

Bachelor Thesis

The Server Location Problem with Restricted Loads on Servers and Links

Alexander Svozil

September 9, 2013

1 Abstract

The server location problem with restricted loads on servers and links (SLRL) is an NP-Complete problem, introduced by Hiroyoshi Miwa et al. in the Paper "Method of Locating Mirror Servers to Alleviate Load on Servers and Links" [NM11]. The problem came up, because the massive volume of data distributed by content delivery networks (CDNs) require well located mirror servers in order not to badly influence the quality of their service. Two examples for CDNs would be "Amazon CloudFront" a traditional commercial CDN, or the "AT&T Inc." a Telco CDN which has advantages over traditional CDNs because they own the so called "last mile", the final leg of the telecommunications networks. CDN nodes are usually deployed in multiple locations, often over multiple backbones, reaching thousands of nodes with tens of thousands of servers. [Wik13a] [Wik13b] The two constraints induced by the choice of mirror servers is the number of maximum nodes accessing a mirror server and maximum number of nodes accessing a mirror server through a specific link. The constraint mentioned first, corresponds to the network load on the (mirror-)servers. The second constraint corresponds to the restricted load on the links. First I want to prove that SLRL is NP-complete. Next, I want to propose my own algorithm as a sequel to the existing greedy algorithm proposed by Miwa et al. [NM11].

Contents

1	Abstract	1
2	Introduction	2
2.1	CDNs	2
2.2	Similar Problems	3
2.2.1	P-Center Problem	3
2.2.2	P-Median Problem	3
2.2.3	NA (Node to Area)-connectivity	3
2.3	Open Shortest Path First	3
3	Server Location Problem with Restricted Loads on Servers and Links	3
3.1	Formal Definition	3
3.1.1	Neighbour Set	3
3.1.2	Load on a Link, $m(e)$ or multiplicity of an edge	3
3.1.3	The Problem	4
3.2	An Instance Of The Problem	5
3.3	Proof of NP-Completeness	7
4	Algorithm	7
4.1	Rules and Assumptions	7
4.2	Other Algorithms	7
4.3	Data (Testinstances)	7
4.4	Implementation	8
5	Conclusion	8

2 Introduction

2.1 CDNs

"A content delivery network is a collaborative collection of network elements spanning the Internet, where content is replicated over several mirrored Web servers in order to perform transparent and effective delivery of content to the end user." [BPV08, p. 3] To perform efficient delivery of data, CDNs need to have several mirrored Web servers and thus it is important to place

them well, because placing the mirror server could be an expensive choice considering no prior calculation where it should go. The expensive choice manifests in high link usages and or a maximal neighbourhood being too big for the server to handle. The mirror servers naturally serve the same data, as they are there to alleviate the load of the other servers providing the data.

2.2 Similar Problems

A lot of thought has already been put into the server location problem. Two problems that describe this problem would be the p-center and the p-median problem. The purpose of these problems is mainly to shorten delay time from the client to the server. There are several similar problems that are well known, like the p-center or the p-median problem.

2.2.1 P-Center Problem

2.2.2 P-Median Problem

2.2.3 NA (Node to Area)-connectivity

2.3 Open Shortest Path First

3 Server Location Problem with Restricted Loads on Servers and Links

3.1 Formal Definition

3.1.1 Neighbour Set

The neighbour set is the set, that belongs to a server. In our problem, it is defined as follows: If the node has a shortest path to a server node, he belongs to this server nodes neighbour set. If the node has several shortest paths to multiple server nodes, it will belong to all of those server nodes neighbour set. The server itself will belong to its own neighbour sets.

3.1.2 Load on a Link, $m(e)$ or multiplicity of an edge

The load on a link is defined as follows: It is defined as the sum of shortest paths, that cross a link. If there are several shortest paths with the same

length to one or more servers every one of them is taken into account. This is done, because we must always consider the worst case when locating a server.

3.1.3 The Problem

Given: *Undirected graph $G(V, E)$ and positive integers k, r and c .*

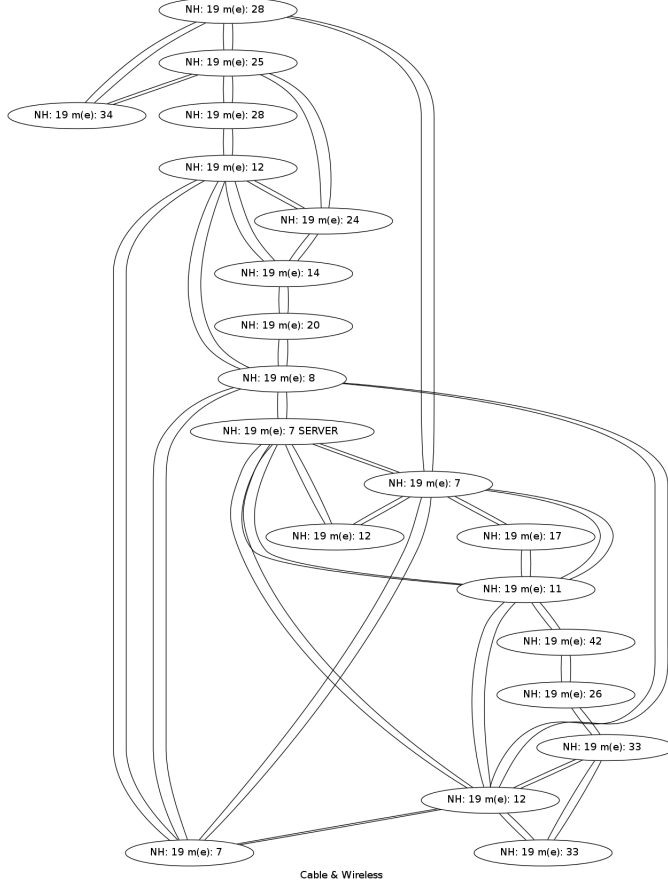
$k = \|S\|$, where S is the set of servers $S = \{s_1, s_2, \dots, s_k\} \subseteq V$

$r \geq \|V_i\|$ ($i = 1, 2, \dots, k$) where V_i is the neighbour set of s_i ($i = 1, 2, \dots, k$)

$c \geq m(e)$, where $m(e)$ is the maximum load $\forall e \in E$.

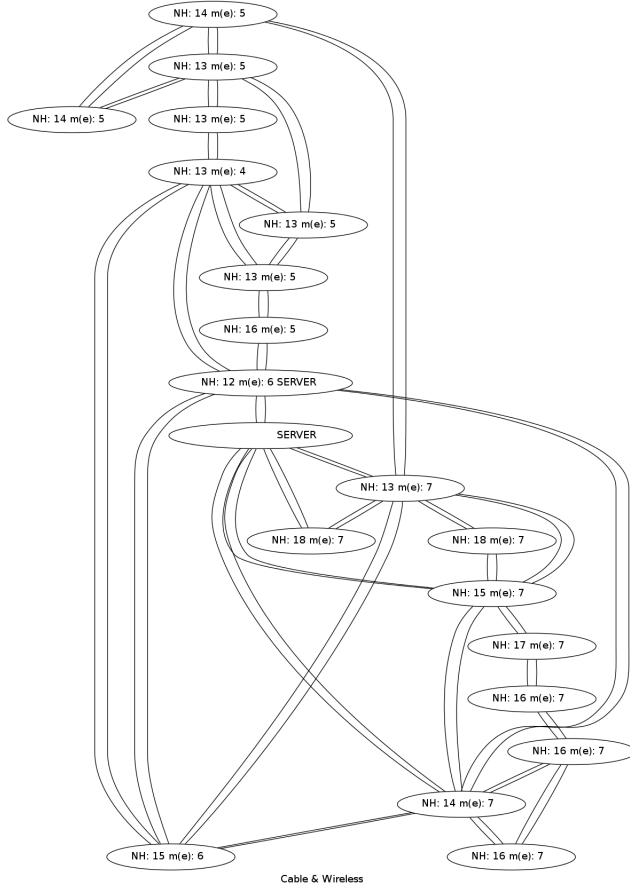
Wanted: *A set of Servers $S = \{s_1, s_2, \dots, s_k\} \subseteq V$ such that $|V_i| \leq r$ ($i = 1, 2, \dots, k$) where V_i is the neighbour set of s_i ($i = 1, 2, \dots, k$) and that $m(e) \leq c$ ($\forall e \in E$).*

3.2 An Instance Of The Problem



The above shown graph is from the barebone network "Cable and Wireless". It is parsed out of a dataset of barebone networks provided by caida.[Cai13] The nodes have two values in their bubbles: "NH and $m(e)$ ". NH stands for the maximum neighbourhood if that node was chosen as a server. Choosing the first server obviously implicates the maximum neighbourhood, because there are no mirror servers yet on the graph.

The multiplicity of an edge e , $m(e)$ is explained above. The number after $m(e)$ in the bubble describes $m(e) \forall e \in E$, not only for a single edge, if one decides to take this node as a server. The greedy algorithm takes the node with the lowest max $m(e)$ first. The great numbers for $m(e)$ are due to more than one shortest path to the server.



This graph shows the second server pick. This time our $m(e)$ is already low enough to suffice the constraint $m(e) \leq c$ thus the greedy algorithm proposed in the paper by Miwa et al. [NM11] chooses the node with the lowest max neighbourhood.

3.3 Proof of NP-Completeness

4 Algorithm

4.1 Rules and Assumptions

4.2 Other Algorithms

The only prior algorithm for this problem is the greedy algorithm proposed by Miwa et al. [NM11]. It greedily chooses the node with the lowest maximal neighbourhood if the constraint $c \geq m(e)(\forall e \in E)$ is fulfilled. Otherwise it will choose a node where $\min(m(e)(\forall e \in E))$ is true.

4.3 Data (Testinstances)

The data taken for the tests is from caida [Cai13]. Parsing the data was one of the most difficult parts, because the data is not formatted well. It does not really follow the syntax proposed in the beginning of the file. I also modified the data in some concerns:

- If the graph G given for the barebone network was not connected, the connected component with less nodes of that graph is cut out.
- Normally the connected cities are listed in this manner: "city,state,country", in order to match with the numbers in the paper [NM11] only city and state are used for identification and the country is left out. Sometimes London is used on one hand as City in Canada and on the other hand in GB, denoted like this London, *, Canada and London, *, England. They are differentiated.
- Sometimes the State has a number. One example would be "Kansas City, KS2 " This is interpreted as "Kansas City, KS".

All those modifications were done to match the graphs proposed by [NM11]. As I did not have the data, I tried to match them with the number of nodes and number of vertices. Unfortunately three instances of my instances do not match the number of nodes and / or vertices proposed from the paper above. The varying instances are: "Sprint", "Cable and Wireless" and "UUNet".

4.4 Implementation

5 Conclusion

References

- [BPV08] Rajkumar Buyya, Mukaddim Pathan, and Athena Vakali. *Content Delivery Networks*. Springer Publishing Company, Incorporated, 1st edition, 2008.
- [Cai13] Caida. Barebones - <http://www.caida.org/tools/visualization/mapnet/data/inc.txt>, 2013. [Online; accessed 9-September-2013].
- [NM11] R. Nakamura and H. Miwa. Method of locating mirror servers to alleviate load on servers and links. In *Applications and the Internet (SAINT), 2011 IEEE/IPSJ 11th International Symposium on*, pages 513–518, 2011.
- [Wik13a] Wikipedia. Content delivery network, 2013. [Online; accessed 4-September-2013].
- [Wik13b] Wikipedia. Last mile, 2013. [Online; accessed 4-September-2013].