

RESEARCH ARTICLE

## A mathematical model for personnel task assignment problem and an application for banking sector

Kenan Cetin <sup>a,b</sup>, Gulfem Tuzkaya <sup>b\*</sup>, Ozalp Vayvay <sup>c</sup>

<sup>a</sup> R&D Center, Vakıfbank, Turkey

<sup>b</sup> Department of Industrial Engineering, Marmara University, Turkey

<sup>c</sup> Department of Business Administration, Marmara University, Turkey

kenan.cetin@vakifbank.com.tr, gulfem.tuzkaya@marmara.edu.tr, ozalp@marmara.edu.tr

---

### ARTICLE INFO

#### Article history:

Received: 26 May 2019

Accepted: 26 October 2019

Available Online: 7 April 2020

#### Keywords:

Generalized assignment problem

Task assignment

Analytical hierarchy process

Banking sector

Linear physical programming

AMS Classification 2010:

90-08, 65K10, 90C10

### ABSTRACT

Efficient planning and management of the workforce resources is one of the most essential requirements for the companies operating in the service sector. For banks, a large number of transactions comes to Central Operations Department from the branches or directly from the customers and their aim is to provide the best operational service with the highest efficiency with the limited workforce resources in the departments. In this study, a real assignment problem was discussed and the problem was considered as Generalized Assignment Problem. For the solution of the problem, related algorithms were listed and examined in the literature survey section. Then, a two-step method is proposed. First step prioritizes the task coming to the system by considering the customer types, service level agreement (SLA) times, cut-off times, task type. In the second step, a multi-objective mathematical model was developed to assign task to employee groups. A preference based optimization method called Linear Physical Programming (LPP) is used to solve the model. Afterward, proposed model was tested on real banking data. For all the tests, GAMS was used as a solver. Results show that proposed model gave better results compared with current situation. With the proposed solution method, the workloads of the profile groups working above their capacity were transferred to other profile groups with idle capacity. Thus, the capacity utilization rates of the profile groups were more balanced and the minimum capacity utilization rate was calculated as 41%.



### 1. Introduction

It is a well-known fact that workforce is the most important resource for the companies operating in the service sector. A company's success or failure depends mainly on the skill level of the people working for it. Without positive and creative employee contributions, organizations are unable to advance and thrive. Thus, they need to recruit employees with the necessary abilities, experience and capabilities to accomplish a company's objectives or events. In this way, both the present and the company's future requirements should also be kept in mind. Therefore; effective and efficient utilization, planning and directing of the workforce resource are the most essential requirements. To enable companies to respond quickly to its clients by managing their available workforce resource effectively, some major challenges related to business

and marketing constraints are needed to be considered such as; number of available workforce, customer segmentation, SLA times and cut-off times, operation type and their processing time, workforce competence, number of operations executed by customers and priority score.

This study's main goal is to develop a task assignment methodology that based on optimization techniques which assigns a set of jobs to a set of employees with different levels of expertise to meet the due dates and satisfy SLAs. As a result, the proposed model aims to assign the proper number of workforce to the appropriate jobs by considering competence, experience and other capabilities of employees, and also prioritize the incoming jobs considering some criteria such as; customer types, amount of money, SLAs, cut-off times and operation type.

\*Corresponding author

In this study, a two-step method is proposed to solve a real life assignment problem. First step prioritizes the jobs coming to the system based on a multi-criteria evaluation. In the second step, a mathematical model is developed to assign jobs to employee groups. The methodology has been proposed in order to make the best assignment in the best way considering different objectives. Our mathematical model has three different objectives. The first objective function seeks to assign tasks to the most appropriately qualified employee. The second objective tries to maximize assignment level of higher priority tasks. Although all tasks are required to be completed, the workload of the employees also wanted to be more balanced. Therefore; the third objective function tries to balance the workloads of the profile groups. After that, the jobs assigned to the related employee groups are pushed to the employees according to the priority score calculated in the first step.

The rest of this paper is organized as follows: A definition of the GAP and an overview of related works is given in Section 2. Solution methodology is explained in Section 3. Section 4 describes the methodology and algorithm proposed to solve the problem. The computational results and conclusions are given in Section 5.

## 2. Literature review

The assignment problem which is the subject of our study is called The Generalized Assignment Problem (GAP) in the literature. In a simple definition, the Generalized Assignment Problem (GAP) is the problem of assigning a set of tasks to a set of agents with a minimum total cost. In each agent, there is a single resource and the resources in the agents have limited capacity. Each tasks that are assigned to an agent, needs a certain number of resource. The generalized assignment problem (GAP) is a well-known, NP-complete combinatorial optimization problem [1]. The first study for GAP in literature is proposed by Kuhn [2]. GAP has been applied in many real world problems ranging from job assignment from computer networks to machine loading in flexible manufacturing systems [3-6].

Several optimization and the approximation algorithms are proposed in order to solve the GAP effectively in the literature. Osman [7] has presented  $\lambda$ -generation mechanism. In this paper, different kind of parameter settings and search methods were examined for hybrid Simulated Annealing (SA) and Tabu Search (TS) algorithms. The results of this technique is compared with SA, branch and bound algorithm and set partitioning heuristics. A genetic algorithm (GA) which tries to improve feasibility and optimality simultaneously was presented by Chu and Beasley [8]. This algorithm was applied on a set of relatively large 84 test problems with 20 agents and 200 jobs. The 60 of these problems were accepted as small-size and optimal solutions can be found. Racer and Amini [9]

presented a hybrid heuristic (HH) method which consists of Variable-Depth-Search Heuristic (VDSH) and Heuristic GAP (HGAP). The HH was tested on 450 test problems and after all, it is found that VDSH gives better solutions, HGAP gives results quickly. Laguna et al. [10] proposed a new heuristic approach to solve the multi-level generalized assignment problem (MGAP). MGAP is different from the classical GAP. Lot sizing problem can be formulated as MGAP. An optimum solution cannot be found by using commercial solvers. Therefore, a new heuristic approach is presented to overcome this problem and also, this approach involves TS applications with neighbourhood search mechanism defined by ejection chains. A Tabu Search Heuristic presented by Diaz and Fernandez [11]. This method uses short term and long term computer memories in order to find feasible solutions and to fix up the penalty weights. In this paper, a relaxed formulation of GAP which is called Relaxed GAP (RGAP) is considered. In this way, the capacity constraints are eliminated and a penalty parameter is added to objective function of the GAP model. Yagiura and Ibaraki [12] proposed a methodology by using the ejection chain algorithms and a neighbourhood construction method. Variable Depth Search (VDS), Tabu Search with Ejection Chains (TSEC) and Path Relinking with Ejection Chains (PREC) were compared on benchmark cases. Randall [13] studied the solution components and the local search heuristics from the literature. And also, two different probabilistic component selection heuristics were proposed with the adaptive and static schemes. As a result, performance of Ant Colony Optimization based methods gives better results against SA and TS. Lourenco and Serra [14] proposed a hybrid approach which combines a Greedy Randomized Adaptive Search Procedure (GRASP) and a Max-Min Ant System (MMAS). MMAS is a generation of the Ant Colony Optimization Algorithm to improve ant system. GRASP is a two-phase iterative randomized sampling method. Alfandari et al. [15] presented a Path-Relinking (PR) heuristics which is a kind of generalized scatter search for the GAP. This algorithm has two phases. The first phase contains LP and local search. In the second phase, paths are created between the feasible solution pairs picked from the first phase. It can be seen in the paper, TS might be very effective compared with PR. Yagiura et al. [16] proposed an algorithm which features ejection chains and a path relinking approach for the GAP. A neighbourhood construction is used to provide more complex and strong moves. And also, this algorithm has a mechanism for fitting parameters to keep the balance among feasible and infeasible regions. Haddai and Ouzia [17] presented an algorithm for generating and improving feasible assignments. This algorithm is applied at each iteration of a subgradient method for the weak Lagrangian relaxation of the GAP. Qu et al. [18] proposed an algorithm for multi-agent assignment problem where there is a need for a group of agents to select assignments from their eligible assignments. The objective is to find an assignment

profile that maximizes the global utility. Jacyna et al. [19] presented a mathematical model to solve task assignment problem of vehicles for a production company. They defined two stages for this problem. The first stage is to identify the tasks, the other is to determine the amount of vehicles required to fulfill these tasks. The algorithm was applied for real data. Demir and Canpolat [20] addressed due date assignment problem. In this study; genetic algorithms, evolutionary strategies and random search techniques are used and compared.

In addition, Cattrysse et al. [21] discussed some extensions of the generalized assignment problem. According to Mozzola [22], the GAP is a well known model for allocation, production planning and scheduling. In their paper, generalization of the GAP called the 0-1 generalized assignment problem with nonlinear capacity constraints (NLGAP) was presented. They aimed to consider capacity interactions among the tasks which are assigned to same employees. The multi-constraint generalized assignment problem (MCGAP) is a generalization of the GAP with multiple resources. LeBlanc et al. [23] proposed a methodology to solve the MCGAP with the consideration of the effects of setup times and costs to permit partitioning the inputs among the different machines. Genetic Algorithm (GA), Simulated Annealing (SA) and Lagrangian Relaxation (LR) are used to obtain results with systematic evaluations. The bottleneck GAP (BGAP) is defined by Mazzola et al. [24] and there are two types of this problem. First one is task based which minimizes the maximum cost of the assignments (TBGAP) and second one is employee based which minimizes the maximum of the total costs assigned to each employee (ABGAP). Martello and Toth [25] introduced approximation algorithms and an exact branch and bound approach to solve BGAP.

In the literature, many studies have been carried out on the service sector. Thomas and Terry [26] presented mixed-integer stochastic programming approach which has two stages for call centers. First stage compounds the staff scheduling and server sizing steps. And the second stage considers the uncertainty in arrival rates from period to period. According to them, the stochastic model generally gives a substantial reduction in the expected operation costs. Rodney and Ward [27] developed an algorithm for staffing and routing problems to minimize the overall workforce. They focused on the necessary agents with limited cross-training. Christian and Rainer [28] proposed a mixed-integer linear programming (MIP) model to minimize labor costs. They considered assigning multi-skilled employees to IT-projects. Krishnamoorthy et al. [29] presented a model to the Personnel Task Scheduling Problem (PTSP). They focused on minimising overall cost of employee with different skills required to perform the given set of tasks. Cordeau et al. [30] proposed an adaptive large neighborhood search heuristic and a construction heuristic for a telecommunication company to overcome technician

and task scheduling problem. Hojati and Patil [31] proposed an integer linear programming model and a heuristic to solve assignment and scheduling problem in service sector for part-time service employee with different availability and skills. The proposed model contains two steps. First step is determining shifts and second step is assigning the proper shifts to employees. Lin et al. [32] presented a problem-specific approach with three stages for crew rostering problem. Fuzzy sets are used to deal with job characteristics and the personal attributes. A linear goal programming model is proposed for effective assignment. Borenstein et al. [33] proposed a stochastic model to solve workforce scheduling problem for the British Telecom, in which technicians with different abilities are assigned to tasks which require different competences.

As can be seen from the literature review, several methods are presented in the literature to address to GAP and a large number studies have examined to deal with this problem.

### 3. Linear physical programming

As Messac et al. [34] stated, optimization problems can be classified into two categories: blind optimization and physical optimization. The decision maker does not really know the nature of the problem or the nature of the solution expected in blind optimization. In physical optimization, the decision maker has information and clearly defined objectives which can be expressed as physically meaningful terms related to the problem. Almost all operational research and engineering problems fall into the second category. In this chapter, linear physical programming (LPP) is described. Physical programming is a technique that requires the retrieval of physically meaningful information from the designer and produces a problem structure that is appropriate to the structure of the designer's preferences [35]. Within the Physical Programming procedure, DM explains their preferences using 4 different classes for each criterion (each criterion is described as belonging to one of 4 different classes). The lower value of the class function is better than the higher value. The ideal value of the class function is zero. Each class, depending on the sharpness of the choice, includes two states: hard and soft. All soft class functions will be a part of the integrated objective function (to be minimized). A class criterion is defined in one of 8 sub-classes: 4 soft (S), 4 hard (H). Physical programming avoids the limits of such a problem structure. In the flexible case, it characterizes the degree of desirability up to 11 intervals. The 6 intervals of the degree of desirability is defined in the 1S and 2S class criteria. 10 of the intervals is defined in class 3S, 11 of the intervals is defined in class 4S.

There are many studies using LPP in the literature. Onut et al. [36] presented a model to allocate the current energy resources to the Turkish manufacturing industry sub-groups by using LPP. Gulsun et al. [37] proposed a multi-objective model for aggregate production

planning and solved by using LPP. A production planning model developed by Maria et al. [38] and multi-objective model is solved by using LPP. Kucukbay and Araz [39] focused on the portfolio selection problem. In this study, fuzzy goal programming and linear physical programming are used and compared.

### 3.1. Linear physical programming problem model

This section will describe the procedure that will shape the problem of physical programming. Physical programming application procedure requires 4 short steps [35]:

1. For each criterion, class function type will be determined by decision maker (DM) among from 4 hard and 4 soft classes.
2. For each criterion, DM will determine the target values.
3. The LPP weight algorithm (LPPWA) is used to obtain weights with the DM inputs specified in the range limits.
4. Then problem is converted to LPP model.

### 4. Problem definition

The Central Operations Department (MOB), which aims to provide the best operational service with the highest efficiency, has a large number of transactions from the branches and directly from the customers during the day, such as loan application, guarantee letters preparation, money order based on written instructions, etc. After the transactions coming to the MOB, they are directed to the related departments. They try to complete the transactions with the limited employee resources in the departments. Transactions occur in multiple steps. As an example of the steps of the process, welcome (reading the customer order and specifying what they want), data entry, document control, approval steps can be provided. There are certain cut-off times for some operations. For example, the final closing time for EFT transactions should not exceed 17:30 as it is linked to the central bank system and the central bank system is being closed at 17:30. In addition, the SLA durations are calculated for the steps (steps of the operations) and the steps are intended to be completed within the calculated SLA period from the moment each step arrives at the MOB.

Depending on the workload intensity, other departments can support the related departments. Employees are empowered to perform certain operations according to their experience and training they have received, and the probabilities of making mistakes with the duration of operations can vary from employee to employee. The competencies of the employees gain importance at the point of giving support to other departments in a busy situation. The competencies of the employees are improved with the help of training programs organized by the bank.

In the current situation; in the MOB departments,

certain employees are selected to determine which transactions are related to their departments from the common pool, and they assign to transactions to the employees in their own departments. Prioritization of transactions and assignment of employees are based on the responsible employee's preferences and general rules could not be defined. Thus, while transactions with less priority levels can be completed, transactions that have already exceeded the SLA durations in other departments can be pretermited. In the departments where the transactions are directly selected by the employees, relatively easy and short transactions can be selected and priority of the transactions are not considered.

In this study, a two-step solution procedure is proposed to solve problems such as administrative difficulties, inefficient use of employees and inadequate management of priorities (Figure 1).

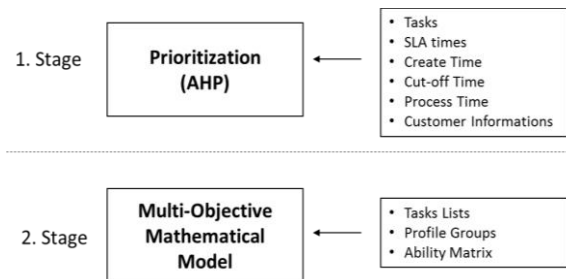


Figure 1. Problem solution procedure

Stage 1 is about the prioritization of tasks coming to the system. Tasks can be prioritized based on customer type, transaction type, urgency status and SLA times. The outputs of this stage are used in the second stage as inputs of mathematical model. Stage 2 is the assignment of tasks to profile groups created by considering specific experience and competencies. The second stage of the problem is the solution of the integer programming model by using the Linear Physical Programming (LPP). In the second stage, the transactions assigned to profile groups are directed to the relevant professionals. Each profile group has more than one employee, and each profile group is competent to conduct the transactions directed to them. Then first stage's results (prioritization stage) are used to push the tasks, assigned to the related profile groups, to the employees.

To summarize briefly, the following problems are observed in the current system:

- Improper assignments of tasks to the profile groups,
- Non-effective use of resources,
- Unbalanced workloads of profile groups,
- Tendency of employees on easy tasks during task selection phase
- Waiting transactions in the pool

In response to these problems; solutions have been

produced with two-step methodology. Since, by applying this approach, task prioritization phase becomes more standardized, their acceptance by the related employees and managers becomes easier. With the new system, the selection of tasks is entirely independent from the initiative of the employees. Hence, the tendency of employees on selecting easier tasks firstly is eliminated and the tasks with higher priority levels are assigned and finalized initially.

Also, the proposed methodology increases the communication level among profile groups. Employees have the flexibility of carrying out tasks of other profile groups when needed. It is possible for an employee to perform urgent and higher priority tasks of another profile group instead of performing a task with less priority score of his/her own profile group. In the current situation, employees wouldn't know the tasks in the queues of other profile groups and may remain idle when the other profile groups are overloaded. By applying the new methodology, capacity utilization balanced can be achieved among profile groups, and this would have a positive effect on the employees' moral. And finally, the proposed methodology would have a positive effect on customer satisfaction level.

In this study, below assumptions are made:

- Each incoming job consists of different tasks.
- Similar tasks are grouped into specific task pools.
- Different types of tasks in the same group require similar competences.
- Each employee in the same profile groups has similar competences.
- Each employee must be part of a profile group.
- Each employee can only be included in one profile group.
- Each task type may be carried out by different profile groups.
- All employees work hours are restricted to their shift start and end times.
- Each employee's completion time is different. However, it was assumed to be equal and average processing times were taken into account.
- Preparation and setup times for the works are neglected.
- Lunch times, break times etc. are neglected.

**4.1. Prioritization stage**

The prioritization stage constitutes the first step of the proposed method. During the working hours, many transactions are coming to the bank. These tasks have different importance levels. The order of importance in the current structure is determined by SLA time, cut-off time, customer information, and type of tasks. Generally, importance level of the tasks are determined by the employees. With our proposed method, the

prioritization structure is unbounded from the employees' initiative and a new structure is introduced. With the new structure, the priority score of each tasks is calculated. Employees will receive the most prioritized task among all tasks assigned to their group according to the calculated priority score. The priority score is calculated according to following rules:

1. Step: Calculation of Tolerance.
2. Step: Calculation of Significance Coefficient.
3. Step: Calculation of Final Score.

Firstly, the tolerance value is calculated. The tolerance value refers to the difference between the SLA time and average process time. The tasks can be kept up to tolerance value in the queue. However, only the tolerance value is not sufficient. Because; although the duration of the SLA is taken into account, there may be different types of tasks with the same duration and there may be different importance ratings among them. There may be a difference in the importance level between two similar transactions according to the customer type or the urgency of the transaction. Therefore; tolerance value is multiplied by the significance coefficient. Significance (Final) scores are calculated by using Analytic Hierarchy Process (AHP) method in second step. Then, final score value is calculated from second step's value for tasks with cut off time.

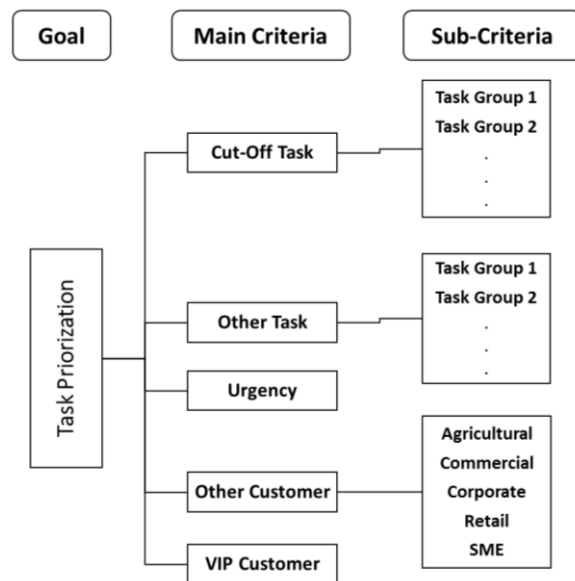


Figure 2. Structure of hierarchy

By using AHP technique, our aim is to standardize the tasks selection logic of employees. With this technique, the criteria are compared with each other. Thus, the importance of the criteria can be expressed numerically. The Analytic Hierarchy Process (AHP) is a multi-criteria decision-making technique and was developed by Saaty [40] to deal with complex decision problems. AHP Scores are calculated as follows and structure of hierarchy can be seen in Figure 2. A questionnaire is prepared to obtain the evaluations and

a 1 to 9 scale is used. According to Saaty's [40] pairwise comparison scale, 9 is extremely important and 1 is equally important. After the surveys were completed, inconsistency rates were performed and final scores were obtained.

And final score is calculated by using each criterion's score as shown in Figure 3.

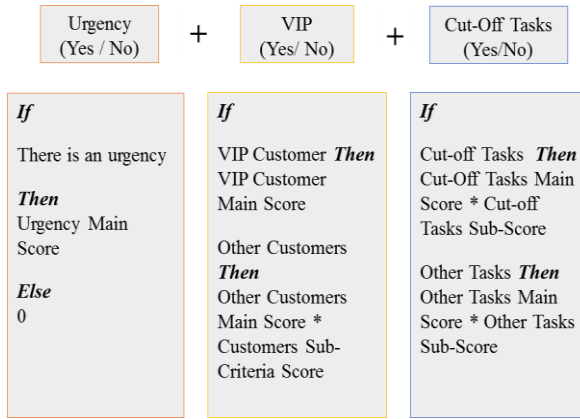


Figure 3. Final score calculation

4.2. Assignment stage

Parameters and decision variables of the model are listed as follows:

Index:

- $j$ : index for tasks  $j=1,2,3,\dots,J$
- $i$ : index for profile groups  $i=1,2,3,\dots,I$

Parameters:

- $\alpha_{ji}$ : Competence level for profile group  $i$  for task  $j$
- $a_{ji}$ : Ability matrix for profile group  $i$  for task  $j$
- $b_j$ : Importance level of task  $j$
- $tp_i$ : Available time for profile group  $i$
- $k_i$ : Available employee number for profile group  $i$
- $p$ : Planning period
- $t_j$ : Process time of task  $j$
- $c$ : Minimum capacity usage

Decision Variables:

$$x_{ji}: \begin{cases} 1, & \text{if task } j \text{ assigned to profile group } i \\ 0, & \text{otherwise} \end{cases}$$

Proposed task assignment model is given as follow:

Objective Function 1:

$$\text{Maximize: } \left( \sum_{j=1}^J \sum_{i=1}^I x_{ji} * \alpha_{ji} \right) \quad (1)$$

Eq. (1) tries to maximize the level of task assignments to appropriate profile groups.

Objective Function 2:

$$\text{Maximize: } \left( \sum_{j=1}^J \sum_{i=1}^I x_{ji} * b_j \right) \quad (2)$$

Eq. (2) tries to maximize assignment level of higher priority tasks.

Objective Function 3:

$$\text{Maximize: } (c) \quad (3)$$

Eq. (3) tries to maximize capacity usage of least occupied profile group. Capacity utilization rates are tried to be balanced.

Subject to:

$$x_{ji} \leq a_{ji} \quad \forall i, \forall j \quad (4)$$

$$\sum_{j=1}^J x_{ji} * t_j \leq tp_i \quad \forall i \quad (5)$$

$$\sum_{i=1}^I x_{ji} \leq 1 \quad \forall j \quad (6)$$

$$\sum_{i=1}^I k_i * p = tp_i \quad \forall i \quad (7)$$

$$\sum_{j=1}^J ((x_{ji} * t_j) / tp_i) \geq c \quad \forall i \quad (8)$$

$$x_{ji} \in \{0, 1\}, \quad \forall i, \forall j \quad (9)$$

Eq. (4) tries to ensure that a task can be assigned to a proper profile group. Eq. (5) tries to ensure that available time of profile groups cannot be exceeded. Eq. (6) tries to ensure that each task should be assigned to a profile group. Eq. (7) gives the relation between the total available time and the number of employees in profile groups. Eq. (8) determines the minimum capacity usage of profile groups. Eq. (9) determines the range of variables.

5. Experimental study

The model is developed by considering the problem of a private bank in Turkey and the related literature. One day banking data taken from this private bank is given in Figure 4, and some details are given below:

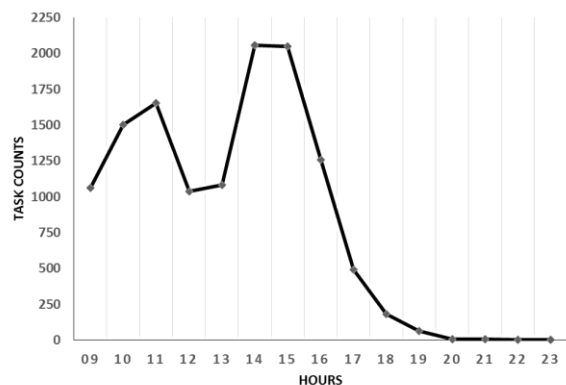


Figure 4. Hour based daily transactions

- Each new job progresses through a separate process. This study covers 6 different process types.
- Each incoming job consists of different tasks. For example; in terms of EFT; depending on the amount can consist of at least 2, up to 5 tasks. If the EFT amount is less than 1000 TL, only two task types are composed, while the amount is above 1000000 TL, five task types are formed.
- Similar tasks are grouped into specific task pools. The employee, assigned to this task pool, is able to do the different jobs such as EFT, remittance, etc. in this task pool.
- Each employee in the profile groups has similar competences.
- Each task type may be carried out by different profile groups.

**5.1. Linear physical programming application and weight determination**

First of all, optimal values are found for each objective function by solving the model by considering them one by one. Class intervals have been determined in the direction of optimum results.

The objective functions is classified as 2S (2nd soft

class). Our preferences and target values for the three goals are as shown in Table 1. Table 2 shows the final weight deviations of performance criteria. Steps of the LPPWA are given below [35] and mathematical relations for weight determination algorithm can be found in [35].

- Step 1 Start:  
 $\beta = 1,1; w_{p1}^+ = 0, w_{p1}^- = 0, \tilde{z}^2 =$   
*small positive number (e. g. 0.1)*  
 $p=0; s=1, n_{ek} = \# \text{ soft criteria}$
- Step 2  $p=p+1$
- Step 3  $p=s+1$   
 Evaluate sequentially;  
 $\tilde{z}^s, \tilde{t}_{ps}^+, \tilde{t}_{ps}^-, w_{ps}^+, w_{ps}^- ,$   
 $\tilde{w}_{ps}^+, \tilde{w}_{ps}^-, \tilde{w}_{min}$   
 If  $\tilde{w}_{min}$  is smaller than the selected small positive number (e.g., 0.01), increase  $\beta$  and go to step 2.
- Step 4 If  $s \neq 5$  then go to step 3.
- Step 5 If  $p \neq n_{sc}$  then go to step 2.

Then, the objective function (to be maximized) is constructed as a weighted sum of deviations ( $d_{ps}$ ) for all ranges and criteria.

**Table 1.** Management preferences concerned objectives (Target values).

| Preference degree  | g <sub>1</sub> | g <sub>2</sub> | g <sub>3</sub> |
|--------------------|----------------|----------------|----------------|
| Ideal              | >2870          | >3525          | > 0.487        |
| Desirable          | 2870 - 2670    | 3525 - 3300    | 0.487 – 0.467  |
| Tolerable          | 2670 - 2470    | 3300 - 3100    | 0.467 – 0.447  |
| Undesirable        | 2470 - 2270    | 3100 - 3000    | 0.447 – 0.427  |
| Highly Undesirable | 2270 - 2070    | 3000 - 2900    | 0.427 – 0.407  |
| Unacceptable       | <2070          | <2900          | < 0.407        |

**Table 2.** Normalized weight deviations of objectives.

|                |                    |                    |                    |                    |
|----------------|--------------------|--------------------|--------------------|--------------------|
|                | $\tilde{w}_{12}^-$ | $\tilde{w}_{13}^-$ | $\tilde{w}_{14}^-$ | $\tilde{w}_{15}^-$ |
| g <sub>1</sub> | 0.426086           | 0.043478           | 0.47826            | 0.052173           |
|                | $\tilde{w}_{22}^-$ | $\tilde{w}_{23}^-$ | $\tilde{w}_{24}^-$ | $\tilde{w}_{25}^-$ |
| g <sub>2</sub> | 0.371212           | 0.037878           | 0.5                | 0.090909           |
|                | $\tilde{w}_{32}^-$ | $\tilde{w}_{33}^-$ | $\tilde{w}_{34}^-$ | $\tilde{w}_{35}^-$ |
| g <sub>3</sub> | 0.4260869          | 0.0434782          | 0.47826            | 0.0521739          |

Our model in the LPP structure is given as follows:

- Piecewise Linear Archimedian Aggregate Function

$$\min j = \sum_{p=1}^3 \sum_{s=2}^5 (\tilde{w}_{ps}^- d_{ps}^- + \tilde{w}_{ps}^+ d_{ps}^+) \quad (10)$$

- Goal Constraints

$$g_1 = \left( \sum_{j=1}^J \sum_{i=1}^I x_{ji} * \alpha_{ji} \right) \quad (11)$$

$$g_2 = \left( \sum_{j=1}^J \sum_{i=1}^I x_{ji} * b_j \right) \quad (12)$$

$$g_3 = (c) \quad (13)$$

$$g_p + d_{ps}^- \leq t_{p(s-1)}^-; d_{ps}^- \geq 0; g_p \leq t_{p5}^- \quad (14)$$

(for all p classes 2S, p=1,2,..., n<sub>sc</sub>, s=2,...,5)

- System Constraints (Hard constraints)

$$x_{ji} \leq a_{ji} \quad \forall i, \forall j \quad (15)$$

$$\sum_{j=1}^J x_{ji} * t_j \leq tp_i \quad \forall i \quad (16)$$

$$\sum_{i=1}^I x_{ji} \leq 1 \quad \forall j \quad (17)$$

$$\sum_{i=1}^I k_i * p = tp_i \quad \forall i \quad (18)$$

$$\sum_{j=1}^J ((x_{ji} * t_j) / tp_i) \geq c \quad \forall i \quad (19)$$

$$x_{ji} \in \{0, 1\} \quad \forall i, \forall j \quad (20)$$

The model is solved by using GAMS 25.0.1 solver, and results are given in Table 3.

Table 3. Results

|              | First Goal | Second Goal | Third Goal         |
|--------------|------------|-------------|--------------------|
| Target Value | 2683       | 3525        | 0.41               |
| Preference   | Desirable  | Desirable   | Highly Undesirable |

Considering the numerical results, the first and second objective function are found in desirable range, and the third objective function is found in a highly undesirable range.

5.2. Results

In this study, GAMS 25.0.1 solver is used. The CPU time was 19 minutes and 32 seconds. Banking daily data is stored on Excel files and daily data was separated into hourly data. Therefore 12 separate datasets were obtained for each day. Then, datasets were tested on the proposed algorithm by using the program. In this study; only the busiest time zone (15:00 pm – 16:00 pm) data is used because of the continuous and large number of transactions coming to the banking system at that time interval. In this time zone, 11 different profile groups are available and each profile group has a different number of employees (see Table 4). According to data; it can be seen in Figure 5 that the capacity of the profile groups cannot be used in a balanced manner. While some profile groups use very small part of their capacities, some profile groups have had to work far beyond their capacity. When the Figure 5 is examined, it is seen that the capacity utilization rates of some profile groups are more than 100%. The reason of this situation can be explained as follows: The transactions performed by the profile groups are expressed in seconds and are based on standard transaction times. However, in order to complete the

transactions in busy profile groups, it is worked in periods well below the standard processing times. For example, in the first profile group, 47 transactions are completed. The standard processing time is 265 seconds. In this case; 47 transactions are completed in a total of 12,455 seconds. But, the capacity of this profile group is 7200 seconds. This conclusion is reached here. Employees in this profile group completed the transactions in an average of 153 seconds. They had to work faster than standart processing times to finish the assigned tasks.

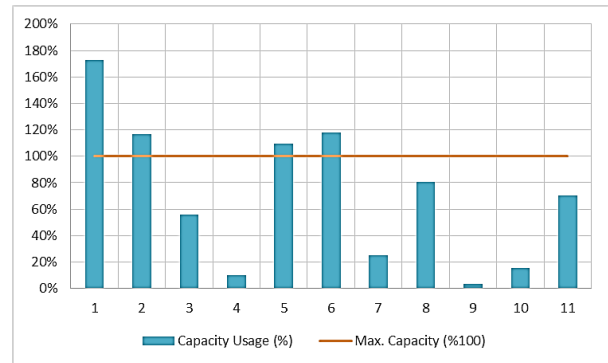


Figure 5. Capacity usage (%) in current situation

In the proposed solution, the capacity utilization level differences among the profile groups started to decrease and can be seen in Table 5. In the current situation, maximum capacity usage of any profile group could be 73% more than its own capacity as can be seen in Table 4. This unfair situation is tried to be balanced with the new methodology and maximum capacity utilization level among profile groups is not exceeded 100% with optimum solution (Table 5, Figure 6). As a result, the amount of unused idle capacity of the profile groups has decreased. As can be seen in Table 3, the result of the third objective function is highly undesirable because it may be due to the narrow range we have determined. We set the range between 0.407 and 0.487. Therefore; we wanted to observe the results by widening the ranges further. First, we set the lowest limit of the highly undesirable range to 0.30 as can be seen in Table 6. In this case; this value corresponds to the tolerable range from the ranges in Table 3 that we obtained earlier.

Table 4. Current situation

| Profile Groups | Number of Employees | Profile Groups Capacity in Seconds | Number of Tasks | Total Demand Time in Seconds | Capacity Usage (%) |
|----------------|---------------------|------------------------------------|-----------------|------------------------------|--------------------|
| 1              | 2                   | 7200                               | 47              | 12455                        | 173%               |
| 2              | 9                   | 32400                              | 75              | 37885                        | 117%               |
| 3              | 17                  | 61200                              | 114             | 34113                        | 56%                |
| 4              | 19                  | 68400                              | 36              | 6713                         | 10%                |
| 5              | 8                   | 28800                              | 232             | 31576                        | 110%               |
| 6              | 38                  | 136800                             | 1199            | 161537                       | 118%               |
| 7              | 36                  | 129600                             | 160             | 32855                        | 25%                |
| 8              | 30                  | 108000                             | 82              | 86973                        | 81%                |
| 9              | 10                  | 36000                              | 19              | 1283                         | 4%                 |
| 10             | 8                   | 28800                              | 26              | 4381                         | 15%                |
| 11             | 4                   | 14400                              | 59              | 10095                        | 70%                |



Table 5. Optimum solution

| Profile Groups | Number of Employees | Profile Groups Capacity in Seconds | Number of Tasks | Total Demand Time in Seconds | Capacity Usage (%) |
|----------------|---------------------|------------------------------------|-----------------|------------------------------|--------------------|
| 1              | 2                   | 7200                               | 38              | 7158                         | 99%                |
| 2              | 9                   | 32400                              | 62              | 32054                        | 99%                |
| 3              | 17                  | 61200                              | 82              | 25092                        | 41%                |
| 4              | 19                  | 68400                              | 245             | 28555                        | 42%                |
| 5              | 8                   | 28800                              | 263             | 27119                        | 94%                |
| 6              | 38                  | 136800                             | 890             | 136778                       | 100%               |
| 7              | 36                  | 129600                             | 335             | 62090                        | 48%                |
| 8              | 30                  | 108000                             | 19              | 68400                        | 63%                |
| 9              | 10                  | 36000                              | 55              | 14774                        | 41%                |
| 10             | 8                   | 28800                              | 39              | 11934                        | 41%                |
| 11             | 4                   | 14400                              | 21              | 5912                         | 41%                |

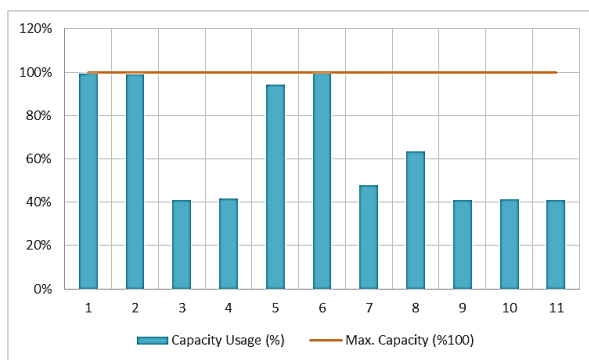


Figure 6. Capacity usage (%) in the proposed solution

When we solve the model for this range, the result for this objective function is found as 0.30, which means that it is highly undesirable. As can be seen in Table 6, we set the lowest limit of the highly undesirable range to 0.25 to extend the range a little further. When we solve the model again for this range, the result for this objective function is found as 0.25 which is highly undesirable.

Table 6. New Preference Values for the Third Goal

|                           | Target values | Target values | Target values |
|---------------------------|---------------|---------------|---------------|
| <b>Ideal</b>              | > 0.487       | > 0.487       | > 0.487       |
| <b>Desirable</b>          | 0.487 – 0.45  | 0.487 – 0.44  | 0.487 – 0.48  |
| <b>Tolerable</b>          | 0.45 – 0.40   | 0.44 – 0.38   | 0.48 – 0.47   |
| <b>Undesirable</b>        | 0.40 – 0.35   | 0.38 – 0.32   | 0.47 – 0.46   |
| <b>Highly Undesirable</b> | 0.35 – 0.30   | 0.32 – 0.25   | 0.46 – 0.45   |
| <b>Unacceptable</b>       | < 0.30        | < 0.25        | < 0.45        |

It is seen that the minimum capacity utilization rates of the profile groups decrease significantly as the range values for the third objective function increase. It ignores the balance among profile groups. Therefore, we can observe that when we narrow the ranges, the minimum capacity utilization rate becomes higher. When we set the minimum target value to 0.45 for testing, the range is still fairly undesirable but a more balanced assignment takes place. The results can be

seen in Table 7.

Once the relevant tasks have been assigned to the relevant profile groups by proposed solution, the priority scores calculated in the first stage is taken into account. The tasks assigned to each profile group are sorted by ascending order according to the calculated priority score and these tasks are done by the appropriate employees defined in the profile groups, respectively.

Table 7. Capacity Usage (%)

| Profile Groups | Minimum Target Value 0.30 | Minimum Target Value 0.25 | Minimum Target Value 0.45 |
|----------------|---------------------------|---------------------------|---------------------------|
| 1              | 99%                       | 99%                       | 93%                       |
| 2              | 99%                       | 99%                       | 94%                       |
| 3              | 30%                       | 25%                       | 45%                       |
| 4              | 91%                       | 100%                      | 45%                       |
| 5              | 100%                      | 100%                      | 94%                       |
| 6              | 100%                      | 100%                      | 98%                       |
| 7              | 30%                       | 25%                       | 45%                       |
| 8              | 66%                       | 73%                       | 63%                       |
| 9              | 30%                       | 25%                       | 45%                       |
| 10             | 31%                       | 26%                       | 45%                       |
| 11             | 30%                       | 25%                       | 45%                       |

## 6. Conclusion

This study investigates personnel task assignment problem in central operational departments for banking sector. Although many methods have been proposed to address personnel task assignment problem, there is no direct solution for this specific problem in the banking sector. Therefore, a two-step methodology has been proposed to solve this real life problem with the consideration of task priorities, task-profile group compability, capacity utilization balance of profile groups.

The proposed method consists of two stages. The first stage is about the prioritization of tasks. At this stage; customer types, transaction types, urgency status, task create times, processing times, cut-off times and SLA

times are taken into account, task priorities are found and the outputs of this stage are used in the second stage as inputs. The second stage is the part where tasks are assigned to profile groups or employees by considering competence, experience and other capabilities of employees. A multi-objective mathematical model is developed for this stage and the linear physical programming technique is used to solve this model.

In our study, real banking data is used and according to results, capacity usage levels of profile groups becomes more balanced and minimum capacity usage among them is increased to at least 41%. As a result, it has been observed that tasks are prioritized in a more precise way and more accurate and balanced task-employee assignments are obtained. There are no unassigned tasks when attempting to make a balanced assignment. When we evaluate the results of objective functions separately, the first and second objective function are found in desirable range, and the third objective function is found in a highly undesirable range.

In this study, completion times for different employees are assumed to be same. In a future study, variations in completion times can be taken into account. Also, number of tasks coming to task list can be forecasted and this can be added to proposed model as a new input. And then, we offered that tasks are assigned to the any available employees according to priority scores. In the future studies, scheduling algorithms can be used to create the tasks lists of the employees at this stage. Our proposed model gives optimal solution for small problem sets. However, it would be difficult to reach the optimum solution if the problem size increases. For larger size problem sets, heuristics /metaheuristics methods can be used.

### Acknowledgments

The authors would like to thank the anonymous reviewers for their helpful and valuable comments.

### References

- [1] Felzl, H., & Raidl, G. R. (2004). An improved hybrid genetic algorithm for the generalized assignment problem, Nicosia, Cyprus: *Proceedings of the 2004 ACM Symposium on Applied Computing*.
- [2] Kuhn, H. W. (1955). The Hungarian method for the assignment problem, *Naval Research Logistics*, 2, 83-97.
- [3] Balachandran, K. R. (1972). Purchasing priorities in queues, *Management Science*, 18, 319-326.
- [4] Mazzola, J. B., Neebe, A. W., & Dunn, C. V. R. (1989). Production planning of a flexible manufacturing system in a material requirements planning environment, *International Journal of Flexible Manufacturing Systems*, 1, 115-142.
- [5] Ross, G. T., & Soland, R. M. (1977). Modeling facility location problems as generalized assignment problems, *Management Science*, 24, 345-357.
- [6] Pentico, D. (2007). Assignment problems: a golden anniversary survey. *European Journal of Operational Research, Open Journal of Discrete Mathematics*, 176, 774-793.
- [7] Osman, I.H. (1995). Heuristics for the generalized assignment problem: simulated annealing and tabu search approaches, *OR Spektrum*, 17, 211-225.
- [8] Chu, P.C., & Beasley, J.E. (1997). A genetic algorithm for the generalized assignment problem, *Computers Operations Research*, 24, 17-23.
- [9] Racer, M., & Amini, M.M. (1994). A robust heuristic for the generalized assignment problem, *Annals of Operations Research*, 50, 487-503.
- [10] Laguna, M., Kelly, J.P., Gonzalez-Velarde, J.L., & Glover, F. (1995). Tabu search for the multilevel generalized assignment problem, *European Journal of Operational Research*, 82, 176-189.
- [11] Diaz, J. A., & Fernandez, E. (2001). A tabu search heuristics for the generalized assignment problem, *European Journal of Operational Research*, 132, 22-38.
- [12] Yagiura, M., Ibaraki, T., & Glover, F. (2004). An ejection chain approach for the generalized assignment problem, *INFORMS Journal on Computing*, pp. 133-151.
- [13] Randall, M. (2004). Heuristics for ant colony optimisation using the generalised assignment problem, in: *Proceedings of IEEE Congress on Evolutionary Computation*, Portland, Oregon, USA, pp. 1916-1923.
- [14] Lourenço, H.R., & Serra D. (2000). Adaptive search heuristics for the generalized assignment problem, *Mathware & Soft Computing*, 7, 1-15.
- [15] Alfandari, L., Plateau, A., & Tolla, P. (2001). A two-phase path relinking algorithm for the generalized assignment problem, *4th International Conference of Metaheuristics*, Porto, Portugal, 175-179.
- [16] Yagiura, M., Ibaraki, T., & Glover, F. (2006). A path relinking approach with ejection chains for the generalized assignment problem, *European Journal of Operational Research*, 169, 548-569.
- [17] Haddadi, S., & Ouzia, H. (2001). An effective lagrangian heuristic for the generalized assignment problem, *INFOR: Information Systems and Operational Research*, 39: 351-356.
- [18] Qu, G., Brown, D., & Li, N. (2019). Distributed greedy algorithm for multi-agent task assignment problem with submodular utility functions, *Automatica*, 105, 206-215.
- [19] Jacyna, M., Izdebski, M., Szczepański, E., & Golda, P. (2018). The task assignment of vehicles for a production company, *Symmetry*, 10, 551.
- [20] Demir, H. I., & Canpolat, O. (2018) Integrated process planning, WSPT scheduling and WSLK due-date assignment using genetic algorithms and evolutionary strategies, *An International Journal of*

- Optimization and Control: Theories & Applications, Vol 8, No 1, pp. 73-83.*
- [21] Cattrysse, D.G., & Van Wassenhove, L.N. (1992). A survey of algorithms for the generalized assignment problem, *European Journal of Operational Research*, 60, 260–272.
- [22] Mozzola, J. B. (1989). Generalized assignment with nonlinear capacity interaction, *Management Science*, 35.
- [23] LeBlanc, L. J., Shtub, A., & Anandalingam, G. (1999). Formulating and solving production planning problems, *European Journal of Operational Research*, 112, 54–80.
- [24] Mazzola, J.B., & Neebe, A.W. (1988). Bottleneck generalized assignment problem, *Engineering Cost and Production Economics*, 14, 61-65.
- [25] Martello, S., & Toth, P. (1995). The bottleneck generalized assignment problem, *European Journal of Operational Research*, 83 (3), 621–638.
- [26] Thomas, R. R., & Terry, P. H. (2010). A stochastic programming model for scheduling call centers with global service level agreements, *European Journal of Operational Research*, 207, 1608-1619.
- [27] Rodney, B. W., & Ward, W. (2005). A staffing algorithm for call centers with skill-based routing, *Manufacturing & Service Operations Management*, 7, 276-294.
- [28] Christian, H., & Rainer, K. (2010). Scheduling and staffing multiple projects with a multi-skilled workforce, *OR Spectrum*, 32:343–368.
- [29] Krishnamoorthy, M., Ernsts, A. T., & Baatar, D. (2012). Algorithms for large scale shift minimisation personnel task scheduling problems, *European Journal of Operational Research*, 219, 34-48.
- [30] Cordeau, J. F., Laporte, G., Pasin, F., & Ropke, S. (2010). Scheduling technicians and tasks in a telecommunications company, *J Sched*, 13: 393–409.
- [31] Hojati, M., & Patil, A.S. (2011). An integer linear programming-based heuristic for scheduling heterogeneous, part-time service employees, *European Journal of Operational Research*, 209, 37–50.
- [32] Lin, H. T., Chen, Y. T., Chou, T. Y., & Liao, Y. C. (2012). Crew rostering with multiple goals: an empirical study, *Computers & Industrial Engineering*, 63, 483–493.
- [33] Borenstein, Y., Shah, N., Tsang, E., Dorne, R., Alsheddy, A., & Voudouris, C. (2010). On the partitioning of dynamic workforce scheduling problems, *J Sched*, 13, 411–425.
- [34] Messac, A., Gupta, S.M., & Akbulut, B. (1996). Linear physical programming: a new approach to multiple objective, *Optimization Transaction on Operation Research*, 8:39-59.
- [35] Messac, A., Melachrinoudis, E., & Sukam, C. P. (2001). Mathematical and pragmatic perspectives of physical programming, *AIAA Journal*, 39(5):885-893.
- [36] Onut S., Tuzkaya U.R., Tuzkaya G., & Gulsun B. (2011). A multi-objective energy resource allocation model for turkish manufacturing industry using linear physical programming, *International Journal of Innovative Computing Information and Control*, vol.7, pp.3147-3169.
- [37] Gulsun B., Tuzkaya G., Tuzkaya U.R., & Onut S. (2009). An aggregate production planning strategy selection methodology based on linear physical programming, *International Journal of Industrial Engineering-Theory Applications and Practice*, vol.16, pp.135-146.
- [38] Maria, A., Mattson, C., Yahaya, A., & Messac, A. (2003). Linear physical programming for production planning optimization, *Journal Engineering Optimization*, 35, pp. 19-37.
- [39] Kucukbay, F., & Araz, C. (2016). Portfolio selection problem: a comparison of fuzzy goal programming and linear physical programming, *An International Journal of Optimization and Control: Theories & Applications, Vol 6, No 2, pp. 115-120.*
- [40] Saaty, T.L. (1977). A scaling method for priorities in hierarchical structures, *Journal of Mathematical Psychology*, 15, 57-68.

**Kenan Cetin** received the B.Sc. degree in 2010 from the Department of Industrial Engineering from Sakarya University, Turkey. He is working as an R&D Specialist in Vakıfbank. His research interests are optimization algorithms, scheduling, artificial intelligence and data mining.


 <http://orcid.org/0000-0003-4894-8215>

**Gülfem Tuzkaya, Ph.D.**, is currently working as an Associate Professor in the Department of Industrial Engineering, Marmara University. Dr. G. Tuzkaya received her MSc. Degree in Industrial Engineering from Yildiz Technical University, in 2002, MBA degree from Istanbul Technical University, in 2005, and Ph.D. degree in Industrial Engineering from Yildiz Technical University, in 2008. She has been in Tilburg University, Netherland, as a visiting researcher in 2010. Her research interests are in the areas of reverse-forward logistics systems design, facility design and operations research applications for operations management problems.

 <http://orcid.org/0000-0001-7683-4405>

**Ozalp Vayvay** received the M.Sc. and Ph.D. degrees in 1993 and 2000 from the Department of Industrial Engineering from Marmara University, Turkey respectively. He is working as a Professor at Business Administration Department of Marmara University. His

*research interest includes supply chain management, quality management, and innovation management.*

 <http://orcid.org/0000-0003-0504-3395>

An International Journal of Optimization and Control: Theories & Applications (<http://ijocta.balikesir.edu.tr>)



This work is licensed under a Creative Commons Attribution 4.0 International License. The authors retain ownership of the copyright for their article, but they allow anyone to download, reuse, reprint, modify, distribute, and/or copy articles in IJOCTA, so long as the original authors and source are credited. To see the complete license contents, please visit <http://creativecommons.org/licenses/by/4.0/>.