2010 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE)

Novel Spectral Method for Server Placement in CDNs

Lan Li
School of Software
Nanchang University
Nanchang, Jiangxi, China
Email: lilan@ncu.edu.cn

Min Tang
China Helicopter Research and Development Institute
Jingdezhen, Jiangxi, China

***Abstract*—In this paper, we present a novel method for replica server placement in CDNs. As we all known, replica server should be placed closer to clients in order to reduce the latency and the bandwidth consumption. To minimize the maximum distance between a node and the nearest center, the minimum K-center problem and the k-HST are pesented by researchers. However, these theoretical methods are either computationally expensive or leave the characteristics of the network out of account. We improved the k-HST by applying the spectral clustering method into the decision problem about the number of surrogate server centers. The deterministic method we present is suitable for ISP or companies who already know the topology of its network to distribute server centers.**

***Keywords*—spectra;CDN;k-HST;Server placement;clustering**

I. Introduction

Recent years have been experienced the rapid spread of information and the ubiquitous access of the browsers. With the tremendously increasing of the amount of information, the frequently experiencs of unacceptable delays for the users is a common phenomena. Content delivery(or distribution) networks(CDNs) [1][2] is a kind of new emerging technology, which has recently been proposed to improve the performance of the response time, bandwith and the accessibility. The placement of surrogates to some suitable positions can let the edge servers become closer to the clients. Content providers will be benefited by reducing latency for their clients and ISPs will be benefited by reducing bandwith consumption if appropriate placement strategies presented in time. In concrete, the issue of the cost-effective replica servers placement is critical to the reduction of the overall traffic in the network.

The existing work on server placement has primarily focus on two aspects, one is the decision of the number of the replica servers and the other is the replica server placement problem. If a Web site aims to improve user-perceived performance, the common method is moving the selective set of content from the origin servers to the surrogate server. We adopt the following notations in this paper[3]: the network is represented by a graph $G(V, E)$, where V is the set of nodes, and $E \subseteq V \times V$ is the set of links. We use $N = |V|$ to denote the number of nodes in G, and $\mathcal{T}$ to denote the number of centers we place in the graph. The basic problem we need to resolve is to decide the number of the centers $\mathcal{T}$ in N nodes and select $\mathcal{T}$ replica servers such that the maximum distance between a node on the graph and the nearest replica server is minimized. According to this, there are two main approaches have been proposed. One is the center placement problems, which use the theoretical approaches such as the k-hierarchically well-separated tree(k-HST) [4] and the minimum K-center problem[3]. The other is the heuristics methods, such as the Greedy replica placement[5], topology-informed placement strategy[6], hot spot [7], tree-based replica placement[8] and scalable replica placement[9].

In this paper, we mainly present a novel spectral clustering way to improve the center placement problems. The graph theoretic spectral algorithm we used in this paper is based on the integer programming formulation of the laplacian matrix on server placement problem, which is a kind of dermininstic technique. The results of our spectral algorithm are within a known factor of the optimal solution. It is applicable for Internet Service Provider (ISP) or companies to use this algorithm to distribute Tracers within its own network or intranets.

The rest of this paper is organized as follows. In section 2, we survey the related work about server placement. Then, we present our Spectral approach for the caching server placement. At last, we give the theoretic analysis for our algorithm.

II. Related work

In this section , we review the primary relevant graph theoretic work about the server placemnet.

A. Combinational Optimization

Our algebraic formulation of caching server clustering problem is based on matrix form representation of a graph known as the Laplacian. Donath and Hoffman[10] and Fiedler[11] firstly find out the relationship between Laplacian and the eigen-vallues can be applied in partitions of undirected graphs. Chung [12] gave a detail and comprehensive survey of Laplacian matrices. In brief, Spectral clustering is a basic approach for partitioning the rows and columns of matrices derived from the data in terms of their eigenvectors. Eigenvalue characteristics have been applied to design various efficient algorithms for the problems in the clustering, partition and grouping. [13][14]

B. k-HST

The k-hierarchically well-separated tree (k-HST) algorithm solve the server placement problem according to graph theory. There are two phases in the algorithm. [4] In the first phase, a node is arbitrarily selected from the entire graph current (parent) partition, and all the nodes that are within a random radius from this node form a new (child) partition. The value of the radius of the child partition is a factor of k smaller than the diameter of the parent partition. This process recursively continues until each node is in a partition of its own. Thus the graph is recursively partitioned and a tree of partitions is obtained with the root node being the entire network and the leaf nodes being individual nodes in the network. In the second phase, a virtual node is assigned to each of the partitions on each level. Each virtual node in a parent partition becomes the parent of the virtual nodes of the child partitions. The length of the links from a virtual node to its children is half the partition diameter. Together, the virtual nodes also form a tree.

C. minimum K-center

The maximum distance from a node to the nearest center is minimized if the placement of a given number of centers is set properly. It is well known as the minimum K-center problem. It can be described as follows: (1) Given a graph G(V, E) with all its edges arranged in ascending order of edge cost c: $c(e_1) \leq c(e_2) \leq \ \leq c(e_m)$, construct a set of square graphs $G_1^2, G_2^2, ., G_m^2$. Each square graph of G, denoted by G^2 is the graph containing nodes V and edges (u, v) wherever there is a path between u and v in G of at most two hops. (2) Compute the maximal independent set M_i for each G_i^2. An independent set of G_2 is a set of nodes in G that at least three hops apart in G and a maximal independent set M is defined as an independent set V such that all nodes in V - V are at most one hop away from nodes in V . (3) Find smallest i such that $M_i \leq K$, which is defined as j. (4) Finally, M_j is the set of K center.

III. Spectral Approach for Server Placement

Given a network G with N nodes, we search for a good partition of the G. We achieve this by treating the partition as an optimization problem where the goal is to optimize the value of an objective function. Firstly, we introduce the algebraic notations related to the structure of G. The node set V is divided into three subsets as $V = I \cup J \cup S$, where I is the set of clients, J is the set of potential nodes on which proxy(caching) server can be established and S is the set of origin servers. We limit our research to a single origin server, that is , $|S| = 1$. And each client is assumed to be served by exactly one proxy server. Given a bound $\mathcal{B}$, We look for smallest set of centers $S_C \in J$ such that the distance between any client i and its closest cener $C_j \in S_C$ is bounded by $\mathcal{B}$. Formally, Our question is to find the minimum $|S_C|$ such that $\exists i \in V : d(i, C_j) \leq \mathcal{D}$.

A. Objective Function

We refer to our objective function as the following addition function.

$$PQ = \sum_{j=1}^{|J|} d(i, C_j). \tag{1}$$

Here, $d(i, C_i)$ is a cost function of serving client i from center C_j.

$$d(i, C_j) = f_{ij}(h_{ij}c_{ij} + (1 - h_{ij})(c_{ij} + c_{js})). \tag{2}$$

Here, f_{ij} is the amount of flow between a client i $\in$ I and a caching server center $C_j \in S_C \in J$. h_{ij} is the fraction of the request originating from node i $\in$ I that can be satisfied by $C_j \in S_C \in J$. c_{ij} is the distance (number of hops or cost) between two nodes i $\in$ V and j $\in$ V. It is obvious that the $d(i, C_j)$ is minimized if h_{ij} is maximized such that we simplifies our objective function as follows.

$$PQ = \sum_{j=1}^{|J|} h_{ij}. \tag{3}$$

$$h_{ij} = \begin{cases} 0 & e_i = 0 \\ \frac{2e_i}{2e_i + \sum_{i=1, i \neq j}^{|J|} (\varepsilon_{ij} + \varepsilon_{ji})} & otherwise \end{cases} \tag{4}$$

here, e_i present the total cost of the internal edges in the partition of center C_j and ε_{ij} present the external edges.

We now give the matrix representation to the PQ optimization function. Let $\mathcal{A}$ denote the adjacency matrix of G, let $d_G(i)$ denote the degree of a client node i and $\mathcal{L}$ denote the Laplacian matrix of the G.

$$\mathcal{L}_{ij} = \begin{cases} -1 & \mathcal{A}_{ij} = 1 \\ d_G(i) & i = j \\ 0 & otherwise \end{cases}$$

Assume that we firstly devide the nodes in G into two parts, and each part has one center server C_j, j = 0,1. Let $x_{ij} \in 0, 1$, if client i $\in$ I is assigined to center(caching) server $C_j \in J, x_{ij} = +1$, and -1 otherwise. We change h_{ij} by introduce the variable x_{ij} and Matrix $\mathcal{L}$ into the formulation. As we discussed above, e_i represent the total cost of the internal edges in the partition of center C_j.

$$2e_i = \sum_{x_i>0, x_j>0} \mathcal{A}_{ij}x_ix_j = \sum_{x_i>0, x_j>0} \mathcal{L}_{ij}x_ix_j \tag{5}$$

According to the definition of Laplacian matrix $\mathcal{L}$, we can make deduction that:

$$e_i = \frac{1}{2} \sum_{x_i} (\mathcal{L}_{ij} - \sum_{x_j} -\mathcal{L}_{ij}x_ix_j), x_ix_j < 0. \tag{6}$$

It is obvious that the denominator $\mathcal{D}$ of h_{ij} can be indicated as follows:

$$\mathcal{D} = \sum_{x_i} \mathcal{L}_{ij} = \sum_{x_i} D_G(i). \tag{7}$$

According to the formulation (5)-(7), we can rewrite the PQ in quadratic matrix form as follows:

$$PQ = 2 - \left(\frac{\sum_{x_i>0, x_j<0} -\mathcal{L}_{ij}x_ix_j}{\sum_{x_i>0} D_G(i)} + \frac{\sum_{x_i<0, x_j>0} -\mathcal{L}_{ij}x_ix_j}{\sum_{x_i<0} D_G(i)} \right) \tag{8}$$

Now, the problem of the maximization of PQ becomes the problem of minimization of PQ^*, which means that our objective function is :

$$PQ^* = Minimize : \left(\frac{\sum\limits_{x_i>0,x_j<0} -\mathcal{L}_{ij}x_ix_j}{\sum\limits_{x_i>0} D_G(i)} + \frac{\sum\limits_{x_i<0,x_j>0} -\mathcal{L}_{ij}x_ix_j}{\sum\limits_{x_i<0} D_G(i)} \right) \tag{9}$$
$$Subject\ to: \ x_i \in \{-1,1\}, 1 \leq i \leq S_C.$$

Here, S_C is the number of server centers.

Therefor, the quadratic optimization problem can be related with the spectral properties of the Laplacian matrix $\mathcal{L}$. We then simplify (9) to let our formulation get more closer to the Laplacian matrix. Let Λ denote the diagonal matrix with $\Lambda_{i,i} = D_G(i)$, $\lambda = \frac{\sum_{x_i<0} D_G(i)}{\sum D_G(i)}$ denote the normalized degree of the center server C_i clustering and $\mathcal{E}$ denote the identity vector. Given $\mu = \frac{\lambda}{1-\lambda}$ and $\mathcal{X}$ is the vector waited to be divided with entries of 1 or -1. Then, we get simplified PQ^* as :

$$PQ^* = \frac{((\mathcal{E}+\mathcal{X}) - \mu(\mathcal{E}-\mathcal{X}))^t L((\mathcal{E}+\mathcal{X}) - \mu(\mathcal{E}-\mathcal{X}))}{\mu\mathcal{E}^t\Lambda\mathcal{E}} \tag{10}$$

Let $\mathcal{H} = \mathcal{E}+\mathcal{X}) - \mu(\mathcal{E}-\mathcal{X})$, we can get a very simplified equation about PQ^* as following:

$$PQ^* = Minimize : \frac{\mathcal{H}^t\mathcal{L}\mathcal{H}}{\mathcal{H}^t\Lambda\mathcal{H}} \tag{11}$$

It is well know that the eigenvalue of the G's Laplacian matrix $\mathcal{L}$ is the minimizer of any quadratic form $\frac{\mathcal{H}^t\mathcal{L}\mathcal{H}}{\mathcal{H}^t\mathcal{H}}$. Given a new variable $\mathcal{I} = \Lambda^{\frac{1}{2}}\mathcal{H}$, the optimization problem comes down to computing the second smallest eigenvalue of matrix $\Lambda^{-\frac{1}{2}}\mathcal{L}\Lambda^{\frac{1}{2}}$.

B. Number of Centers

According to the optimization problem we discussed above, we sort the entries of eigenvector $\mathcal{H}$ and find a proper dividing point that generating the partitionS_{C1}, S_{C2}, here C_1, C_2 is the center of each cluster. After we get the two clusters of caching servers, we can recursively use the process of finding second smallest eigenvalue of matrix $\Lambda^{-\frac{1}{2}}\mathcal{L}\Lambda^{\frac{1}{2}}$ to achieve the further partition of each cluster. Finally, we count the number of the divided partitions, which is the number of the clustering center. We summarize the recursive bisection algorithm as Algorithm1 shows:

IV. Evaluation and Analysis

According to Kannan [15], the quality of a clustering is measured by two parameters : α and ϵ. α is the minimum conductance of the cluster ϵ the ratio of the weight of inter-cluster edges to the total weight of all edges. The algorithm of estimating number of centers will generate the clusters that have a conductance volume within a factor $\frac{1}{C_i \log^2 S_C}$ if we use the theoretic measurement. The algorithm we presented is efficient for small graph, whereas, for a CDN provider, the number of relica servers is limited. Therefore, our algorithm give a deterministic clustering way to divide the potential replica servers.

Algorithm 1 Improved k-HST tree
INPUT : the adjacent matrix $\mathcal{A}$ of Graph G(V,E)
OUTPUT: $|S_C|$, the number of clustering centers of the set of nodes of V.

```
Construct the diagonal matrix of degrees Λ;
Create the Laplacian matrix 𝓛;
|S_C| = 1;
while d(i, C_j) > 𝓑 do
    Improved k-HST tree(𝓐_ij);
    for evey node 𝓐_ij in Graph do
        Compute the second smallest eigenvalue of matrix Λ^{-1/2} 𝓛 Λ^{1/2}
        if (𝓗^t 𝓛 𝓗)/(𝓗^t Λ 𝓗) is minimized then
            𝓐_ij ∈ S_Ci;
        end if
    end for
    |S_C| = |S_C| + 1;
end while
```

References

[1] G. Pallis and A. Vakali, "Insight and perspectives for content delivery networks," *Communications of the ACM*, vol. 49, no. 1, p. 106, 2006.

[2] G. Peng, "Cdn: Content distribution network," *Arxiv preprint cs/0411069*, 2004.

[3] S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, and L. Zhang, "On the placement of internet instrumentation," in *IEEE INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, vol. 1, 2000.

[4] Y. Bartal, "Probabilistic approximation of metric spaces and its algorithmic applications," in *Annual Symposium on Foundations of Computer Science*, vol. 37. Citeseer, 1996, pp. 184–193.

[5] P. Krishnan, D. Raz, and Y. Shavitt, "The cache location problem," *IEEE/ACM Transactions on Networking (TON)*, vol. 8, no. 5, pp. 568–582, 2000.

[6] S. Jamin, C. Jin, A. Kurc, D. Raz, and Y. Shavitt, "Constrained mirror placement on the internet," in *IEEE INFOCOM*, vol. 1. Citeseer, 2001, pp. 31–40.

[7] L. Qiu, V. Padmanabhan, and G. Voelker, "On the placement of web server replicas," in *IEEE INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, vol. 3, 2001.

[8] B. Li, M. Golin, G. Italiano, X. Deng, and K. Sohraby, "On the optimal placement of web proxies in the internet," in *IEEE INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, vol. 3, 1999.

[9] Y. Chen, R. Katz, and J. Kubiatowicz, "Dynamic replica placement for scalable content delivery," *Peer-to-Peer Systems*, pp. 306–318, 2002.

[10] W. Donath and A. Hoffman, "Algorithms for partitioning of graphs and computer logic based on eigenvectors of connection matrices," *IBM Technical Disclosure Bulletin*, vol. 15, no. 3, pp. 938–944, 1972.

[11] M. Fiedler, "Eigenvectors of acyclic matrices," *Czechoslovak Mathematical Journal*, vol. 25, no. 100, pp. 607–618, 1975.

[12] F. Chung and C. C. on Recent Advances in Spectral Graph Theory, "Spectral graph theory," 1997.

[13] Q. Han, Y. Ye, H. Zhang, and J. Zhang, "On approximation of max-vertex-cover* 1," *European Journal of Operational Research*, vol. 143, no. 2, pp. 342–355, 2002.

[14] U. Zwick, "Outward rotations: a tool for rounding solutions of semidefinite programming relaxations, with applications to max cut and other problems," in *Proceedings of the thirty-first annual ACM symposium on Theory of computing*. ACM, 1999, p. 687.

[15] R. Kannan and A. Vetta, "On clusterings: Good, bad and spectral," *Journal of the ACM (JACM)*, vol. 51, no. 3, pp. 497–515, 2004.