data and \( F_4 \) needs to download \( \beta = \alpha/(d - k + 1) = 75 \text{ Mb} \) data from each provider. The bandwidths of the links range from 30 \( \text{Mbps} \) to 100 \( \text{Mbps} \). As shown in Fig. 1(a), the maximum transmission rate from each storage server to \( F_4 \) is
Fig. 1. Examples for providers and routing paths selection. 30Mbps. Fig. (b), (c) and (d) respectively show different selections of the providers and routing paths. Next, we show different regeneration time of these three kinds of selections as follows:

In Fig. (b), $F_1$, $F_2$ and $F_3$ are selected as providers. Since all the routing paths pass through the link between $R_3$ and $R_5$. The bandwidth is only 30Mbps. Therefore, the transmission rate can be achieved per provider is 10Mbps and the regeneration time is $75/10 = 7.5$ seconds.

In Fig. (c), $F_1$, $F_3$ and $F_5$ are selected as providers. As shown in the figure, two routing paths pass through the link between $R_3$ and $R_5$. The transmission rates for $F_1$ and $F_3$ can be achieved are 15Mbps. Although the maximum transmission rate of $F_5$ can be 30Mbps, the regeneration time depends on the transmission time of $F_1$ and $F_3$. Therefore, the regeneration time is $75/15 = 5$ seconds.

In Fig. (d), $F_1$, $F_3$ and $F_5$ are also selected as providers. In this figure, different routing paths are selected. The transmission rates for $F_1$, $F_3$ and $F_5$ can be achieved are 30Mbps, respectively. Therefore, the regeneration time is $75/30 = 2.5$ seconds.

The example above has well demonstrated that not only the selection of providers but also routing paths can significantly affect the regeneration time,
which motivates the work of this paper. In this paper, we focus on selecting a
given number of providers and deciding the routing paths from them to the new
comer to minimize the regeneration time. The main contributions of this paper
are summarized as follows.

– We define the providers and routing paths selection (PRPS) problem in the
  DSS with NC and model the problem as a mathematical programming.
– For the case that the providers are given, we model the PRPS problem as
  a linear programming. Based on the linear programming, we then formulate
  the optimal provider selection and data routing problem as an integer linear
  programming.
– We develop an efficient near-optimal algorithm based on linear programming
  relaxation (BLP).
– We conduct extensive simulation experiments and the results show the ef-
  fectiveness of the proposed algorithm.

The rest of the paper is organized as follows: in section 2 we introduce the
related works. In section 3 we show network model and notations. We propose
a linear programming (LP) in section 4 to calculate the optimal regeneration
time when the selection of the providers is fixed. In section 5 we formulate
the optimal provider selection problem as a mixed integer linear programming
(MILP) problem and develop an efficient near-optimal algorithm based on LP-
relaxation (BLP). We conduct extensive simulation experiments in section 6.
Finally, we conclude the paper in section 7.

2 Related Works

Hu et al. considered functional minimum storage regenerating (FMSR) codes,
which achieved the minimum repair bandwidth [13]. They mainly proved the
existence of FMSR codes, and proposed a deterministic FMSR code construction
by a large number of rounds of repair.

However, in the practical network, minimizing the repair bandwidth does not
mean to minimize the regeneration time. Based the heterogeneous network, Sun
et al. proposed a tree-structure regeneration model to reduce the regeneration
time, which can reduce significantly the regeneration time [14]. To further reduce
the regeneration time, Wang et al. reconsidered how to solve the problem in the
heterogeneous network [15]. They firstly proved that building an optimal regen-
eration tree was NP-complete and then proposed a novel algorithm to construct
a near-optimal regeneration tree.

In [20, 21], Gong et al. studied the provider selection problem of DSS in
overlay networks, which are represented as a complete graph, to minimize the
regeneration time. Comparing with previous works shown in [20, 21], the main
differences of our paper is summarized as follows: Firstly, routing paths from each
server to the new comer are not given and the general physical network topology
is studied in this paper. Secondly, multiple routing paths can be utilized from
each selected server, i.e., provider, which makes the transmission topology from
the providers to the new comer may not be a tree. Thirdly, each link in the network can be used simultaneously by different flows from different providers and the bandwidth of the link can be shared between them. Moreover, the total bandwidth of each link is assigned to different flows based on the algorithm proposed in Sec. 5 instead of assigning the bandwidth equally to different flows. Finally, by considering the general physical network with heterogeneous bandwidth on each links, we jointly study the provider selection from survival servers, the routing path decision in physical network and the bandwidth assignment of each link together to optimize the regeneration time.

Moreover, considering the practical application, Frank et al. used multiple cloud storage providers and dynamically changed cloud performance with the change of the network to reduce data retrieval time [16]. They firstly proposed a new data distribution mechanism and moved data among different clouds to reduce data retrieval time, which was similar to the regeneration time.

3 Problem Formulation

In this section, we mainly give the definitions of the problem studied in this work. Specifically, we firstly introduce the network model. We then introduce important parameters and variables to be used in the rest of this paper.

3.1 The Network Model

The network consists of routers and storage servers. We assume that each server is connected with a router. Different links in the network may have different bandwidths or transmission rates. In the DSS, the servers distribute in different geographical area of the world [2,3]. In repair phase, the lost data are regenerated by minimum-storage regenerating code with functional repair [7–9], [10]. Moreover, we assume each provider sends the same amount of data packets to new comer (such transmission model has been investigated [13,14]). We also assume the providers for regenerating the data in the network can be controlled to minimize the regeneration time.

We model the network as a directed graph $G = (N, E)$, where $N$ consists of the routers. Since each storage server is connected with a router, we use the router to represent the connect server. We denote the set of routers, each of which connects to a available storage server, as $M_p$, and $M_p \subseteq N$. Moreover, we use $g$ to represent the new comer or the connected router, $g \in N$. We assume the router connected to each server represents it to transmit and receive data. Therefore, in this paper, the problem is equivalent to selecting $d$ routers as providers from the subset of routers $M_p$. Specifically, we want to select $d$ routers to regenerate the lost data, denoted $M_d$, $M_d \subseteq M_p$. Therefore, we also call the subset of routers $M_p$ as servers in the rest of this paper.

\[1\] If a server is connected with multiple routers, we can add a virtual router and this virtual router is connected with multiple routers.
3.2 Notations

In order to facilitate the discussion, we define the parameters and variables as follows:

Parameters used in the rest of the paper are shown as follows:

- $M_p$: The subset of routers, each of which connects a surviving server which can be selected as providers, $M_p = \{n_1, n_2, ..., n_{|M_p|}\}$.
- $g$: The router which connects the new comer.
- $M_n$: The set of the routers in the network, not including $M_p$ and $g$.
- $N$: The set of the routers in the network, $N = M_p \cup M_n \cup \{g\}$.
- $E$: The set of links between routers in the network $G$.
- $N(u)$: The set of downstream neighbor nodes of the router $u$. There exists a link from node $u$ to each node in $N(u)$.
- $N'(u)$: The set of upstream neighbor nodes of the router $u$. There exists a link from each node in $N'(u)$ to node $u$.
- $B_{uv}$: The bandwidth of link $e_{uv}$ from $u$ to $v$ in the network, $e_{uv} \in E$.
- $\beta$: The number of transmission data each provider need to send.
- $d$: The number of the providers.

Decision variables of the problem are shown as follows:

- $\lambda$: The regeneration time.
- $f_{uv}^i$: The traffic load of traffic flow $i$ on link $e_{uv}$. We note that each flow denotes the data transmission from a available server to the new comer $g$.
- $r_i$: The transmission rate routed to $g$ for traffic flow $i$.
- $w_i$: 0-1 variable, which indicates whether a server is selected as a provider.

3.3 Problem definition

In the DSS, server failures are unavoidable. Therefore, it is desirable to regenerate the lost data in order to maintain the system reliability. In this paper, we consider selecting optimal servers as the providers for regenerating the lost data to minimize the regeneration time, which contains the selection of $d$ providers and routing paths from them to new comer, named providers and routing paths selection (PRPS) problem.

4 Problem Formulation

In this section, we firstly assume $d$ providers have been fixed, and give the mathematical formulation to minimize the regeneration time. Specifically, to solve the PRPS problem, we then develop a mixed integer linear programming (MILP) to select $d$ optimal servers as providers.
4.1 The PRPS Problem Formulation with Fixed Providers

When given parameter: \( M_d \), which denotes the set of fixed providers, \( M_d = \{ s_1, s_2, \ldots, s_d \} \), \( M_d \subseteq M_p \). Let \( Q = \{ 1, 2, \ldots, d \} \). We assume there are \( d \) flows, each of which from one provider to new comer. \( s_i \) is equivalent to the source node of traffic flow \( i \), \( i \in Q \). The non-linear programming formulation for the case of fixed providers is shown as follows:

\[
\text{Minimize : } \lambda \\
\text{Subject to: } \\
\lambda \geq \beta / r_i, \forall i \in Q \\
\sum_{v \in N(u)} f^i_{uv} = \sum_{v \in N'(u)} f^i_{vu}, \forall i \in Q, \forall u \in N - \{ s_i \} \\
\sum_{v \in N'(u)} f^i_{vu} - \sum_{v \in N(u)} f^i_{uv} = r_i, \forall i \in Q, u = g \\
0 \leq \sum_{i \in Q} f^i_{uv}, \forall u \in N, \forall e_{uv} \in E \\
0 \leq f^i_{uv}, \forall i \in Q, \forall e_{uv} \in E
\]

The objective (1) is to route the data stored in providers to the new comer through the network such that the regeneration time is minimized. Constraint (2) gives the regeneration time no less than transmission time from each provider to new comer \( g \). \( r_i \) denotes the the transmission rate routed to \( g \) for traffic flow \( i \). Constraints (3)-(5) put network flow constraints between each provider and new comer. Constraint (6) gives the bandwidth constraint for different flows through the same link in the network. Constraint (7) gives value range of variables.

Although we have given the constraints of the problem, the constraint (2) is a non-linear constraint. Next, we try to convert the non-linear constraint to a linear constraint. The objective can be equally converted by the three steps:

- First, the optimal value of the objective is equivalent to minimize the value of \( \max_{i \in Q} (\beta / r_i) \);
- Second, the value \( \max_{i \in Q} (\beta / r_i) \) can be simplified to the value \( \beta \max_{i \in Q} (1 / r_i) \), and is equivalent to \( \beta / \min_{i \in Q} (r_i) \);
- Finally, the objective can be converted to minimize the value of \( \beta / \min_{i \in Q} (r_i) \), which is equivalent to maximize the value of \( \min_{i \in Q} (r_i) \). Suppose the maximum value of \( \min_{i \in Q} (r_i) \) is \( r \), \( \lambda \) can be obtained by \( \beta / r \).

Therefore, the linear programming (LP) can be shown as follows:

\[
\text{Maximize : } r \\
\text{Subject to: } \\
r \leq r, \forall i \in Q
\]
The objective (3)-(7) and Constraint (9) show $r$ is the maximum value of $\min_{i \in Q}(r_i)$. Therefore, the minimum regeneration time is equivalent to $\beta/r$.

4.2 The PRPS Problem

In this section, we assume $d$ providers are not fixed but should be selected from the set of available servers $M_p$. We assume there are $|M_p|$ flows, each of which from one server $n_i$ to the new comer $g$, $n_i \in M_p$. Let $Q = \{1, 2, ..., |M_p|\}$. We then formulate PRPS problem as a mixed integer linear programming (MILP):

\[
\text{Maximize : } r
\]

Subject to:

\[
\begin{align*}
\sum_{v \in N(l)} f_{lv}^i &= \sum_{v \in N'(l)} f_{vl}^i, \forall i \in Q, l = n_i \\
\sum_{v \in N(u)} f_{uv}^i - \sum_{v \in N'(u)} f_{vu}^i &= r_i, \forall i \in Q, u = g
\end{align*}
\]

In above MILP, we use the 0-1 variable $w_l$ denotes whether server $n_l \in M_p$ is selected as a provider. Constraint (13) shows that only $d$ servers can be selected as providers. Constraints (13)-(15) and Constraint (17) give that there has a non-zero transmission rate $r_i$ from server $n_i$ to the new comer $g$ if only if server $n_l \in M_p$ is selected as a provider. Constraint (12) is equivalents to

\[
o_i = \begin{cases} 
  r_i, & \text{if } r_i > 0 \\
  \theta, & \text{if } r_i = 0.
\end{cases}
\]

Let $\theta$ be a sufficiently large number. Consider the objective and Constraint (11),

\[
r = \min_{i \in Q} o_i.
\]

Since $\theta$ is set to be a sufficiently large constant value, i.e., $\theta > \max_{i \in Q} r_i$,
we have $r = \min_{i \in Q \text{ and } r_i > 0} r_i$, which means the objective $r$ is to maximize the minimum transmission rate of the providers. Finally, we can get the minimum regeneration time, which is equivalent to $\beta/r$. 
Algorithm 1 The Effective Algorithm Based on LP-Relaxation (BLP)

Step 1: Solve the LP-relaxation with the objective (10), constrains (11)–(19), (21) to obtain an optimal solution \{w^M_l\}.

Step 2: For each \( l \in M_p \), set \( w^*_l = 1 \) for the \( d \) largest \( w^M_l \) in \( \{ w^M_l | l \in M_p \} \).

Step 3: Set other \( w^*_l = 0 \).

Step 4: Fix the \( d \) selected providers with \( w^*_l = 1 \), solve the LP with the objective (8), constrains (3)–(7), (9) and obtain the value of \( f^i_{uv} \), which denotes the data transmission rate on link \( e_{uv} \) for each provider \( i \).

Step 5: Return the sub-optimal solution: \{w^*_l | l \in M_p \} and \{f^i_{uv} | e_{uv} \in E, i \in Q \}.

5 An Efficient Algorithm for Optimal Providers and Routing Paths Selection

According to the MILP proposed in above section, we can obtain the minimum regeneration time. The PRPS problem can be optimally solved by the proposed MILP formulation when the size of the problem is small. However, when the problem size is large, the computational complexity of MILP is considerably large, which has been proved as NP-hard problem [17]. Therefore, when the problem size is large, we need to develop a novel efficient algorithm to select \( d \) providers and corresponding routing paths.

Next, we propose a novel efficient algorithm based on the LP-relaxation of the proposed MILP, which contains three steps is shown as follows:

1) We replace the constraint (20) by:

\[
0 \leq w_l \leq 1, \forall l \in M_p
\]  

(21)

We obtain the LP-relaxation of the proposed MILP.

2) We solve the obtained LP-relaxation and use \( w^M_l \) to represent the optimal solution, where some values of \( w^M_l \) may not be integers.

3) We select \( d \) servers as providers based on the values of \( w^M_l \). Specifically, Algorithm 1 gives the BLP algorithm. For each variable \( w^M_l \), \( l \in M_p \), the first \( d \) servers, which have first \( d \) highest value, will be selected as the providers, and the corresponding variable \( w^M_l \) is set to 1. Otherwise, it is set to 0.

It is known that LP can be efficient solved. The computational complexity will take \( O(M_p \log M_p) \) for sorting the elements in \( w^M_l \). Therefore, the time complexity of our algorithm is dominated by solving the LP relaxation.

6 Performance Evaluation

In this section, we will give simulation results to compare with the proposed BLP algorithm, a random selection (RS) scheme and the maximum-flow (MF) scheme.

The RS scheme randomly selects \( d \) available servers as the providers. On the other hand, in the MF scheme, we calculate maximum-flow \( H_u \) from each
available server \( u \) to \( g \) individually, \( u \in M_p \). Specifically, for each value \( H_u \), the first \( d \) servers, which have first \( d \) highest value, will be selected as the providers.

Moreover, let \( \lambda_r \) denote the regeneration time achieved by the RS scheme, \( \lambda_m \) denote the regeneration time by MF scheme, and \( \lambda_b \) denote the regeneration time achieved by the BLP algorithm. Compared with the RS scheme, the improvement ratio of the BLP algorithm is defined as \( I_{RS} = (\lambda_r - \lambda_b)/\lambda_r \). On the other hand, the improvement ratio of the BLP algorithm is defined as \( I_{MF} = (\lambda_m - \lambda_b)/\lambda_m \) compared with the MF scheme. For different combination of parameters, we will give the average results showing in the following figures.

### 6.1 Simulation Setup

In this simulation, we assume the nodes in the network are randomly generated in the \( 10 \times 10 \) \( m^2 \) square region, where the nodes denote servers, the routers and the new comer. As we know, the encoding time from each provider and decoding time on the new server are ignored with the reason these operations can be performed with the data transmission at the same time [10].

Moreover, the random network graph \( G \) is generated by the widely used Waxman algorithm [18]. The nodes in \( G \) are a Poisson process mainly with three kinds of variables, denoted as \( a \), \( b \) and \( c \). \( a \) denotes the intensity of the Poisson process. Two nodes are connected by a link with probability \( P(u, v) = be^{-d(u,v)/(c+L)} \), \( d(u,v) \) is Euclidean distance and \( L \) is the maximum distance between any two nodes. In the simulation, we firstly let the set of nodes with serial number in \( \{1, \ldots, D\} \) as \( M_p \), i.e., the set of available servers, and let the node has largest serial number as the new comer, denoted as \( g \). Other nodes in \( G \) are routers.

In our simulations, we have six parameters as follows:

- \( a \), \( b \), \( c \): The parameters for Waxman model.
- \( d \): The number of providers to be selected from \( M_p \).
- \( U \): The bandwidth of the link distributes uniformly in a certain range \( U \);
- \( D \): The number of the servers in the set \( M_p \).

In this section, the file is coded with the redundancy of an \( (n=10, k=2) \)-MDS code, that we can recover the file from any 2 of 10 storage servers. The size of the file is \( M = 1024Mb \) and each server stores \( \alpha = M/k = 512Mb \). Moreover, with the minimum-storage regenerating (MSR) codes [8], each provider needs to send \( \beta \) data to the new comer for regenerating the lost data and \( \beta \) can be calculated as the formula \( \beta = \alpha/(d - k + 1) \), where \( d \) represents the number of providers. Therefore, regeneration time \( \lambda \) can be obtain with the formula \( \beta/r \) in our simulations. We set \( \theta \) as a sufficiently large number when solve the LP relaxation of the MILP. Moreover, we use CPLEX [19] to solve the linear programming in the simulation.
6.2 Simulation Results

In this section, we change different combinations of parameters to compare the RS method, MF method and the proposed BLP algorithm.

In Fig. 2 we set $b=0.4$, $c=0.4$, $D=7$, $d=4$ and $U=[10, 40]$. Fig. 2(a) shows the regeneration time of all algorithms decreases with the increase of $a$. With the increase of $a$, there are more nodes and links in the network, which leads to multiple paths can be selected to routing the data and higher transmission rate can be achieved. Therefore, the regeneration times of the three algorithms decrease. Fig. 2(b) shows our algorithm can achieve lower regeneration time compared with other two schemes. In Fig. 2(b), the improvement ratios of BLP decrease with the increase of $a$ compared with RS and MF. The reasons include that (1) the number of $D$ is fixed, the optimization space is limited; and (2) when multiple paths can be selected to routing the data and higher transmission rate can be achieved with the increase of $a$, the selection of providers has lower impact on the regeneration time. The improvement ratio of BLP still achieves 30% compared with RS scheme when $a$ is sufficiently large.

In Fig. 3 we set $a=0.4$, $c=0.4$, $D=7$, $d=4$ and $U=[10, 40]$. In Fig. 4 we set $a=0.4$, $b=0.4$, $d=4$ and $D=7$. Fig. 3(a) and Fig. 4(a) show regeneration times of all algorithms decrease with the increase of $b$ and $c$, respectively. Although the density of the nodes in the generated network does not change, the number of
links increases with the increase of $b$ and $c$, which also leads to higher bandwidth between the servers and the newcomer. Therefore, the regeneration times of the three algorithms decrease and the BLP algorithm outperforms other two schemes. Fig. 3(b) and Fig. 4(b) shows the improvement ratios of BLP decrease with the increase of $a$ compared with RS and MF. The reasons are similar with the case shown in Fig. 2.

In Fig. 5, we set $a=0.4$, $b=0.4$, $c=0.4$, $D=10$ and $U=[40, 90]$. Fig. 5(a) shows that the regeneration time of all algorithms decreases because the range of link bandwidth increases. Fig. 5(b) shows that the improvement ratio of BLP decreases. The reason is that the optimization space of the BLP algorithm is small when the range of the link bandwidth decreases. The simulation results also show that
the proposed BLP algorithm is suitable in heterogenous network. Note that the proposed BLP algorithm can reduce about 40% and 10% regeneration time comparing with RS scheme and MF scheme, respectively, when the range of the link bandwidth is sufficiently large.

In Fig. 7, we set $a=0.4$, $b=0.4$, $c=0.4$, $d=4$ and $U=[0, 40]$. The Fig. 7(a) shows the regeneration time of all algorithms decreases with the increase of $D$. The more servers can be selected to be providers, the higher probability that can find "good" providers with higher bandwidth between them to the new comer. Therefore, the the regeneration time of all algorithms decreases. Fig. 7(b) shows the improvement ratio of BLP increases with the increase of $D$ compared with RS and MF schemes. When the value of $D$ increases, the optimization space becomes larger.

### 7 Conclusion

In this paper, we investigate a problem of optimal provider selection to minimize the regeneration time. Specifically, we first give the definitions of the PRPS problem in the DSSs with NC. For the special case that the providers are given, we model the PRPS problem as a mathematical programming. On the basis of
the special case, we then formulate the optimal provider selection problem as a mixed integer linear programming problem and develop an efficient algorithm based on LP-relaxation (BLP) to solve the PRPS problem. Finally, extensive simulation experiments have been conducted, and the effectiveness of the proposed algorithm are shown from the results. Specifically, when the range of link bandwidth is large, the proposed BLP algorithm can reduce about 40% and 10% regeneration time comparing with RS scheme and MF scheme, respectively.

References

1. Cisco prediction, http://www.ctiform.com/news/guonei/438872.html
2. S.Ghemawat, H.Gobioff, S.-T.Leung: The google file system. In: ACM SIGOPS Operating Systems Review vol.37,no.5.pp.29–43.(2003)
3. B.Calder:Windows Azure Storage: A Highly Available Cloud Storage Service with Strong Consistency. In:Proc. of the 23rd ACM Symposium on Operating Systems Principles.(2011)
4. I.Reed, G.Solomon,Polynomial Codes over Certain Finite Fields. Journal of the Society for Industrial and Applied Mathematics,8(2):300–304.(1960)
5. R.Ahlsweede, N.Cai, S.-Y.R.Li, R.W.Yeung;Network Information Flowem. In:Proc. of the 8th International Workshop on Quality of Service(IWQOS), 46(4):1204-1216.(2000)
6. Y.Hu, H.Chen, P.Lee, Y.TangNCCloud: applying network coding for the storage repair in a cloud-of-clouds. In: Proc. of the 10th USENIX Annual Technical Conference.(2012)
7. K.V.Rashmi, N.B.Shah, P.V.Kumar:Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a productmatrix construction. In:Proc. of 19th International Workshop on Quality of Service(IWQOS), 57(8):5227C-5239.(2011)
8. V.R.Cadambe,S.A.Jafar, H.Maleki:Distributed data storage with minimum storage regenerating codes - exact and functional repair are asymptotically equally efficient. arXiv:1004.4299 [cs.IT].(2010)
9. Y.Wu, A.G.Dimakis, K.Ramchandran:Deterministic regenerating codes for distributed storage. In:Proc. of 45th Annual Allerton Conference on Communication, Control, and Computing.(2007)
10. J.Li, S.Yang, X.Wang, B.Li:Tree-structured Data Regeneration in Distributed Storage Systems with Regenerating Codes. In:Proc. of the 29th International Conference on Computer Communications.(2010)
11. J.Li, S.Yang, X.Wang, X.Xue, B.Li:Structure of data regeneration with network coding in distributed storage systems. In:Proc. of 17th International Workshop on Quality of Service(IWQoS).(2009)
12. D.Cullina, A.G.Dimakis, T.Ho:Searching for minimum storage regenerating codes. In:Proc. of 47th Annual Allerton Conference on Communication, Control, and Computing.(2009)
13. Y.Hu, Patrick P.C.Lee, Kenneth W.Shum:Analysis and Construction of Functional Regenerating Codes with Uncoded Repair for Distributed Storage Systems. In:Proc. of the 32nd IEEE International Conference on Computer Communications, vol.63, no.01, pp. 31–44.(2013)
14. W. Sun, Y. Wang, X. Pei: Tree-structured parallel regeneration for multiple data losses in distributed storage systems based on erasure codes. In: Proc. of Communications, China, no. 4, pp. 113–125. (2013)

15. Y. Wang, D. Wei, X. Yin, X. Wang: Heterogeneity-Aware Data Regeneration in Distributed Storage Systems. In: Proc. of the 33rd IEEE International Conference on Computer Communications. (2014)

16. Marton S., Frank H.P. Fitzek, Daniel E. Lucani, Morten V. Pedersen: Dynamic Allocation and Efficient Distribution of Data Among Multiple Clouds Using Network Coding. In: Proc. of the 3rd International Conference on Cloud Networking. (2014)

17. Al-khedhairi A. Simulated Annealing Metaheuristic for Solving p-Median Problem. International Journal of Contemporary Mathematical Sciences. (2008)

18. Waxman B M. Routing of multipoint connections[J]. International Journal of Selected Areas in Communications. 1988, 6(9): 1617–1622.

19. IBM ILOG CPLEX http://www-03.ibm.com/software/products/zh/ibmilogcplex

20. Q. Gong, J. Wang, D. Wei, J. Wang and X. Wang: Optimal node selection for data regeneration in heterogeneous distributed storage systems. In: Proc. of the 44th International Conference on Parallel Processing (ICPP). (2015)

21. Q. Gong, J. Wang, Y. Wang, D. Wei, J. Wang, X. Wang: Topology-Aware Node Selection for Data Regeneration in Heterogeneous Distributed Storage Systems. preprint arXiv:1506.05579 (2015)