JST: Smart Systems and Devices
Volume 34, Issue 1, January 2024, 009-015

Solving Resource Allocation Problem in Wifi Network by Dantzig-Wolfe Decomposition Algorithm

Ta Anh Son**, Nguyen Thi Thuy
School of Applied Mathematics and Informatics, Hanoi University of Science and Technology, Ha Noi, Vietnam
**Corresponding author email: son.taanh@hust.edu.vn*

Abstract

Online advertising and advancements is a recent trend in marketing technology, in this context we consider a new form of contract which allows advertisers to specify in the Wifi system. Based on the structure of the system, we have to organize and manage resource allocation such that the guaranteed display is satisfied. We introduce a new mathematical model and develop an optimization framework that aims to optimize "fairness" of allocation each campaign over its targeted location. Because of large scale problem, the Dantzig-Wolfe decomposition is proposed for solving it. Dantzig-Wolfe decomposition is a technique for dealing with large scale linear programming and modified to solve linear integer programming, nonlinear programming. Especially, it is used mostly in linear programming when its size is very large, and its structure is appropriate. The technique has been successfully applied in a variety of contexts. In this paper, we introduce a new model of a resource allocation problem in Wifi network and represent Dantzig-Wolfe decomposition for solving this problem by dividing the number of advertisement impressions when users access the Wifi network. The numerical simulation shows the efficiency of our proposed method.

Keywords: Dantzig-Wolfe decomposition, resource allocation, wifi network, online advertising.

1. Introduction

Wifi marketing refers to the use of Wifi technology to promote products, services, or a brand to potential customers. This type of marketing involves collecting data about Wifi users and using it to deliver targeted advertising, promotional messages, or other forms of content to their devices while they are connected to a Wifi network. Wifi marketing is typically done through a captive portal, which is a landing page that users are redirected to when they connect to a Wifi network. The captive portal can be customized with branding, advertising, and other types of content, and can also include a login process that requires users to provide some personal information or complete a survey before they can access the network. Wifi marketing has become increasingly popular in recent years because it allows businesses to engage with customers in a more targeted and personalized way. By collecting data about Wifi users, businesses can gain valuable insights into their preferences, behaviors, and needs, which can be used to create more effective marketing campaigns and improve the overall customer experience.

Online advertising (see [1]), also known as digital advertising, is a form of advertising that uses the internet to promote products, services or brands. Online advertising is delivered through various digital channels such as search engines, social media, display advertising, email marketing, mobile advertising and video advertising. Online advertising provides businesses with the opportunity to reach a large and diverse audience, as well as to target specific demographics based on factors such as age, gender, location, interests and online behavior. This makes online advertising a highly effective and cost-efficient way to reach potential customers. Online advertising has become increasingly important in recent years, as more and more people spend time online and less time consuming traditional media like TV, radio and print. As a result, businesses are shifting their advertising budgets towards digital channels, in order to reach their target audience where they are spending their time. Overall, online advertising has revolutionized the way businesses promote their products and services and is a critical component of any modern marketing strategy.

Online graphical presentation promoting is a type of internet publicizing where sponsors can unequivocally or certainly target clients visiting Web pages, and show graphical (e.g., picture, video) advertisements to those clients (see [2]). Likewise with most types of web based publicizing (see [3]), one of the focal inquiries that emerges with regards to online graphical presentation promoting is that of resource allocation, i.e., deciding how to allocate supply/inventory (client visits) to request (advertiser campaigns) in order to upgrade for different publisher and advertiser destinations. However, even formulating the inventory allocation problem for online graphical display advertising is quite challenging (see [4-6]).

ISSN: 2734-9373
https://doi.org/10.51316/jst.171.ssad.2024.34.1.2
Received: February 22, 2023; accepted: September 19, 2023

Allocating resources is frequently not fair in practice. Less or more, depending on the availability of resources. A network of free Wifi spots is maintained by a Wifi marketing company. The company's resources are the number of users accessing this system. Advertisements for any partner brands will appear when users log in to the free Wifi system. To optimize the system, exposure must be scheduled across multiple locations and campaigns.

The problem is to divide the number of advertisement impressions to users at places so that during the days when the campaign wants to display, the number of impressions at each day is as *fair* as *possible*. We concentrate on minimizing the sum of difference between advertisement impressions and average number of expected runs. This is a practical large-scale problem, so we introduce a new model of a resource allocation problem in Wifi network and proposed to use Dantzig-Wolfe decomposition to solve this problem.

The remainder of the paper is organized as follows. In Section 2 we introduce a resource allocation problem in Wifi network and present a new mathematical model of this problem. A brief introduction of Dantzig-Wolfe Decomposition Algorithm is organized in Section 3. Section 4 shows the numerical experiment, and some conclusions are presented in Section 5.

2. Resource Allocation Problem in Wifi Network

Resource allocation problem in Wifi networks refers to the challenge of efficiently distributing available network resources, such as bandwidth, power, and channel capacity… to multiple users who are competing for these resources (see [1, 5]). This is a critical issue in Wifi networks because there is often more demand for network resources than supply, and users have varying levels of priority and requirements for these resources.

However, resource allocation in Wifi networks is a complex problem due to several factors such as varying network conditions, the presence of interference, and the number of users… As a result, resource allocation algorithms must be designed to be adaptive and dynamic, taking into account changing network conditions and user behavior to ensure optimal performance. In this paper, a resource allocation in Wifi networks related to Wifi maketing is introduced as follows.

2.1. Problem

A free Wifi system that displays advertisement when customers access at p places $P = \{P_1, P_2, \ldots, P_p\}$. These places are divided into r sets of non-intersecting places $T = T[1] \cup T[2] \cup \ldots \cup T[r]$ with $T[t] \cap T[k] = \varnothing$ $t \neq k$, $1 \leq t, k \leq r$. Each place has the number of accesses from day 1 to day n is $R_{11}, R_{12}, \ldots, R_{1p}, R_{21}, \ldots, R_{2p}, \ldots, R_{n1}, \ldots, R_{np}$. There are m advertising campaigns, C_i ($i = 1, 2, \ldots, m$), needed to display at P on days $1, 2, \ldots, n$. For each campain C_i, let us set

- q_i is the number of expected runs of campaign C_i,
- $D(C_i) = D_i = \{ j_{i_1}, j_{i_2}, \ldots, j_{i_n} \} \subset \{1, 2, \ldots, n\}$ is the set of days campaign C_i will run,
- $P(C_i) = P_i = \{ k_{i_1}, k_{i_2}, \ldots, k_{i_p} \} \subset \{1, 2, \ldots, p\}$ is the set of places campaign C_i will run.

The problem is finding ways to divide the number of advertisement impressions for each day and each campaign in the most *fair* manner. It mean that the target is the number of impressions per campaign per day, per location, proportional to the resource per day and per location respectively. The objective focuses on minimizing the sum of difference between advertisement impressions and average number of expected runs of each campain under the limited resources constraints.

2.2. Model

Consider each set of places to be a sub-problem of the original problem. At the t^{th} set of places, q^t_{ijk} is the number of advertisements of campaign i run on day j at place $k \in T[t]$.

Select the objective function as the sum of difference between advertisement impressions and average number of expected runs of each campain. So that for the k_i days running campaign C_i, the daily runs are as *fair* as possible, then we calculate the total across all advertising campaigns C_i. The objective function is selected as

$$\min \sum_{t=1}^{r} \sum_{i=1}^{m} \sum_{j \in D_i} \sum_{k \in P_i \cap T[t]} \frac{v_i}{2a^t_{ijk}} \left| q^t_{ijk} - a^t_{ijk} \right| \tag{1}$$

where v_i is the priority of campaign C_i and the weighted average of expected runs of campaign each day at each place be:

$$a^t_{ijk} = \frac{q_i R_{jk}}{\sum_{x \in D_i, y \in P_i} R_{xy}} \quad \forall i \in \{1, 2 \ldots, m\}, j \in D_i, k \in P_i \tag{1a}$$

Suppose that $y^t_{ijk} \geq 0$ satisfy:

$$\frac{v_i}{2a^t_{ijk}} \left| q^t_{ijk} - a^t_{ijk} \right| \leq y^t_{ijk} \tag{2}$$

Then objective function (1) can be rewriten as:

$$\Leftrightarrow \min \sum_{t=1}^{r} \sum_{i=1}^{m} \sum_{j \in D_i} \sum_{k \in P_i \cap T[t]} y^t_{ijk} \tag{3}$$

subject to the following set of constraints. Consider the places $T[1]$, we have the set of constraints:

$\forall i \in \{1,2...,m\}, j \in D_i, k \in P_i \cap T[1]$

$$q^1_{ijk} - \frac{2a^1_{ijk}}{v_i} y^1_{ijk} \le a^1_{ijk} \quad (4)$$

$\forall i \in \{1,2...,m\}, j \in D_i, k \in P_i \cap T[1]$

$$-q^1_{ijk} - \frac{2a^1_{ijk}}{v_i} y^1_{ijk} \le -a^1_{ijk} \quad (5)$$

$\forall j \in \{1,...,n\}, k \in T[1]$

$$\sum_{i=1}^{m} \sum_{j \in D_i} \sum_{k \in P_i \cap T[1]} q^1_{ijk} \le R_{jk} \quad (6)$$

In a simialar way, we have the set of constraints for the set of places T[2], …, T[r], repestively.

…

$\forall i \in \{1,...,m\}, j \in D_i, k \in P_i \cap T[r]$

$$q^r_{ijk} - \frac{2a^r_{ijk}}{v_i} y^r_{ijk} \le a^r_{ijk} \quad (7)$$

$\forall i \in \{1,...,m\}, j \in D_i, k \in P_i \cap T[r]$

$$-q^r_{ijk} - \frac{2a^r_{ijk}}{v_i} y^r_{ijk} \le -a^r_{ijk} \quad (8)$$

$\forall j \in \{1,...,n\}, k \in T[r]$

$$\sum_{i=1}^{m} \sum_{j \in D_i} \sum_{k \in P_i \cap T[r]} q^r_{ijk} \le R_{jk} \quad (9)$$

If $j \notin D_i \; or \; k \notin P_i \cap T[t], \; t = \{1,...,r\}$

$$q^t_{ijk} = 0 \quad (10)$$

$$\sum_{t=1}^{r} \sum_{j \in D_i} \sum_{k \in P_i \cap T[t]} q^t_{ijk} = q_i \qquad \forall i \in \{1,...,m\} \quad (11)$$

$$q^t_{ijk}, y^t_{ijk} \ge 0 \quad (12)$$

We have two constraints (4) and (5) represent for (2). Constraint (6) represents the total number of ads displayed on day j at location k does not exceed the number of visitors on day j at the corresponding location k. For the set places T[r], we have the set of constraints as (7), (8) and (9). Constraints (10) show that if campaign C_i is not display on day j or at place k, the number of advertisements of campaign i run on day j or at place k is 0. Contraints (11) reperesents the total display of campain C_i equals to the number of expected runs. Constraints (12) are sign constraints.

According to structure of linear programming with complicating constraints, for each independent set of places t *(t=1,…, r)* is a sub-problem t with the set of constraint similar as (4), (5), (6) for the set of place T[t]:

$$\min \sum_{i=1}^{m} \sum_{j \in D_i} \sum_{k \in P_i \cap T[t]} y^t_{ijk} \quad (P^t)$$

subject to

$\forall i \in \{1,2...,m\}, j \in D_i, k \in P_i \cap T[t]$

$$q^t_{ijk} - \frac{2a^t_{ijk}}{v_i} y^t_{ijk} \le a^t_{ijk} \quad (13)$$

$\forall i \in \{1,2...,m\}, j \in D_i, k \in P_i \cap T[t]$

$$-q^t_{ijk} - \frac{2a^t_{ijk}}{v_i} y^t_{ijk} \le -a^t_{ijk} \quad (14)$$

$\forall j \in \{1,...,n\}, k \in T[r]$

$$\sum_{i=1}^{m} \sum_{j \in D_i} \sum_{k \in P_i \cap T[1]} q^t_{ijk} \le R_{jk} \quad (15)$$

If $j \notin D_i \; or \; k \notin P_i \cap T[t], \; t = \{1,...,r\}$

$$q^t_{ijk} = 0 \,, \quad (16)$$

$$q^t_{ijk}, y^t_{ijk} \ge 0 \,. \quad (17)$$

And the complicating constraints (11), if these constraints are removed, the original problem be come a serial of separate r linear programming problem (P^t) *(t=1, …, r)*, are explained as follows:

$$\sum_{t=1}^{r} \sum_{j \in D_i} \sum_{k \in P_i \cap T[t]} q^t_{ijk} = q_i \qquad \forall i \in \{1,...,m\} \,. \quad (18)$$

This problem is a large-scale linear programming problem, then we propose to use Dantzig-Wolfe Decomposition to decompose and solve a series of small linear programming problem to obtain the optimal solution.

3. Dantzig - Wolfe Decomposition Algorithm

The Dantzig-Wolfe Decomposition Algorithm, also known as the column generation algorithm, is a powerful optimization technique used to solve large-scale linear programming problems. The algorithm was developed by George Dantzig and Philip Wolfe in the 1960s. The Dantzig-Wolfe Decomposition Algorithm works by breaking a large linear programming problem into smaller subproblems, which can be solved independently and in parallel. This approach is particularly useful for problems with a large number of variables or constraints, as it allows for more efficient use of computational resources (see [7-9]). The algorithm involves solving a master problem, which contains a subset of the original variables and constraints, and a set of subproblems, each of which contains a different subset of variables and constraints. The subproblems are then solved separately, and their solutions are combined to provide a solution to the master problem. This process is repeated iteratively, with new variables and constraints added to the master problem until a feasible solution is found. The Dantzig-Wolfe Decomposition Algorithm is widely used in a variety of applications, including transportation and logistics, network design, and production planning. The algorithm has been shown to be highly effective in solving complex optimization problems, and it is often used in conjunction with other techniques, such as branch and bound or heuristics, to further improve solution quality and efficiency.

3.1 Problem Structure

Consider the linear programming problem:

$$\min_{x^1,\dots,x^r} \left(c^1\right)^T x^1 + \left(c^2\right)^T x^2 + \dots + \left(c^r\right)^T x^r \qquad \text{(O)}$$

subject to

$$A^1x^1 \qquad\qquad = b^1, \qquad (19)$$

$$A^2x^2 \qquad = b^2, \qquad (20)$$

$$\dots$$

$$A^rx^r = b^r \qquad (21)$$

$$L^1x^1 + L^2x^2 + \dots + L^rx^r = b^0, \qquad (22)$$

$$0 \le x^i \le x^{i(up)}, \qquad i = 1, \dots, r. \qquad (23)$$

where constraints (19) - (21) have a decomposable structure in r blocks, each of size n_i (i = 1,2,...,r), i.e, they can be written as

$$x^i \in \mathbb{R}^{n_i}, \ A^i \in \mathbb{R}^{m_i \times n_i}, \ b^i \in \mathbb{R}^{m_i} \quad \forall i \in \{1,\dots,r\}.$$

The structure of the matrix representing the problem (O) is described as shown in Fig. 1. The constraints (22) is seen as the last row in the matrix structure and they do not have decomposable structure, they are the complicating constraints.

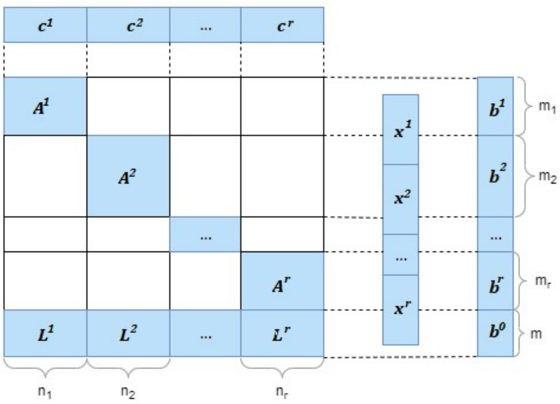

Fig. 1. Decomposable matrix with complicating constraints.

Upper bounds $x^{i(up)}$ ($i = 1, \dots, r$) are considered for all optimization variables x^i ($i = 1, \dots, r$). This assumption allows dealing with a compact feasible region, leading to a simpler theoretical analysis of the original problem.

If ignoring complicating constraints, there will be a relaxed version of the original problem:

$$\min_{x^1,\dots,x^r} \sum_{i=1}^{r} \left(c^i\right)^T x^i \qquad \text{(RP)}$$

subject to $A^ix^i = b^i, \quad i=1, \dots, r$

$0 \le x^i \le x^{i(up)}$, $i=1, \dots, r$.

Therefore, the decomposed i^{th} sub-problem ($i = 1,2, \dots, r$) is

$$\min_{x^i} \left(c^i\right)^T x^i \qquad \text{(RP}^i\text{)}$$

subject to $A^ix^i = b^i$

$0 \le x^i \le x^{i(up)}$.

3.2 Decomposition

3.2.1. Master problem: Let $c = [c^1\ c^2\ \dots\ c^r] \in R^n$, $x = [x^1\ x^2 \dots x^r]$, L = $[L^1\ L^2 \dots L^r]$. The feasible region of (RP) is

$$X = \{x \in R^n \mid Ax = b,\ 0 \le x \le x^{up}\}$$

and (O) problem is equivalent to the following problem:

$$\min_x c^T x \qquad \text{(O')}$$

subject to $\mathcal{L}x = b^0$

$$x \in X$$

Since X is a convex polyhedron, by polyhedral convex set representation theorem then x can be represented as a convex combination of the vertices of X, so (O') can be represented in the form of linear programming:

$$\min \sum_{j \in J} \alpha_j c^T x^{(j)} \qquad (24)$$

$$\text{subject to} \quad \sum_{j \in J} \alpha_j \mathcal{L} x^{(j)} = b^0 \qquad (\lambda) \qquad (25)$$

$$\sum_{j \in J} \alpha_j = 1 \qquad (\sigma) \qquad (26)$$

$$\alpha_j \ge 0 \quad \forall j \in J \qquad (27)$$

where λ and σ are the corresponding dual variables, J is the set of extreme points of X.

Suppose that $|J| = p$, problem (24) - (27) is equivalent to the following problem called Master problem (MP)

$$\min \sum_{s=1}^{p} \alpha_s c^T x^{(s)} \qquad \text{(MP)}$$

$$\text{subject to} \quad \sum_{s=1}^{p} \alpha_s \mathcal{L} x^{(s)} = b^0 \qquad (\lambda) \qquad (28)$$

$$\sum_{s=1}^{p} \alpha_s = 1 \qquad (\sigma) \tag{29}$$

$$\alpha_s \geq 0 \qquad s = 1, ..., p. \tag{30}$$

Note that, superindices of form (s) to refer to the s^{th} extreme points in J.

3.2.2 Relaxed problem: The reduced cost of the new weighting variable α corresponds to the tentative new basic feasible solution can be computed as

$$c^T x - \begin{bmatrix} \lambda^T & \sigma \end{bmatrix} \begin{bmatrix} \mathcal{L}x \\ 1 \end{bmatrix} = \left(c^T - \lambda^T \mathcal{L}\right) x - \sigma. \tag{31}$$

If the tentative basic feasible solution is to be added to the set of previous ones, the reduced cost associated with its weighting variable should be negative and preferably a minimum. To find that minimum reduced cost and basic feasible solution, we solve the relaxed problem

$$\min \left(c^T - \lambda^T \mathcal{L}\right) x - \sigma \tag{32}$$

$$\textit{subject to} \quad Ax = b, \quad 0 \leq x \leq x^{up}. \tag{33}$$

This problem can also be expressed as independent subproblems according to the initial constraint blocks. Moreover, σ is a constant, so when removed, the solution of the problem does not change. Therefore, we have a new problem equivalent to the problem (32) - (33) as

$$\min_{x^1,...,x^r} \sum_{i=1}^{r} \left(\left(c^i\right)^T - \lambda^T L^i \right) x^i \tag{SP}$$

$$\textit{subject to} \quad A^i x^i = b^i, \qquad i = 1, 2, ..., r, \tag{34}$$

$$0 \leq x^i \leq x^{i(up)} \qquad i = 1, 2, ..., r. \tag{35}$$

3.3 Dantzig - Wolfe Decomposition Algorithm

The detailed steps of the algorithm are presented below:

Step 0: *Initialization.*

Let $k = 1$ - the iteration counter.

Obtain p distinct solutions of the relaxed problem by solving p times each of the r sub-problems below:

$$\min_{x_i} \left(\hat{c}_s^i\right)^T x^i$$

$$\textit{subject to} \quad A^i x^i = b^i$$

$$0 \leq x^i \leq x^{i(up)} \qquad i = 1, 2, ..., r.$$

where, $\hat{c}_s^i \left(i = 1,...,r;\ s = 1,...,p\right)$ are arbitrary cost coefficients to attain the p initial solutions of the r sub-problems.

Step 1: *Solve master problem.*

Solve (MP), (28)-(30) problem obtain $\alpha_1^{(k)},...,\alpha_p^{(k)}$ and dual vari-ables $\lambda^{(k)}$, $\sigma^{(k)}$.

Step 2: *Solve relaxed problem.*

With $\lambda^{(k)}$ solve (SP), (34)-(35) obtain solution of relaxed problem $x^{(k+1)} = [(x^1)^{(k+1)} \ ... \ (x^r)^{(k+1)}]$ and objective function value

$$g^{(k)} = \sum_{i=1}^{r} \left(\left(c^i\right)^T - \left(\lambda^{(k)}\right)^T L_i \right) \left(x^i\right)^{(k+1)}$$

Step 3: *Check STOP conditions.*

If $g^{(k)} \geq \sigma^{(k)}$, the original problem has an optimal solution

$$x^{j*} = \sum_{s=1}^{p} \alpha_s^{(k)} \left(x^j\right)^{(s)}, \qquad j = 1,...,n$$

and the algorithm concludes.

Else if $g^{(k)} < \sigma^{(k)}$, add $x^{(k+1)}$ to the set of extreme points.

Update

$k \leftarrow k + 1$

$p \leftarrow p + 1$

Return Step 1.

4. Numerical Experiment

The program was developed using Python language and CPLEX 12.8 solver on a personal computer with a Core i5-5200U 2.2GHz processor and 16G RAM.

4.1. Data 1

To test the Dantzig - Wolfe Decomposition Algorithm, we consider a small data (Data1: 2 campaigns and 2 places) given in Table 1.

Table 1. Information of data 1

	Day 1	Day 2
place - p1	500	500
place - p2	500	500

Campaign	Priority	Start day	End day	Total	Places
c1	0.3	1	2	500	p1, p2
c2	0.7	2	2	100	p1

Consider two cases to compare results:

- Problem does not decompose, i.e, only one set of places.

- Problem decomposes into two sets of places, $\{p_1\}$ and $\{p_2\}$. In this case, problem (3) - (12) is solved by the Dantzig - Wolfe Decomposition Algorithm..

Table 2. Results after running data 1

Campaign	Day 1		Day 2		Sum
	p1	p2	p1	p2	
c1	125	125	125	125	500
c2	0	0	100	0	100

TH1. The result of the non-decomposition problem.

Campaign	Day 1		Day 2		Sum
	p1	p2	p1	p2	
c1	125	125	125	125	500
c2	0	0	100	0	100

TH2. The result of the decomposition problem

Table 3. Results after running data 2

Campaign	q	[51]		[25, 26]	
		Sum	Error	Sum	Error
c0	100000	100000	**0.0**	100016	**0.00016**
c1	98310	98310	**0.0**	98311	**0.00001**
c2	18342	18342	**0.0**	18037	0.01663
c3	89585	89585	**0.0**	89585	**0.0**
c4	84370	84370	**0.0**	78758	0.06652
c5	100000	100000	**0.0**	99999	**0.00001**
c6	100000	100000	**0.0**	100003	**0.00003**
c7	100000	100000	**0.0**	100003	**0.00003**
c8	95774	95774	**0.0**	95772	**0.00002**
c9	100000	100000	**0.0**	100001	**0.00001**

Campaign	q	[10, 41]		[41, 10]	
		Sum	Error	Sum	Error
c0	100000	100015	**0.00015**	100016	**0.00016**
c1	98310	98310	**0.0**	98311	**0.00001**
c2	18342	18128	0.01167	18057	0.01554
c3	89585	89585	**0.0**	89585	**0.0**
c4	84370	78667	0.0676	78738	0.06675
c5	100000	100000	**0.0**	100001	**0.00001**
c6	100000	100005	**0.00005**	100003	**0.00003**
c7	100000	100003	**0.00003**	100004	**0.00004**
c8	95774	95772	**0.00002**	95772	**0.00002**
c9	100000	100001	**0.00001**	100000	**0.0**

Campaign	q	[13, 13, 13, 12]		[2, 3, 4, 42]	
		Sum	Error	Sum	Error
c0	100000	100017	**0.00017**	100017	**0.00017**
c1	98310	98311	**0.00001**	98312	**0.00002**
c2	18342	18342	**0.0**	18098	0.01330
c3	89585	89585	**0.0**	89585	**0.0**
c4	84370	78453	0.07013	78697	0.06724
c5	100000	100001	**0.00001**	99999	**0.00001**
c6	100000	100004	**0.00004**	100004	**0.00004**
c7	100000	100004	**0.00004**	100003	**0.00003**
c8	95774	95772	**0.00002**	95772	**0.00002**
c9	100000	100001	**0.00001**	99999	**0.00001**

According to the structure of problem (3) - (12) have two following accses:

Case 1: We need to find the value of variables:

$q_{111}^1, q_{112}^1, q_{121}^1, q_{122}^1, q_{221}^1 \geq 0$

and solutions of this problem in this case are

$q_{111}^1 = q_{112}^1 = q_{121}^1 = q_{122}^1 = 125, q_{221}^1 = 100$

Case 2: We need to find the value of two sets of variables:

$\{q_{111}^1, q_{121}^1, q_{221}^1\}$ *and* $\{q_{112}^2, q_{122}^2\}$. They are $\{125,125,100\}$ and $\{125,125\}$.

The solution of the problem presented as required by the campaigns is shown in the following Table 2, which shows that either the non-decomposition problem or decomposition problem are guaranteed to run on time and sufficient number of required campaigns.

4.2. Data 2

We will increase the size of the problem: 10 campaigns, run in 51 days, at 51 places (Data 2). The result of this case is shown in Table 3.

Table 3 shows good results, due to the convergence properties of the Dantzig - Wolfe Decomposition Algorithm, which has been shown for the linear programming problem, the error between the total number of campaigns run and the number of requests is small (0 - 7%):

- When we consider a special case, the problem has only one set of places ([51]) and shape is 8640 × 6702. In this case, the problem has a nearly complete solution and error of campaigns are 0%. It means that the result of solving the original linear programming is almost the same with using Dantzig - Wolfe Decomposition Algorithm.

- Decompose 51 places into 2 sets of places with 3 cases [25,26], [10,41], [41,10] and into 4 sets of places with 2 cases [13,13,13,12], [2,3,4,42]; classifying the places into equal or unequal sets, the results are quite stable.

From the numerical simulation, we see that for various ways to decompose and using Dantzig - Wolfe Decomposition, the results are quite good: all the gap are less than 0.1 (10%), 52/60 datasets have gap less than 1% (**the bold errors** in Table 3). The results is promising to apply this method to the large-scale problem.

5. Conclusion

In this paper, we proposed a new model for allocating and serving online advertising for the Wifi marketing company.

Our structure brings a client level viewpoint into the normal total displaying of the advertisement allocation problem in the Wifi networt systerm. Our formulation can optimize for the allocation of each campaign in its targeted locations. We showed that the problem can be solved efficiently using Dantzig-Wolfe Decomposition Algorithm and promising to apply in the very large-scale problem.

References

[1] Hana Choi, Carl F. Mela, Santiago R. Balseiro, Adam Leary, Online display advertising markets: A literature review and future directions, Information Systems Research, vol. 31, No. 2, 2020. https://doi.org/10.1287/isre.2019.0902

[2] Hana Choi, Carl Mela, Santiago R.Balseiro, Adam Leary, Online display advertising markets: A literature review and future directions, SSRN Electronic Journal, Jan. 2019. https://doi.org/10.2139/ssrn.3070706

[3] A. Z. Broder, Computational advertising and recommender systems, In Proceedings of the 2008 ACM conference on Recommender systems, 2008. https://doi.org/10.1145/1454008.1454009

[4] J. Yang, E. Vee, S. Vassilvitskii, J. Tomlin, J. Shanmugasundaram, T. Anastasakos, O. Kennedy, Inventory allocation for online graphical display advertising, arXiv:1008.3551, Aug. 2010. https://doi.org/10.48550/arXiv.1008.3551

[5] J. Yang, E. Vee, S. Vassilvitskii, J. Tomlin, J. Shanmugasundaram, T. Anastasakos and O. Kennedy, Inventory allocation for online grahpical display Advertising using multi-objective optimization, In Proceedings of the 1st International Conference on Operations Research and Enterprise Systems, pp. 293-304, 2012. https://doi.org/10.5220/0003752802930304

[6] Hong Zhang, Lan Zhang, Lan Xu, Xiaoyang Ma, Zhengtao Wu, Cong Tang, Wei Xu, Yiguo Yang, A Request-level guaranteed delivery advertising planning: forecasting and allocation, Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20), pp. 2980-2988, Aug. 2020. https://doi.org/10.1145/3394486.3403348

[7] James Richard Tebboth, A Computational study of Dantzig-Wolfe Decomposition, Thesis submitted for the degree of Doctor of Philosophy in the University of Buckingham 2001.

[8] Antonio J. Conejo, Enrique Castillo, Roberto M´ınguez, Raquel Garc´ıaBertrand, Decomposition Techniques in Mathematical Programming, Springer-Verlag Berlin Heidelberg, The Netherlands, 2006. https://doi.org/10.1007/3-540-27686-6

[9] Ted Ralphs, Anahita Hassanzadeh, Jiadong Wang, Matthew Galati, Menal Gu¨zelsoy, Scott Denegre, Decomposition methods for discrete optimization, In Proceeding INFORMS Computing Society Conference, Jan. 2013.