

# A Heuristic Algorithm Based on Combinatorial Optimization Problem in Content Distribution Networks

Jun-fang Guo

Wenzhou University, Zhejiang, 325035, China

e-mail: gjf@wzu.edu.cn

## Abstract

*Content Distribution Network (CDN) is an effective approach to improve the quality of Internet service. This paper firstly reviews some of the related works which have contributed to the current state of-the-art; the focus is on the fundamental assumptions, mathematical models and specific references to solution approaches. What's more, a mathematical programming formulation based on combinatorial optimization problem in content distribution networks is proposed, and then a corresponding heuristic algorithm for this optimization problem is given, the total cost of this model is studied in CDN. Finally, the theoretical analysis shows that the proposed algorithm performs better than previous algorithms.*

**Keywords:** facility location, content distribution network, approximation algorithm, modeling

**Copyright © 2014 Institute of Advanced Engineering and Science. All rights reserved.**

## 1. Introduction

Content Distribution Network is an effective approach to improve the quality of Internet service. CDN replicates the content from the origin place to the replica servers scattered over the Internet and serves a request from a replica server close to which the request originates in. What's more, A Content Distribution Network (CDN) is also an interconnected system of computers on the Internet that provides Web content rapidly to numerous users by duplicating the content on multiple servers and directing the content to users based on proximity. Recently, with the emergence of Content Distribution Network (CDN); CDNs are targeted for speeding up the delivery of normal Web content and reducing the load on origin servers and the network. A content provider can sign up for the service and let its content place on the content servers. The on-demand users replicate its content when requesting it, or replicating beforehand, by pushing the content on the content servers. CDNs are used by Internet service providers to deliver static or dynamic Web pages but the technology is especially well suited to streaming audio, video, and Internet television programming.

Generally speaking, the CDN model can be as follows [1], suppose an autonomous system (AS) with  $n$  servers, where each server can serve certain number of requests per unit time. A request enters the system through ingress Provider Edge (PE) routers. Cost for serving one particular request will depends upon distance between server and PE within the AS. CDN providers always aim at optimizing the connection cost over all input requests.

The above problem can extend to the classical facility location problem. The facility location problem [2] is one of the fundamental combinatorial optimization problems in the branch of optimization or operations research in mathematics. The problem of locating facilities is not new direction to the operations research community. The challenge of this problem is that where the best site facilities located has inspired a rich, colorful and ever growing body of literature. To cope with the multitude of applications encountered in the business world and in the public sector, some expanding family of models have emerged [3-5]. Location-allocation models cover formulations which range in complexity from simple linear, single-stage, single-product, incapacitated deterministic models to non-linear probabilistic models. Algorithms include, among others, local search and mathematical programming based approaches. It is the purpose of this paper to review some of the work which has contributed to the current state of-the-art. The focus is on the fundamental assumptions, mathematical models and specific references to solution approaches.

This paper makes two primary contributions. At first, it proposes a mathematical programming formulation for this problem in content distribution networks. Second, an algorithm is developed for this problem. The remainder of this paper is as follows: In section 2, we discuss some review on facility location problem. Section 3 develops a problem formulation to this facility location in content distribution networks. Section 4 provides a heuristic solution procedure used to solve the problem. Finally, conclusions are provided in Section 6 of the paper.

## 2. Related Works

In a CDN, content exists in multiple copies on strategically dispersed servers. This is known as content replication. A large CDN can have thousands of servers, making it possible to provide identical content to many users efficiently and reliably even at times of maximum Internet traffic or during sudden demand "spikes." When a specific page, file, or program is requested by a user, the server closest to that user (in terms of the minimum number of nodes between the server and the user) is dynamically determined. This optimizes the speed with which the content is delivered to that user. The use of CDN technology has obvious economic advantages to enterprises who expect, or experience, large numbers of hits on their Web sites from locations all over the world. If dozens or hundreds of other users happen to select the same Web page or content simultaneously, the CDN sends the content to each of them without delay or time-out. Problems with excessive latency, as well as large variations in latency from moment to moment (which can cause annoying "jitter" in streaming audio and video), are minimized. The bandwidth each user "sees" is maximized. The difference is noticed most by users with high-speed Internet connections who often demand streaming content or large files. Another advantage of CDN technology is content redundancy that provides a fail-safe feature and allows for graceful degradation in the event of damage to, or malfunction of, a part of the Internet. Even during a large-scale attack that disables many servers, content on a CDN will remain available to at least some users. Still another advantage of CDN technology is the fact that it inherently offers enhanced data backup, archiving, and storage capacity. This can benefit individuals and enterprises that rely on online data backup services. CDN scheme of distribution given in Figure 1.

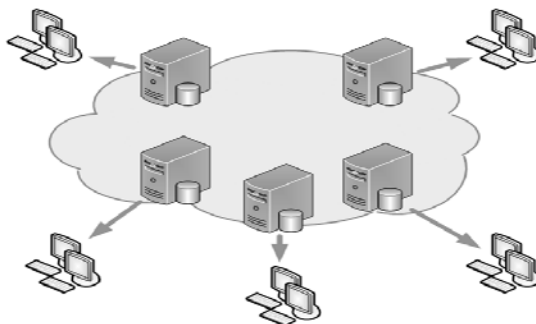


Figure 1. CDN Scheme of Distribution

There three elements of a CDN, to deliver feature such as file access and caching, a CDN must contain the three elements: Request, Distribution, and Delivery. The "request" element of a CDN deals with the ability of users and systems to ask for specific content, whether it be a file or a video. Because a request occurs at the user end, protocols have been developed to intercept and redirect these requests to the hardware components or content engines closest to the user. Once a request has been made, the content engine can decide whether it can answer the request or proxy it on the user's behalf. The "distribution" element of the CDN determines which decision (answer or proxy) is appropriate. Content has to come from somewhere within the architecture (origin servers), and based on patterns of use and requests, CDN administrators can distribute it appropriately. The choice of distribution, in turn, directly

affects the details of a request. Finally, the "delivery" element is responsible for getting the content to the correct locations within the architecture. This element relies heavily on the routing and switching infrastructure for reliable and efficient delivery.

The first approximation algorithm with constant approximation ratio of at most 3.16 for the facility location problem was developed by Shmoys and Tardos [6] with the filtering of Lin and Vitter [7], a randomized of the filtering and rounding technique that can be derandomized to yield improved performance guarantees. This approximation algorithm for the Incapacitated Facility Location problem is based on LP rounding that solves the relaxation of the integer program for Incapacitated Facility Location and rounding the solution. The algorithm of Shmoys, Tardos and Aardal first partitions the cities into clusters and then for each cluster opens exactly one facility, which services all the cities in it.

The main drawback of this approach is that most of the time the imbalance in bounding facility and service costs. Using a different rounding approach, that is the randomized rounding technique, Chudak and Shmoys [8] gave an algorithm with ratio  $1 + 2/e \approx 1.736$ , which is a significant improvement on the previously approximation guarantees, their techniques use the property of optimal dual solution to the linear program and complementary slackness, randomized rounding that can be easily derandomized using the method of conditional expectations, as well as a generalization of decomposition techniques. This was further improved to 1.582 by Sviridenko [8] by rounding an optimal fractional solution to a linear programming relaxation. Recently, Byrka [9] used the idea of scaling-up fractional solution and techniques from [10] to obtain another LP-rounding algorithm with currently best known ration of 1.5. Guha and Khuller [11] have shown that the metric incapacitated facility location problem is Max SNP-hard, they have also shown that the existence of a  $r$ -approximation algorithm for  $r < 1.463$ . The best known ration of 1.5 is very proximal to the result of 1.463. All these algorithms are based on solving linear programming relaxations of the facility location problem and rounding the solution obtained. The drawback of these algorithms based on LP-rounding is that they need to solve large linear programs, and so have prohibitive running times for most applications.

### 3. Classical Facility Location Models

#### 3.1. Incapacitated Facility Location Model

Incapacitated Facility Location [2] is one of the classical problems that have been widely studied in operations research, which is defined as follows. We are given several set as follows: a finite set of potential facilities  $F$ ; a finite set of cities or demands  $C$ ; a non-negative number  $f_i$  as the opening cost of facility  $i$ ; a connection cost  $c_{ij}$  between facility  $i$  and city  $j$  for every facility-city pair  $(i, j)$ ,  $i \in F, j \in C$ . The objective of the problem is to open a subset of the facilities in  $F$ , and connect each city to open facilities, such that the total cost of opening facilities and connecting cities to open facilities is minimized. We will consider the metric version of this problem, i.e., the connection distances are symmetric and satisfy the triangle inequality, or equivalently satisfy  $c_{ij} + c_{iq} \leq c_{jq}$  for all  $i, j, q \in F$  and  $j, q \in C$ . This is natural assumption, since connection costs are often associated with the distance between points in Euclidean space representing facilities and cities. Therefore we speak of metric connection costs if the above condition is satisfied. We make this assumption in the following sections. The Incapacitated Facility Location problem can be formulated by the following integer linear program.

Minimize:

$$\sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in C} c_{ij} x_{ij}$$

Subjected to:

$$\forall j \in C : \sum_{i \in F} x_{ij} \leq 1 \tag{1}$$

$$\sum_{j \in C} x_{ij} \geq 1 \quad \forall i \in F \quad (2)$$

$$x_{ij} \leq y_i \quad \forall i \in F, j \in C \quad (3)$$

$$y_i \in \{0,1\} \quad \forall i \in F \quad (4)$$

In the above formulation,  $y_i$  is an indicator variable denoting whether facility  $i$  is open, and  $x_{ij}$  is an indicator variable denoting whether city  $j$  is connected to facility  $i$ . The first constraint ensures that each city  $j \in C$  is connected to at least one facility, whereas the second ensures that cities are connected or served only by an open facility, that is whenever a city  $j$  is assigned to facility  $i$ , then a facility must have been opened (and paid for).

In addition, an extended version is considered [10]. The case in which each facility  $i$  can be assigned to serve at most  $u_i$  cities, that indicates the capacity of this facility, i.e., the number of cities it can serve, where  $u_i$  is a positive integer, if facility  $i$  is opened  $y_i$  times, it can serve at most  $u_i y_i$  cities. That is a capacitated version of Incapacitated Facility Location. First of all, there are two variants of the problem, depending on whether each city's demand must be assigned to only one facility, or the demand may be fractionally split among more than one (completely) open facility. We focus on the latter case. We allow the demand of cities to be split, i.e. assigned to multiple open facilities. While the total demand assigned to a facility must not exceed its capacity.

What's more, in real application, the  $k$ -median problem is considered [14, 15]. Similar to the facility location problem, there are two respects different from the facility location problem in the following. There is no cost for opening facilities, and there is an upper bound  $k$  on the number of facilities, where  $k$  is an input parameter that is not fixed. The objective is to select a set of  $k$  facilities (or centers) so as to minimize the sum of the assignment costs for the city.

### 3.2. The Metric Fault Tolerant Facility Location Model

In the previous sub-section, we have mentioned the Incapacitated Facility Location problem, but in Incapacitated Facility Location, only one connection is required for each city which is not the case in many application scenarios and therefore researchers extend the problem by proposing Fault-Tolerant Facility Location. In the following, we will introduce the Fault-Tolerant Facility Location problem [3]. The Fault-Tolerant Facility Location problem is a generalization of the Incapacitated Facility Location problem where each client has independent fault-tolerance requirement on connectivity. In many setting it is essential to provide safeguards against failures by designing fault-tolerant solutions. For example, in a distributed network, we want to place caches and assign data requests to caches so as to be resistant against caches becoming unavailable due to node or link failures. A common solution is to replicate data items across caches and build some resilience in the network. This motivates the Fault-Tolerant Facility Location problem wherein a city  $j \in C$  has a requirement  $r_j$  and has to be assigned to  $r_j$  distinct open facilities instead of just one. The facilities other than the closest one are 'backup' facilities for the city, and will be used only if the closer facility (or the link to it) fails. In this problem, we are given a set of facilities  $F$  and a set of cities  $C$ , and each city  $j$  has a requirement  $r_j$ . Each facility  $i$  has an opening cost  $f_i$ , the cost  $c_{ij}$  is to establish a connection between facility  $i$  and city  $j \in C$ . The FTFL problem can be formulated by the following integer linear program.

Minimize:

$$\sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in C} c_{ij} x_{ij}$$

Subjected to:

$$\sum_{j \in C} x_{ij} = r_j \quad (1)$$

$$y_i \geq \sum_{j \in C} x_{ij} \quad (2)$$

$$x_{ij} \in \{0,1\} \quad (3)$$

$$y_i \in \{0,1\} \quad (4)$$

In the above formulation, variable  $y_i$  indicates whether facility  $i$  is open, and  $x_{ij}$  indicates whether client  $j$  is assigned to facility  $i$ . The connection cost or assignment cost incurred for  $j$  is the sum of the distances from  $j$  to these  $r_j$  facilities. The objective of the problem is to open proper number of the facilities and assign each city  $j$  to  $r_j$  distinct open facilities so as to minimize the total facility opening and client assignment costs. Note that the first constraint ensures the number of connections required by each city and the second constraint reflects that a city can only be assigned to an open facility.

#### 4. A Novel Facility Location Model in Content Distribution Networks

##### 4.1. Problem Formulation

Content distribution networks problems involve strategic decisions which influence tactical and operational decisions [16-19]. In particular, they involve facility location, transportation [12] and inventory decisions, which affect the cost of the distribution system and the quality of the customer service level. So, they are core problems for each company. With incorporation of CPU computing capacity of each server as new parameter, old model of facility location problem in CDN gets mapped to multi-index quadratic assignment problem. In our new model, we consider CPU computing capacities while finding out optimal connection cost for serving input requests.

In this section, a relaxation of the facility location problem called facility location in content distribution networks problem is studied. Given a set of CDN providers, a set of clients, and corresponding server operating cost at each site as well as connection cost for each site-client pair, the facility location in content distribution networks problem requires to allocate a proper number of servers, and further such that clients can fetch objects from the nearest CDN server that containing the requested object. In this problem, we are given a set  $S$  of CDN servers and a set  $C$  of  $n_c$  clients. At each server  $i \in S$ , an unlimited number of servers with a non-negative cost  $s_i$  as costs can be operated. There is also a connection cost  $c_{ij}$  between each client  $j \in C$  and all servers at site  $i$ : Note that each client requires a pre-specified number of connections for fault tolerance purpose. Let the connectivity requirement of client  $j \in C$  be  $r_j$ . The objective is to optimally allocate a certain number servers from each  $i$  to serve every client  $j$  with  $r_j$  requests while minimizing the total combined cost for server operating and service access incurred. The facility location in content distribution networks problem can be formulated by the following integer linear program.

Minimize:

$$\sum_{i \in S} f_i y_i + \sum_{i \in S} \sum_{j \in C} c_{ij} x_{ij}$$

Subjected to:

$$\sum_{j \in C} x_{ij} \leq r_j \quad (1)$$

$$\sum_{j \in C} y_i \leq x_{ij} \quad (2)$$

$$x_{ij} \in \{0, 1, \dots, N\} \quad (3)$$

$$y_i \in \{0, 1, \dots, N\} \quad (4)$$

In the above formulation, non-negative integer  $y_i$  indicates how many servers are opened at  $i$  and  $x_{ij}$  indicates how many connections are established between  $i$  and  $j$ . The first constraint is to ensure that each client  $j \in C$  has fault tolerance of connections, that is to say, it is assigned totally  $r_j$  connections but it can tolerate up to  $r_j - 1$  connection failures. The second constraint insures that each client  $j \in C$  has fault tolerance of servers that is assigned distant servers for any two connections and is able to tolerate up to  $r_j - 1$  server. In this section, we consider the metric version of the problem where the given connection costs satisfy triangle inequality.

The differences between the facility location in Content Distribution Networks problem and two Facility Location problems mentioned in the mentioned above sub-sections are as follows: the facility location in Content Distribution Networks problem is quite close to the well studied Fault-Tolerant Facility Location problem which has the same objective function and constraints as the facility location in Content Distribution Networks problem, except for this condition, the range of variants is different. For  $x_{ij}$  and  $y_i$  are non-negative integers in the facility location in Content Distribution Networks but binary integers (0 or 1) in Fault-Tolerant Facility Location problem. The Fault-Tolerant Facility Location problem restricts at most one server, without the restriction on the maximum number of servers at each site, the facility location in Content Distribution Networks problem is less constrained and therefore incurs less cost which is desirable in real applications. The facility location in Content Distribution Networks problem can be applied in the deployment of ATMs, server farms applications, of which multiple servers can be deployed if necessary and service the clients.

#### 4.2. A Fast Algorithm for the Facility Location in Content Distribution Networks

CDN consists of several individual components, or infrastructures, whose purpose is to provide a service to the user community. An example of an infrastructure component could be consolidated file servers, which act as repositories for user files and application data. These servers sit atop another component of the architecture: the routing and switching infrastructure. The foundation for network architecture lies in the routing and switching infrastructure, which provides transport for all other infrastructure components and their various forms of data.

Some researchers have proposed a heuristic known as "heuristics concentration", and applied this to the p-median problem. Heuristic concentration (in the context of the p-median problem) is based on running an efficient heuristic numerous times as to identify a subset of the potential facility sites that may warrant further investigation. After repeatedly running a randomized greedy heuristic, the information about which nodes are in the "p" set is collected, and a final subset of nodes is used when solving the subset problem to optimality by using integer-linear programming. Thus, using a subset of the potential facility nodes results in an optimal solution if all nodes that are in the optimal "p" set are in the subset. Also, this subset is limited, and the time to find the optimal solution to this sub-problem is only a fraction of that of optimally solving the problem using the full set of potential facility sites. It is our intent to propose a heuristic for solving problem Refurb that takes advantage of the above observations. Then some authors compared heuristic concentration to tabu search and concluded that heuristic

concentration is superior to tabu search for the p-median problem. We then propose several replica placement locations for CDN servers that can determine the optimal number of replicas we should select from a given set of potential location such that clients can fetch objects from the nearest CDN server that containing the requested object.

**Algorithm 1.** A Fast Algorithm for the Facility Location in Content Distribution Networks

Begin

Step1. Give the facility location in Content Distribution Networks like a rooted binary tree, with servers and clients at leaves.

Step2. Initialize DP for clients:

If there is a client with bandwidth demand  $N$  and CPU demand  $C$  at a leaf  $v$

For a demand, let for any server  $x$ , dynamic programming value associated with  $(v, 0, 0, 0, 0, \text{NONE}, x)$  be  $N * C * \text{distance between } x \text{ and } v$ ; all other values corresponding to this leaf are set to infinity;

If server  $i$  is located at this leaf,

Let  $(v, 0, 0, 0, 0, \text{NONE}, \text{NONE})$  gets value 0;

Put value infinity for any other DP cell corresponding to this subtree;

Step3. Update the dynamic programming memoization table values bottom-up. If the node  $u$  has two children  $u_1$  and  $u_2$ , a cell  $X = (u, \text{F-N}, \text{F-C}, N, C, x, y)$  is updated (if this gives a better value) from  $X1 = (u1, \text{F-N1}, \text{F-C1}, N1, C1, x1, y1)$  and  $X2 = (u2, \text{F-N2}, \text{F-C2}, N2, C2, x2, y2)$  to  $\text{cost}(X1) + \text{cost}(X2)$ , if  $C$  is consistent with  $X1$  and  $X2$ .

Step4. Build the answer recursively from  $\text{DP}(r, 0, 0, 0, 0, \text{NONE}, \text{NONE})$ , with  $r$  being the root of the tree.

End

### 4.3. The Cost of Facility Location in Content Distribution Networks

For a given server location, we can calculate  $c_{ij}$ , the distance for client  $j$  to its nearest replica, and  $c_i$ , the distance from replica  $i$  to the origin server. To make a theoretical analysis of the total cost, we make an assumption about the relationship between the average distance from clients to origin servers and the average distance from clients to their nearest replicas. Based on it, we derive a model for the total cost of replica placement changing as the number of replicas increase.

Assume the average distance from clients to origin servers (over all clients, origin server pairs) is  $r$ . As we put more replicas, the average distance from clients to their nearest replicas will decrease. It can be modeled as a function of  $r$  and the number of replicas  $k$ , denoted as  $f(r, k)$ . We make a hypothesis about the relationship by defining

$f(r, k) = \frac{r - 1}{b^{k-1}} + 1$ , where  $b$  is a decay factor and  $b > 1$ . For a network with a few hundreds of

nodes,  $b$  will be large because increasing the number of replicas will reduce the distance from a client to its nearest replica significantly. For a network of the Internet size;  $b$  tends to be very close to 1, but it is still an exponential decrease when we increase the number of replicas.

Based on the above model, we analyze the total cost of facility location in content distribution networks when the number of replicas changes. We illustrate the relationship between three distances:  $c_{ij}$ , the distance from a client to its nearest replica,  $c_i$ , the distance from the replica to the origin server, and  $co_j$ , the distance from the client to the origin server. We know  $co_j$  can be modeled by the average distance  $r$  and  $c_{ij}$  can be modeled by the function  $f(r, k)$ . The other term used in calculating the total cost is  $c_{ij} + c_i$ . Usually replicas are located at access points of clients to the network and therefore we have  $c_i \ll co_j$ . Combining it with the

triangle relation, we have  $co_j \leq c_{ij} + c_i \leq co_j + c_{ij}$ . As we put more replicas, the replica is more likely to be located on the path from the origin server to the client and therefore  $c_{ij} + c_i \gg co_j$ . With this simplification we can use  $r$  to model the cost  $c_{ij} + c_i$ .

## 5. Conclusion

In this paper, the novel facility location model in Content Distribution Networks is proposed. We formulate this problem as a combinatorial optimization problem. As CDN servers with finite storage capacity, and we treat each CDN server as a node with finite storage capacity, the goal of this optimization problem is to minimize the total CDN server operating cost as well as connection cost for each client-server pair such that clients fetch objects from the nearest CDN server containing the requested object, the goal of this optimization problem is to minimize the total operating cost of CDN servers as well as connection cost for each client-server pair such that clients can fetch objects from the nearest CDN servers that containing the requested object. Then a corresponding heuristic algorithm for this optimization problem is given. In addition, we also studied the total cost of this model in CDN.

## Acknowledgements

The author gratefully acknowledges the funding of Zhejiang Provincial Natural Science Foundation of China under Grant No. LY12F01016.

## References

- [1] S Jin, A Bestavros. *Greedy dual web caching algorithm: exploiting the two sources of temporal locality in web request streams*. Proceeding of the 5th Web Caching and Content Distribution Workshop. 2000; 22-24.
- [2] G Cornuejols, GL Nemhauser, LA Wolsey. *The incapacitated facility location problem*. Discrete Location Theory. 1990; 119-171.
- [3] Byrka, A Srinivasan, C Swamy. *Fault-tolerant facility location: a randomized dependent LP-rounding algorithm*. Proceeding of the 14th conference on Integer Programming and Combinatorial Optimization. 2010; 244257.
- [4] C Swamy, D Shmoys. Fault-tolerant facility location. *ACM Transaction on Algorithms*. 2008; 4(4): 1-27.
- [5] S Guha, AM Kamesh Munagala. A constant factor approximation algorithm for the fault-tolerant facility location problem. *Journal of Algorithms*. 2003; 48(2): 429-440.
- [6] DB Shmoys, E Tardos, K Aardal. *Approximation algorithms for facility location problems*. Proceedings of the 29th Annual ACM Symposium on Theory of Computing. 1997; 265-274.
- [7] JH Lin, JS Vitter. *e-approximation with minimum packing constraint violation*. Proceedings of the 34th ACM STOC. 2002; 771-782.
- [8] M Mahdian, Y Ye, J Zhang. Approximation algorithms for metric facility location problems. *SIAM Journal on Computing*. 2006; 36(11): 411-432.
- [9] M Sviridenko. *An improved approximation algorithm for the metric incapacitated facility location problems*. Proceedings of 16th International Conference on Integer Programming and Combinatorial Optimization. 2009; 240-257.
- [10] J Byrka, KI Aardal. An optimal bi-factor approximation algorithm for the metric incapacitated facility location problem. *SIAM Journal on Computing*. 2010; 39(10): 2212-2231.
- [11] S Guha, S Khuller. *Greedy strikes back: Improved facility location algorithms*. Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms. 1998; 127-132.
- [12] TG Crainic, G Laporte. Planning models for freight transportation. *European Journal of Operational Research*. 1997; 97(5): 409-438.
- [13] D Xu, S Zhang. Approximation algorithm for facility location with service installation costs. *Operations Research Letters*. 2008; 36(1): 46-50.
- [14] M Thorup. Quick k-Median, k-Center, and Facility Location for Sparse Graphs. *SIAM Journal on Computing*. 2005; 34(2): 405-432.
- [15] D Fotakis. Incremental Algorithms for Facility Location and k-Median. *Theoretical Computer Science*. 2006; 361(2-3): 275-313.
- [16] L Qiu, VN Padmanabhan, GM Voelker. *On the placement of web server replicas*. Proceedings of IEEE Infocom. 1999; 21-25.



- 
- [17] I Cidon, S Kutten, R Soffer. *Optimal allocation of electronic content*. Proceedings of IEEE Infocom. 2001; 322 ~326.
- [18] Huiting Chen, Xiaolin Luo, Yongquan Li, Huachen Fu, Xiaolei Zheng, Xiaotao Yuan. III-conditioned Load Flow Study Based on Asynchronous Parallel Computing in Rural Power Distribution Networks. *JCIT: Journal of Convergence Information Technology*. 2012; 7(19): 261~268.
- [19] Changhong Yu. A TVC based Clock Fault Detection Circuit for Modern Clock Distribution Network. *AISS: Advances in Information Sciences and Service Sciences*. 2011; 3(3): 75~80.