TWMS J. Pure Appl. Math., V.14, N.1, 2023, pp.69-90
DOI:10.30546/2219-1259.14.1.2023.69

A DYNAMIC PACKING APPROACH FOR INTERNET-OF-THINGS AND LOGISTICS APPLICATIONS

UĞUR ELİİYİ[1] AND EFENDİ NASİBOV[2]

Abstract. A novel dynamic rectangular packing model for a resource allocation problem, which has many applications in Logistics Internet-of-Things (L-IoT) is considered in this study. We use an optimization approach to deal with an IoT-based problem, whose objective is to maximize the profit obtained from packing the data demand over a sequence of time frames, while satisfying several quality of service constraints. We propose a nonlinear integer programming model for the problem that optimizes demand partitioning and rectangle packing simultaneously, for the first time in literature. By introducing effective upper and lower bounds for this practically important problem, a computational experimentation is designed for assessing the model and bound performances. An extensive discussion and recommendations for policy-making are included based on the computational results.

Keywords: bin packing, optimization, logistics Internet-of-Things, nonlinear integer programming.

AMS Subject Classification: 90C11, 52C15, 90C30, 90B22, 90C90, 68M20.

1. Introduction

Modern technology has gotten to be one of the significant parts of most peoples lives around the world. In particular, the Internet and the Internet of Things (IoT) have become the most indispensable parts of our lives in this era, since the number of physical objects connected to the Internet increased immensely as coverage increased [4]. The total number of objects in the IoT network had surpassed 12.5 billion by 2011 [5], and is expected to reach to an incredible amount of 50 billion by the end of 2050 thanks to advances such as 5G-Internet [41]. Advances in IoT technology such as Machine-to-Machine (M2M) communication and Radio Frequency IDentification (RFID) rendered possible a variety of applications in many business sectors like transportation, healthcare and agriculture, as well as implementations in smart grids, smart homes and cities, energy environment and libraries [18].

As IoT brought a major change to the field of information technology, the concept became especially important for today's modern logistics. A literature survey study on IoT and its applications was conducted by Atzori et al. in 2010 [6]. Since then, as the field is highly dynamic and attracts much attention from both academicians and practitioners, the literature has grown considerably. As a result, another very recent survey on IoT in the industrial marketplace has been performed by Salih et al. [50]. The authors provided a comprehensive review on benefits and drawbacks of IoT in addition to its variety of applications. They also identified the role of IoT in telemedicine and healthcare, as well as the benefits of IoT in the COVID-19 pandemic.

[1]Izmir Bakircay University, Department of Business, Izmir, Turkey
[2]Dokuz Eylul University, Department of Computer Science, Izmir, Turkey
e-mail: ugur.eliiyi@bakircay.edu.tr, efendi.nasibov@deu.edu.tr
Manuscript received February 2021.

Another nice review in the specific area of energy sector applications of IoT can be found in [40], which aims to provide energy policy-makers an overview of the impact of IoT technologies in the optimization of energy systems.

One of the most important applications of the Internet of Things is in the field of logistics. As a result, Logistics Internet-of-Things (L-IoT) emerged as a relatively new concept, which can be defined as IoT used for logistics applications. During the last decade, practitioners and academics have paid close attention to L-IoT as a direct result of its practical importance. An excellent survey focusing on this emerging important concept is provided by Golpra et al. [18]. By systematically analyzing the most recent 560 papers in the literature, the authors conducted a comprehensive review to draw a framework of the past and shed light on potential directions for future research. The reader is referred to this survey study for a deeper understanding of L-IoT concepts and categorization of the existing work. While identifying the existing research gaps, the results of the survey clearly suggested that focusing on mathematical approaches for optimally modeling and designing IoT-based systems is rather crucial. Despite this critical importance, the research into the integration of mathematical modeling and optimization approaches in L-IoT is still in its early stages. Therefore, mathematical models and optimization for supporting L-IoT implementations is stated to be a significant research direction.

As an example to rare studies employing mathematical models and optimization in the L-IoT literature, Tu et al. [57] focused on a case study on a cyber-physical system (CPS) application in production logistics and supply chain application domain, especially on the integration of RFID and CPS. The authors proposed a quantitative evaluation method for cost-benefit analysis of adopting IoT-based CPS. They also used an IoT-based CPS test-bed platform for performance evaluation of the developed method. In another study, the same authors utilized a design research approach using process analysis and Petri net modeling, in order to propose a unified IoT framework to model distributed IoT processes, devices and objects [56]. For integrating a vendor-managed inventory strategy into the general multi-project multi-resource multi-supplier construction supply chain (CSC) network design and facility location, Golpra used a Mixed Integer Linear Programming approach [16]. The developed framework allowed an efficient dynamic scheduling of IoT resources.

Within a smart city context, the integration of IoT with data access networks, Geographic Information Systems (GIS) and network optimization was considered in a study for smart waste collection [19]. The study involved the use of an IoT prototype embedded with sensors, which can read, collect, and transmit trash volume data over the Internet. The data generated by the sensors were collected and utilized for obtaining optimal waste collection strategies. A realistic scenario analysis was performed for illustrating the efficiency of the obtained solutions. Other studies tried to optimize energy consumption by using cloud-based optimization algorithms [64, 17].

In the M2M wireless communication in IoT, the Medium Access Control (MAC) scheduler is responsible for two tasks concerning resource allocation. The first task is to determine the IoT devices that will be served in a specific time frame, thus forming a service queue for data transmission while also determining the amount of data required for each such device. Quality of Service (QoS) considerations in wireless communication in IoT, such as minimum throughput requirements regarding data transmission rates and delay constraints are much more demanding than in wired networks due to the highly dynamic nature. Therefore, this first task depends mainly on the QoS parameters used for the network and is studied under names of burst construction [43] and packet scheduling [62].

The second task of the MAC scheduler is to assign time and frequency intervals to each communicating IoT device, which is referred to as frame packing, packet mapping or burst mapping in the literature [7, 8, 42, 43, 51, 52]. Frame packing corresponds to a rectangular packing problem with additional constraints. This problem is a special case of the two-dimensional bin packing problem (2D-BPP), which is generally defined as packing of a given set of various 2D items into the minimum number of two-dimensional rectangular bins without any overlapping [31]. It is generally assumed that the item sizes are fixed, and the items are packed orthogonally with each edge parallel to the corresponding bin edge. When the items are rectangular, the problem is called a rectangular packing problem. User terminals in wireless telecommunication or the IoT devices in M2M communication are mapped to rectangular areas (referred to as bursts) within the frame. The MAC scheduler is required to pack these rectangles in the frame as efficiently as possible under technology and traffic-specific constraints as well as QoS considerations. Hence, this second task of scheduling is a significant research area for all equipment makers and network service providers.

So-In et al. [51] provided an extensive review for main issues considered in designing the scheduling mechanisms at the MAC scheduler. The authors listed various criteria and scheduler design factors with respect to different QoS considerations, which shape the algorithms and heuristics proposed in literature. Throughput and power consumption are counted among major criteria to be considered while safeguarding system scalability. QoS constraints such as maximum transfer delay and minimum transfer rates affect the packing of variable-sized rectangles over a sequence of time frames.

The 2D-BPP is NP-hard as it is a generalization of the one-dimensional bin packing problem (1D-BPP). In the 1D-BPP, a set of positive values has to be partitioned into the minimum number of subsets, such that the sum of values in each subset does not exceed the predetermined bin capacity [38]. The 1D-BPP is proven to be NP-hard in the strong sense [15, 29]. Many applications of the different types of bin packing problems in supply chains have been identified, such as container, pallet, vehicle or cargo loading, cutting stock, packaging design, trim loss, load balancing, scheduling, project management and financial budgeting [13]. The 2D-BPP is closely related to other problems and finds many industrial applications such as packaging, loading, mechanical and electronic designs, cutting of materials in production (the cutting stock problem), multiprocessor scheduling, task assignment, and machine scheduling.

The optimization model introduced in this study aims at profit maximization in a rectangular data packing environment of the MAC scheduler, which has its applications in L-IoT as well as other areas. This important resource allocation and scheduling problem can be directly applied to the frame packing problem at the MAC-layer of dedicated IoT gateways. A recent study has shown that, when traffic patterns of IoT devices can be predicted, proactive solutions to the massive access problem become viable [49]. Our study brings such a proactive packing approach to the massive access problem through a novel optimization model, which handles demand partitioning and rectangle packing simultaneously. The model also employs a scheduling perspective for handling the time dimension of the problem. This unique feature allows for a more efficient solution of the difficult optimization problem on hand, and brings a contribution to literature. Eliiyi and Nasibov [14] visited dynamic modeling within a wireless telecommunications context by briefly defining the problem environment and possible constraints. However, their packing model included no explicit objective function, and no computational results were provided. The time dimension included in the developed modeling approach defines a dynamic environment, corresponding to repetitive packing/loading, such as container loading or pallet loading in logistics. Hence, the model proposed in this study is also valid for other environments where

rectangular packing is of concern. Effective container and pallet loading approaches can help to reduce logistics and transportation costs considerably. The problems are computationally hard, and receives continuous attention from researchers as there is room for improvement for efficient solutions. Although this logistics problem involves the packing of 3-D objects into a 3-D container, the problem is often studied via its 2-D reduction [46, 55].

In light of the above discussion, our study aims to fill the gap in literature by considering an optimization problem that has a wide range of application areas in L-IoT, from wireless M2M communication in IoT within the Industry 4.0 context, to container loading, pallet loading, and many storage operations. In summary, our study aims to contribute to the literature by;

- Introducing a novel and flexible mathematical model that can be adapted for a variety optimization problems existing in the field of L-IoT,
- Handling demand partitioning and packing optimization in an integrated manner,
- Providing an efficient formulation to a three dimensional packing problem via reducing it to a rectangular packing problem and utilizing a scheduling approach for handling the third dimension,
- Introducing lower and upper bounding procedures for the problem,
- Providing optimal and efficient solutions to this important problem for the first time in literature.

The outline for the remainder of the paper is as follows. We provide a review of the related literature in the next section along with our problem definition. In Section 3, the notations used in the formulation are presented as well as the nonlinear integer programming model. The developed upper and lower bounding procedures for the defined problem are introduced in Section 4. The results of our computational experiment are reported and discussed next. Finally, the conclusions and future research directions are summarized in the last section.

2. Problem definition and related literature

The optimization problem considered in this study aims at profit maximization in a 2D packing environment in L-IoT. Namely, we extend the frame packing problem in literature to include time-related service level constraints. A popular and critical application of the problem occurs in the massive access and data scheduling at the MAC-layer of IoT gateways, as mentioned in the previous section. From this perspective, the L-IoT systems architecture proposed in [18] applies to the problem in this study. On the other hand, the defined optimization problem directly affects the performance of L-IoT systems, especially through the core cloud computing and the upper three (service-oriented interface, application and business) layers. The problem is also applicable to container loading and pallet loading, and to other general packing environments. Due to this broad range of application areas, we define our problem in a general frame packing perspective, as opposed to specifying a single problem environment. As our focus is on a resource allocation problem, which is defined as an extended packing problem on a 2D structure, we first include the related literature in this section and then define our problem in comparison to the existing studies.

Lodi et al. [31, 33] reviewed solution approaches for general 2D-BPP. The authors listed the heuristics and exact algorithms in literature for bin packing, where the objective is to pack all items into the minimum number of bins. The authors also reviewed solution approaches for the Strip Packing Problem, which is defined as a special 2D-BPP, in which all items are rectangles and the frame to be packed is a strip with a fixed width and infinite height. The objective in this case becomes strip height minimization. In another noteworthy study, Martello and Vigo

[37] proposed lower bounds that are used in an exact branch-and-bound optimization algorithm for the 2D-BPP. A good survey on strip packing problems has been published more recently by Oliveira et al. [44] .

Kenyon and Rmila [27] presented an approximation scheme for the 2D-strip packing problem based on a linear programming relaxation. Caprara and Monaci [10] dealt with the 2D-knapsack problem aimed at packing a maximum-profit subset of available rectangles. They proposed four exact algorithms based on 1D-relaxation, and proved the worst-case performance of the associated upper bound. A fast and recursive heuristic for the 2D-strip packing problem was proposed in [66]. Wei et al. [60] also tackled the 2D-strip packing problem via transforming it into a series of 2D rectangular packing problems. For an extensive literature review of packing problems, the reader can refer to [59].

There is considerable work on 2D-BPP within the wireless data transmission context, since the problem gained increasing importance with growing IoT network. Most studies in literature mainly focused on the packing of the downlink (data receiving direction) transmissions. One of the very few studies on uplink (data sending direction) packing was performed in [62]. The authors presented a packet scheduling algorithm to provide QoS support for different data traffic classes. The authors also investigated factors that affected network performance through computational simulation. A more recent study including uplink transmission focused on the minimization of power consumption for green communications in the design of 5G and subsequent wireless communication systems [48]. Namely, the authors considered the optimization of various unmanned aerial vehicle (UAV) system parameters (e.g. vehicle position, height, beamwidth), while tackling the resource allocation for uplink communications between ground IoT devices and a UAV sending data through a series of mathematical models.

Most studies on the downlink frame allocation aimed to maximize the packing performance in a single frame. In these studies, the problem includes two-stages corresponding to two periodical scheduling decisions. The first stage is involved with determining the sizes and shapes of the rectangles to be packed for each users data for transmission, whereas the second stage includes decisions on how to pack the corresponding rectangles on the frame. In [11], Cohen and Katzir named these stages as macro and micro scheduling problems and analyzed their computational complexities via developing two inter-dependent approximation algorithms. They proved that macro scheduling can be solved in pseudo-polynomial time through transformation into the Multiple-Choice Knapsack Problem. The authors simplified the second stage by reducing the number of rectangles to be packed. Most of these studies averaged and simplified the QoS constraints of service users such as demand levels, subscription priorities, minimum transmission rate or maximum allowable delay [51].

A genetic algorithm for the downlink frame packing problem was proposed in [42]. Simulating the network traffic with three different QoS classes, the authors applied the Next-Fit-Decreasing-Height strip packing approach to maximize capacity utilization of the packed frame. They based their genome modeling in the genetic algorithm on variable width and height dimensions that could change according to the defined objective of the problem. For example, the minimum power consumption objective for downlink frame packing forced the allocated rectangles to smaller widths, hence smaller processing times on mobile stations in [52]. Other researchers also studied packing methods in telecommunications [8, 43].

Many studies in literature tried to simplify the downlink frame packing problem for obtaining better solutions. In [53], So-In et al. enhanced their algorithm by fixing the width or height of the data bursts corresponding to rectangles in order to reduce complexity. On the other hand, Israeli et al. approached the problem in a rather unique way in [21], naming it as a sequential

rectangle placement problem. The authors aimed to pack a queue of data packets using strict hierarchical priorities: For one users (or devices) rectangle to be packed, all higher-priority packets were to be packed before it. They showed the problem to be NP-hard. Wang et al. also proposed a low-complexity algorithm for the problem in [58] by adapting a quasi-human-based heuristic previously developed in [63]. Another contribution to the problem within the communications context was brought in [32], in which the authors introduced two highly efficient heuristics for downlink subframe allocation. They defined two different allocation models and proved NP-hardness for both models. The optimization algorithm developed for the defined Adjacent Permutation Zone problem employed a heuristic from the knapsack literature [36]. Extensive analysis and comparison of the algorithms with relevant work from literature [52, 58] proved superior performance of their algorithms.

A fairness-based transaction packing algorithm for blockchain-empowered industrial IoT systems was proposed in [23]. The fairness problem was reduced to the subset sum problem and a heuristic approach was proposed as well as a min-heap-based exact algorithm. Omer et al. proposed a mixed integer programming scheduling formulation for satisfying the maximum number of requests with deadline constraints, which adopted three-tier fog computing architecture [45]. In another study, a decoupled learning strategy was used to tackle scheduling in the massive access problem [24]. Joint forecasting and scheduling of incoming requests in a massive access environment was also studied in [49] at the MAC-layer. A quantum computing-based optimization technique was introduced in [9] to maximize data accuracy in IoT, and the performance of the developed approach was tested in terms of data cost, data accuracy and data temporal efficiency.

Apart from the mentioned studies, the literature on IoT contains rare studies involving mathematical models and optimization, most of which are mainly on energy minimization. As our problem also tries to use resources in the most efficient manner, some recent examples of these studies are worth mentioning. An interesting paper focused on terminal equipment (TE) allocation and energy consumption optimization in IoT in a heterogeneous communication system [20]. In their bilateral random system model, the authors used convex optimization for allocation of the TEs with different QoS demands and aimed at minimum energy expense cost of the whole IoT system. The allocation was modeled via a Mixed-Integer Nonlinear Programming (MINP) model. For energy minimization in smart homes, Kazmi et al. [25] used a multiple knapsack type mathematical formulation for scheduling of the home appliances. The study also included simulation of the proposed approach for validating the performance of the proposed scheduling algorithms. In another recent study, a hybrid metaheuristic of whale optimization and simulated annealing was utilized for minimizing the energy consumption of sensors in an IoT network [22]. Different control rules and delay-dependent inequalities were used for improving the robustness and design parameter selection of a class of neural networks, which can lead to increased speed and accuracy for deep learning on state-of-the-art systems of IoT devices [28]. Other metaheuristic approaches for energy consumption minimization can be found in [34], and a literature review on such approaches is available in [65].

For network sustainability in a wireless sensor network (WSN), multiple objectives such as coverage, connectivity and energy consumption should be considered to improve QoS. Considering this multi-objective structure, the coverage rate, connectivity and energy consumption was handled simultaneously through an adaptive coverage and connectivity scheme in [47].

The rectangular packing problem has its applications also in the logistics context. For example, a single batch processing machine scheduling problem with 2D packing constraints was studied within a manufacturing context in [30]. The authors tried to minimize the makespan

of the jobs on a single batch processing machine, which corresponded to a generalization of the 2D-BPP. On the other hand, Wei et al. [61] jointly considered routing and loading for the well-known capacitated vehicle routing problem with 2D loading constraints. The authors proposed a simulated annealing algorithm to solve four variants of the problem.

Within a container loading context, which is crucial for efficient logistis operations, the problem was studied by Toffolo et al. [55] for a real-life application. Their problem was on multiple container loading, where many items were to be packed into containers of different types and sizes. Additional constraints were also considered for this practical problem. The authors employed a two-dimensional heuristic decomposition for the problem and obtained efficient solutions. Another study [39] recently handled a 2D guillotine placement problem, which encompasses a set of small rectangular items to be packed in a rectangular frame, using orthogonal guillotine placements and some constrained patterns. The problem has many industrial applications in glass, wood, marble and granite cutting. The authors proposed mixed integer linear programming formulations for the problem and achieved optimal or near-optimal solutions in realistic run times. In [35], Marmolejo-Saucedo utilized the concept of digital twins for solving a large-scale supply chain optimization problem in a digital platform in real time, in which bin packing and vehicle routing were solved through an interface of a commercial supply chain management platform using heuristic algorithms.

In this study, we consider an extension of the frame packing problem in literature that incurs time-related service level constraints. As mentioned before, while the service level constraints in the defined packing problem may correspond to QoS constraints in data packing, they may as well represent service quality considerations in container or pallet loading. The inclusion of these extra constraints brings our problem closer to a multiple knapsack problem with identical capacities and extra constraints [26]. In our problem, the demand is to be packed into a predetermined number of identical bins, which are defined as rectangular frames. From a practical perspective, one dimension of these identical frames may represent the unit transfer time (e.g. 5 milliseconds), while the other dimension may correspond to a predetermined frequency bandwidth if the problem is considered at the MAC-layer. These frames are depicted in Fig. 1 in the context of wireless data transfer in IoT. On the other hand, the time dimension might represent a time period as long as 4 hours in a pallet or container loading environment, where the second dimension might represent the container cross-section or container storage yard dimensions.

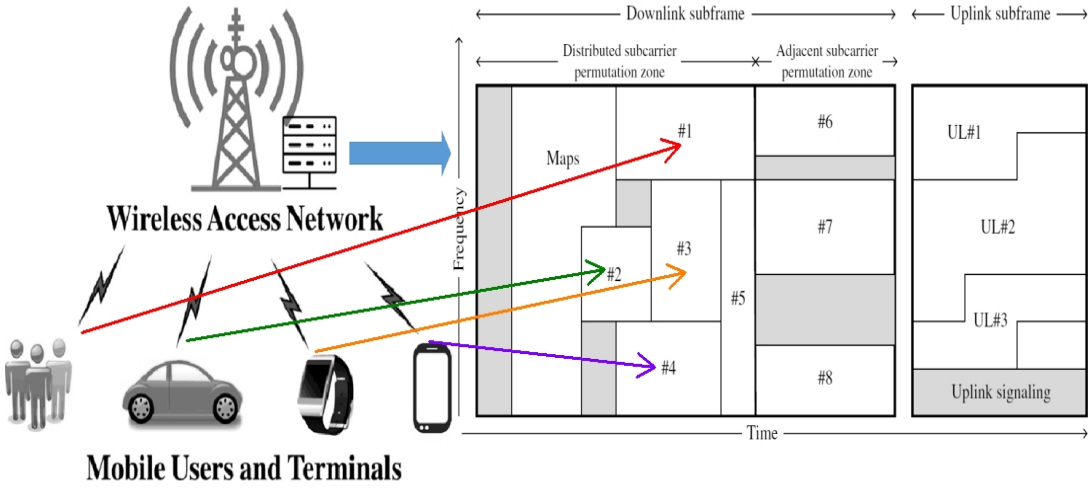

Figure 1. Problem environment in the context of wireless data transfer [3, 32].

Within this general problem definition, the data demand for each user, or the packages to be fitted inside the container are to be packed as rectangular blocks. The total area to be occupied by the rectangles of a single user over a series of frames corresponds to the total demand of that user. However, the dimensions of the rectangle for a user that will be packed in a single frame are not fixed in advance in our problem, in contrast to a traditional 2D-BPP. In other words, while the user's total area to be packed is fixed, the shape of the corresponding rectangle is a decision variable. The user's demand can be partitioned in any way through frames, as long as the QoS constraints are satisfied. This feature of the problem is directly applicable in data transmission context in IoT. While the the total area to be packed (amount of data to be sent) is a parameter, the partitioning of packing through frames is a decision variable. In a pyhsical loading environment, the definition corresponds to allowing the load of a single customer to be divided between many containers. This flexible approach facilitates a more efficient loading operation, as it relaxes the traditional problem and broadens the feasible solution space. Since decreased logistics costs are possible through this approach, it is viable for shipping agencies from a practical perspective as long as a certain service quality is maintained. On the other hand, this complicating aspect of the problem requires nonlinear terms to be incorporated into the mathematical model, as will be explained in the next section.

The model proposed in the next section is a contribution to the literature as it brings a dynamic time dimension into the modeling approach, as well as capturing some problem aspects of practical importance. The specifics of the problem and the mathematical model are explained in detail in the following section.

3. The mathematical model

As it was stated in the previous section, existing work on frame packing aimed at maximizing the packing performance of a single frame through various strip packing techniques. Almost all studies in literature assumed identical user demand rectangles per frame, and they considered the problem in a static manner. In our modeling approach, we include a dynamic resource allocation sequence by allowing the user demand to be partitioned among several frames while maintaining service quality. For this purpose, the 2D structure of the frame packing problem is extended to 3D through additional knapsack and partitioning constraints. Hence, our optimization approach brings a contribution to the literature via modeling the planning horizon as a sequence of frames for representing the dynamic nature of the problem. By allowing varying demand sizes for each user per frame, our objective is to solve multiple interdependent two-dimensional packing problems at once. For this reason, we hereafter name our problem as the Dynamic Rectangular Packing (DRP) Problem.

The objective of the DRP problem is to allocate multiple frames over the planning horizon to the most profitable subset of incoming demand. The characterization of profit would depend on the context where the problem is solved. Apparently, the users who are charged or prioritized more than others by the service provider are likely to be favored in profit maximization. In this respect, the profit values could signify actual monetary returns for demand processing. On the other hand, the profit levels might correspond to customer priorities in other environments. The profit could also be associated directly with demand as in a subset sum problem. Namely, the higher the demand of a user, the higher its profit level.

For any of these definitions of the profit parameter, it might be argued that there is little difference between solving the problem on a single frame or through a sequence of frames. The selection of the most profitable subset corresponds to a one-dimensional knapsack problem in

both cases. However, when it comes to obtaining an optimal or even a feasible packing of user rectangles within a frame, the number of rectangles to be packed is an important parameter for practical purposes. The size of the single frame packing problem can reduce significantly when the users are distributed over a frame sequence. Therefore, instead of a simple averaging of user demand, which would correspond to equal rectangle sizes in each frame, we allow different partitions of demand through frames while maintaining feasibility. This approach brings an opportunity for obtaining acceptable solutions in shorter times. Considering the above characteristics, our model is a realistic representation of two-dimensional packing and optimal scheduling.

Our modeling approach converts a modified multiple knapsack problem with identical knapsack capacities to a combination of 1D knapsack and simultaneous partitioning-and-packing problems. When the profits are identical to demands, the user selection process is analogous to a multiple subset-sum problem. Partitioning is restricted by demand sizes as well as model constraints. It is assumed that each user can be allocated to at most one rectangular area in a frame. We also assume that the total demand is greater than the total available packing area during the planning horizon in order to allow selection.

3.1. **Notations.** Two indices are used for identifying the users and the frames in the model. The user index is denoted by $i \in I = \{1, .., n\}$, where n is the total number of users to be allocated. We index the identical frames by $j \in J = \{1, ..., m\}$, where m is the number of frames in the planning horizon.

Other parameters defined for resource and service level requirements are as follows.

d_i Total demand (in area units/slots) for user i;

s_i Minimum area per frame to be allocated to user i;

ϕ_i Total area to be allocated to user i over the m frames in the planning horizon, computed as $\phi_i = Min\{ms_i, d_i\}$;

p_i Profit obtained from allocating ϕ_i slots to user i over the planning horizon (may be defined in any preferred unit);

λ_i Maximum allowed delay (in frames) for user i before service interruption;

W Frame width;

H Frame height;

WH Frame area;

α_i Minimum number of frames to which user i should be assigned, computed as $\alpha_i = \lceil \phi_i / WH \rceil$;

θ_i The latest frame for assigning user i ($\theta_i \leq \lambda_i$ for ongoing packings, whereas $\theta_i = m + 1$ for upcoming packings);

In addition to the above parameters, the following binary and nonnegative integer decision variables are defined to formulate the model:

$$u_i = \begin{cases} 1, \text{if the demand of user } i \text{ is packed in the planning horizon} \\ 0, \text{otherwise} \end{cases} , \forall i \in I$$

$$z_{ij} = \begin{cases} 1, \text{if a rectangle of user } i \text{ is packed in frame } j \\ 0, \text{otherwise} \end{cases} , \forall i \in I, \forall j \in J$$

x_{ij} x-coordinate of the left-bottom corner of the rectangle for user i in frame j;

y_{ij} y-coordinate of the left-bottom corner of the rectangle for user i in frame j;

w_{ij} Width of the rectangle for user i in frame j;

h_{ij} Height of the rectangle for user i in frame j;

a_{ij} Area of the rectangle for user i in frame j;

$r_{ij} = \begin{cases} \theta_i \text{ - } a_{i1}, \text{ for } j = 1 \\ \mathrm{r}_{i(j-1)} \text{ - } a_{ij}, \text{ for } j > 1 \end{cases}$, $\forall i \in I$, Remaining demand of user i after packing frame j;
$\sigma_{i,j}$, γ_{pqjk} and δ_{pqj} - Auxiliary binary variables.

3.2. **The DRP Model.** Based on the above definitions, the objective function of the DRP model, which aims at maximizing the total profit obtained from packed users, is expressed as follows:

$$Max\ Z_K = \sum_{i \in I} p_i u_i. \tag{1}$$

The objective expressed in (1) may correspond to actual profit maximization or priority maximization, as explained before. Note that this function only regards user demands that are completely packed. For example, if the profit values are chosen to be identical to user demands in an IoT setting, the objective function would correspond to maximizing the amount of whole data packs successfully transmitted.

Demand Satisfaction Constraints

The following constraint sets in (2) and (3) link the decision variables u_i and z_{ij} to one another in order to ensure consistency. These constraints ensure that, if a user's demand is selected to be packed in the planning horizon, at least α_i frames are used for completely packing it. Else, no packing should be made for that user to avoid waste. In IoT context, these constraints guarantee either all information bits of a data packet are successfully delivered, or none at all.

$$\sum_{j \in J} z_{ij} \geq u_i \alpha_i, \forall i \in I, \tag{2}$$

$$z_{ij} \leq u_i,\ \forall i \in I, \forall j \in J. \tag{3}$$

Another demand satisfaction constraint is expressed by the nonlinear relations in (4). This constraint set also handles the partitioning of user demand through the frames by determining the width and height of each user rectangle. The constraint allows the rectangle sizes to vary among frames, as long as the total demand of the user is satisfied.

$$\sum_{j \in J} w_{ij} h_{ij} \geq u_i \phi_i,\ \forall i \in I. \tag{4}$$

QoS Constraints

The maximum delay constraints in (5) ensure service quality by avoiding unwanted delays for any packed user demand. These constraints are especially crucial in IoT for providing a sustainable service. They are also directly applicable to pyhsical loading environments where customers have deadlines for service. These constraints guarantee that, for any selected user, the whole demand of that user is packed until the specified latest frame θ_i.

$$\left.\begin{array}{l} 1 - WH\sigma_{ij} \leq r_{ij} \leq \theta_i(1 - \sigma_{ij}) \\ \sum\limits_{k=j+1}^{j+\lambda_i} z_{ik} \geq z_{ij} - \sigma_{ij}, j \leq n - \lambda_i \end{array}\right\} \sigma_{ij} \in \{0,1\},\ \forall i \in I,\ \forall j \in J. \tag{5}$$

By controlling the values of r_{ij} and z_{ij} variables with parameter λ_i, the the QoS constraints are imposed in (5). Here, an auxiliary binary variable σ_{ij} is used for modeling, which takes the value of 0 if $r_{ij} > 0$ (nonzero remaining demand), and 1 otherwise.

Constraint set (6) is also necessary for assuring QoS in packing via controlling variable r_{ij}. This constraint set enforces that there is no remaining demand after the last frame for any packed user:

$$r_{im} \leq \phi_i(1 - u_i), \ \forall i \in I. \tag{6}$$

The following constraint set in (7) forces the packing of a user, if its demand continues from previous packing (i.e. if its packing is an ongoing transfer). This aspect of our model facilitates dynamic packing through time. The transfers that have started in the previous packing but could not finish are considered, and their QoS constraints are satisfied through (7):

$$m - \theta_i \leq u_i(1 + m - \theta_i) - 1, \ \forall i \in I. \tag{7}$$

Constraint set in (8), together with the information from (7), is necessary for ensuring a bound on the latest frame to start the packing of the ongoing transfers. Namely, these constraints determine the partitioning of the ongoing transfers while maintaining the specified QoS level.

$$\sum_{j=1}^{\theta_i} z_{ij} \geq u_i, \ \forall i \in I. \tag{8}$$

Packing Constraints

The constraint set in (9) limit the boundaries of each user rectangle for fully fitting in a particular frame. These constraints determine the exact placements of the user rectangles within a frame in terms of the x- and y-coordinates.

$$\left.\begin{array}{l} x_{ij} + w_{ij} \leq W \\ y_{ij} + h_{ij} \leq H \end{array}\right\} \forall i \in I, \ \forall j \in J. \tag{9}$$

The packing of user rectangles within frames are further controlled by the constraint set in (10). These define the lower and upper bounds on variables x_{ij}, y_{ij}, w_{ij}, h_{ij}, a_{ij} in relation with z_{ij}. Namely, the relations in this set of constraints ensure that the complete and exact demand of a user is packed as user rectangles through frames, if that user is selected. In IoT data transmission, over-allocation is avoided with this definition.

$$\left.\begin{array}{l} z_{ij} - 1 \leq x_{ij} \leq (W-1)z_{ij} \\ z_{ij} - 1 \leq y_{ij} \leq (H-1)z_{ij} \\ z_{ij} \leq w_{ij} \leq W z_{ij} \\ z_{ij} \leq h_{ij} \leq H z_{ij} \end{array}\right\} \forall i \in I, \ \forall j \in J. \tag{10}$$

The following constraints in (11-13) ensure feasible placements of user rectangles within a frame, preventing overlaps of different users. For identfying users that are allocated to the same frame, the auxiliary variable δ_{pqj} is used in (11). These set of constraints determine which users will be packed together, and therefore important for scheduling and packing decisions. Here, δ_{pqj} assumes the value of 1 if users p and q are packed together in frame j, and 0 otherwise.

$$\left.\begin{array}{l} z_{pj} + z_{qj} \leq \delta_{pqj} + 1 \\ z_{pj} + z_{qj} \geq 2\delta_{pqj} \end{array}\right\} \delta_{pqj} \in \{0,1\}, \ \forall p,q \in I, p < q, \forall j \in J. \tag{11}$$

Using the information provided by δ_{pqj}, constraint set (12) manages the horizontal positions of user rectangles within a frame. The auxiliary variable γ_{pqj1} assumes the value 1 if the rectangle of user p is situated on the left of the packed rectangle for user q in frame j, and γ_{pqj2} is similarly

defined for the rectangle of user q. These constraints guarantee that there will be no horizontal overlaps of user rectangles that are packed in a single frame.

$$\left.\begin{array}{l} x_{pj} + w_{pj} - x_{qj} \leq (1 - \gamma_{pqj1})W \\ x_{qj} + w_{qj} - x_{pj} \leq (1 - \gamma_{pqj2})W \\ \gamma_{pqj1} + \gamma_{pqj2} \leq 2 - \delta_{pqj} \end{array}\right\} \gamma_{pqj1}, \gamma_{pqj2} \in \{0,1\},\ \forall p,q \in I,\ p < q,\ \forall j \in J. \tag{12}$$

These feasible packing constraints are crucial for both data transmission and loading applications. Another such constraint set is expressed in (13), which guarantees non-overlapping vertical positions of user rectangles within a single frame. The information provided by δ_{pqj} is also used in these constraints:

$$\left.\begin{array}{l} y_{pj} + h_{pj} - y_{qj} \leq (1 - \gamma_{pqj3})H \\ y_{qj} + h_{qj} - y_{pj} \leq (1 - \gamma_{pqj4})H \\ \gamma_{pqj3} + \gamma_{pqj4} \leq 2 - \delta_{pqj} \end{array}\right\} \gamma_{pqj3}, \gamma_{pqj4} \in \{0,1\},\ \forall p,q \in I,\ p < q,\ \forall j \in J. \tag{13}$$

In (13), γ_{pqj3} equals 1 if the rectangle of user p is packed below the rectangle of user q in frame j, and γ_{pqj4} is defined vice versa.

Finally, the following constraint set is formulated for an efficient packing of the user rectangles. Namely, constraint set (14) links all binary variables used in (11-13) for avoiding any unnecessary overlap checks, therefore breaking symmetry and bringing efficiency into the model:

$$\left.\begin{array}{l} \gamma_{pqj1} + \gamma_{pqj3} \leq 2(1 - z_{pj} + \delta_{pqj}) \\ \gamma_{pqj2} + \gamma_{pqj4} \leq 2(1 - z_{qj} + \delta_{pqj}) \\ \gamma_{pqj1} + \gamma_{pqj2} + \gamma_{pqj3} + \gamma_{pqj4} \geq \delta_{pqj}) \end{array}\right\} \forall p,q \in I,\ p < q,\ \forall j \in J. \tag{14}$$

The DRP model presented in this section extends the rectangular allocation of the frame packing problem to a pseudo-3D structure by considering the sequential time frames. The model concurrently partitions and packs multiple user demands within the planning horizon. The model as a whole is directly applicable to the MAC scheduler problem of IoT.

4. Upper and lower bounds for the DRP problem

Before presenting our lower and upper bounding procedures for the DRP, we first prove that a special case of DRP where all ϕ_i are integer multiples of W or H is NP-hard in the ordinary sense. The NP-hardness proof is through transformation from the Binary Knapsack Problem (BKP), which is known to be weakly NP-hard [15]. The following theorem states this result formally:

Theorem 4.1. *The DRP problem with either;*

- $\phi_i = \omega_i W,\ \forall i$, *where* ω_i *is a nonnegative integer multiplier, or*
- $\phi_i = \omega_i H,\ \forall i$, *where* ω_i *is a nonnegative integer multiplier*

is NP-hard in the ordinary sense.

Proof. An efficient strip packing solution without any over-allocation is readily available for a selected user subset for both of the special cases defined in the theorem. Since the demand of each user is an exact multiple of the width/length of the frame, an optimal solution assigns a rectangular area to each user exactly equal to its demand. Therefore, our concern is to find whether there exists a feasible subset of users F such that $\sum_{i \in F} p_i \geq P$, where P is a positive integer.

To prove the complexity result, we reduce BKP to this special case of DRP. In an instance of a BKP, we are given a set of items $i = 1, ..., k$. Each item i is associated with two positive integers q_i and v_i, reflecting size and value of i, respectively. Moreover, let Q and V be two positive integers. We are asked to find a subset $U \subseteq \{1, ..., k\}$ such that $\sum_{i \in U} v_i \geq V$ and $\sum_{i \in U} q_i \leq Q$. Given this instance of BKP, we construct the following instance of DRP:

- $n = k$ users,
- $\phi_i = q_i$, $\forall i$,
- $p_i = v_i$, $\forall i$,
- $A = Q/m$ for each of the m frames,
- $P = V$.

Note that solving this instance of DRP will solve the BKP instance, as well. As BKP is NP-hard in the ordinary sense, then so are the given special cases of DRP. □

Corollary 4.1. *The DRP problem is NP-hard in the ordinary sense.*

Proof. In the absence of the specific demand structure in Theorem 4.1, an optimal solution without any over-allocations is not trivial for a selected subset of users. As the general version of a problem is at least as hard as its special cases, the proof follows. □

Next, we establish an upper bound for the DRP problem through a continuous knapsack problem (CKP) relaxation. Consider the following relaxation of DRP, where;

- QoS constraints are omitted,
- Partial demand satisfaction is allowed for each selected user,
- Portion of demand packed to any frame (the area) of any user is not required to be integer, and
- Portion of demand packed to any frame is not required to be a rectangle.

The above relaxation of DRP reduces the problem to the following instance of CKP:

$$Max\ Z_{CKP} = UB_{DRP} = \sum_{i=1}^{n} p_i u_i, \tag{15}$$

subject to

$$\sum_{i=1}^{n} \phi_i u_i \leq mA, \tag{16}$$

where;

- n is the number of items,
- ϕ_i, $\forall i \in \{1, ..., n\}$ are the item sizes,
- p_i, $\forall i \in \{1, ..., n\}$ are the item values,
- $0 \leq u_i \leq 1, i \in \{1, .., n\}$ are the continuous decision variables, and
- mA is the single knapsack capacity.

Theorem 4.2. *The solution of the CKP problem defined by (15) and (16) provides an upper bound for the DRP.*

Proof. An exact solution of the defined CKP, having a packing capacity equal to the total area of all frames, provides a relaxation on DRP as follows. First, it relaxes the rectangular packing requirement and assumes that the demand of the selected user subset can be packed using any tetris-like pattern. Second, it allows a partial demand allocation for the last selected user. Hence, it provides an upper bound for the DRP. □

An exact solution for the continuous knapsack problem defined in Theorem 4.2, hence an upper bound for the DRP, can be obtained in polynomial time using Algorithm 1 [12].

Algorithm 1. Upper Bound for DRP.

(1) Index the items (corresponding to users in the DRP problem) in nonincreasing order of their p_i/ϕ_i ratios.
(2) Find the first item (in order) that does not fully fit into the knapsack. Mathematically, find:

$$s = min\{i : \sum_{j=1}^{i} \phi_i > mA\}.$$

(3) The optimal solution is:

$$u_i = 1, i \in \{1, .., s-1\} \text{ and } u_i = 0, i \in \{s+1, .., n\}$$

$$u_s = (\frac{mA - \sum_{i=1}^{s-1} \phi_i}{\phi_s})$$

and the upper bound value is computed as:

$$UB_{DRP} = \sum_{i=1}^{s} p_i u_i$$

In order to find a simple and tight lower bound for the DRP problem, we propose Algorithm 2, which is presented next.

Algorithm 2. Lower Bound for DRP.

(1) Consider a W by H single frame having area A. Initialize;

$$\phi_i \longleftarrow \phi_i/m, \ \forall i \in I, \ D_{min} \longleftarrow min\{W, H\}, \ D_{max} \longleftarrow max\{W, H\}.$$

(2) Compute the number of unit strips of length D_{min} required for packing each demand. A unit strip of length D_{min} is defined as a rectangle having one dimension as D_{min} and the other as 1. Then, the number of unit strips necessary for user i is:

$$N_i = \lceil \frac{\phi_i}{D_{min}} \rceil, \ \forall i.$$

(3) Index the users in nonincreasing order of their p_i/ϕ_i ratios.

(4) Find the first item (in order) that does not fully fit into the frame as;

$$s = min\{i : \sum_{j=1}^{i} N_i > D_{max}\}.$$

(5) The lower bound solution is:

$$u_i = 1, i \in \{1, .., s-1\} \text{ and } u_i = 0, i \in \{s, .., n\}$$

and the lower bound value is computed as:

$$LB_{DRP} = \sum_{i=1}^{s-1} p_i u_i.$$

We employed the developed above upper and lower bounds in obtaining an exact solution for DRP. Namely, the upper and lower bounding procedures in Algorithm 1 and Algorithm 2 were executed before solving the developed optimization model. This approach yielded good initial feasible solutions for the problem, as well as tight starting upper bounds. The performance of the mathematical model increased significantly and more efficient solutions were obtained, as will be explained in the next section.

5. Computational experimentation

For validating and testing the developed model, a data set was generated for computational experimentation. The computational results are presented in this section. The data set was designed by considering 24 different combinations of the parameter levels. 10 instances were generated from each combination, adding up to a total of 240 instances. The overall experiment design is summarized in Table 1.

Table 1. Parameter levels for generated data.

Parameter name	Levels
n (# of users)	10, 20, 40, and 80
m (# of frames)	2, 4, and 8
$A = WH$ (frame area in slots)	360 (12x30) and 1200 (20x60)
d_i	$U(16, 128)$
p_i	$U(48, 240)$
λ_i	$U(1, m)$
θ_i	$m + 1$ (all customers are new)

The demand and profit values were uniformly generated, independently from each other. All users were assumed to belong to a single demand class, and the profits were generated randomly for all users. Problem generation and solution processes were automated through a console application developed in C# programming language on MS Visual Studio. The model was initially verified and validated for very small instances of the problem, which can be solved by brute-force enumeration in acceptable solution times. In these pilot runs, our nonlinear integer programming model obtained the optimal solutions for DRP instantly. After verification and

validation, the model was run on GAMS 23.9 using the nonlinear solvers BARON 11.5 [54] and SCIP 2.1.2 [1, 2] for optimizing all generated instances. The PC configuration used for computational experiments was 2.30 GHz CORE i7 3610QM, 8 GB RAM. In our experiment, an initial set of runs were obtained without employing any of the proposed bounds. The results are presented in Table 2.

Table 2. Results without incorporating any bounds.

n	$A = WH$	m	# of incumbent solutions	Average gap (%)	Average # of rectangles	Average # of users packed
10	12x30	2	6	7	15	8
20	12x30	2	4	51	12	8
		4	2	89	7	4
40	12x30	2	0	-	-	-
		4	1	96	2	2
		8	0	-	-	-
	20x60	2	4	68	21	13
80	12x30	2	0	-	-	-
		4	0	-	-	-
		8	0	-	-	-
	20x60	2	1	75	19	12
		4	7	93	7	5

A 20-minute runtime limit was employed for all solutions. Within this time limit, no problem instance could be solved to optimality. Furthermore, as it can be observed from the *# of incumbent solutions* column of Table 2, feasible solutions of only 25 instances could be obtained by either solver within the given runtime limit. In particular, the SCIP solver obtained feasible solutions for 23 problem instances in total. Among these solutions, 21 had better objective function values than BARON, which could obtain only 9 feasible solutions.

In Table 2, the gap percentage averaged over 10 instances is presented as a performance measure in the *Average gap (%)* column. The average gap percentage is computed as the difference between the best upper bound solution and the best lower bound solution, as a percentage of the upper bound solution. If an optimal solution is achieved by the solver within the given time limit, it is used as the incumbent solution. The computed gaps are then averaged over the 10 problem instances of that experiment setting. The formula becomes:

$$\%\ Gap = 100(\frac{BestUB_{solver} - Incumbent\ Solution}{BestUB_{solver}}).$$

As it can be seen from Table 2, except for the smallest problem instances with 10 users, the average gap percentages are quite high. The average number of packed rectangles and packed users are also shown in the last two columns of the table.

In Table 2, the objective function value of the problem instance having the smallest gap of 4.3% belonged to a 10-user problem instance. We use this instance for illustrative purposes. The instance includes 10 users with a total demand of 735 (in units or slots, may correspond to bits in IoT). The user demands are to be packed over 2 identical rectangular frames having dimensions 12x30. The corresponding model has 621 decision variables and 1336 constraints. The upper bound obtained for this instance is 1550, which corresponds to an infeasible solution. For this instance, SCIP solver found an incumbent solution with an objective function value

of 1483 in just 13 seconds, which was proven to be optimal via an unlimited-time run. The corresponding optimal packing is illustrated in Fig.2.

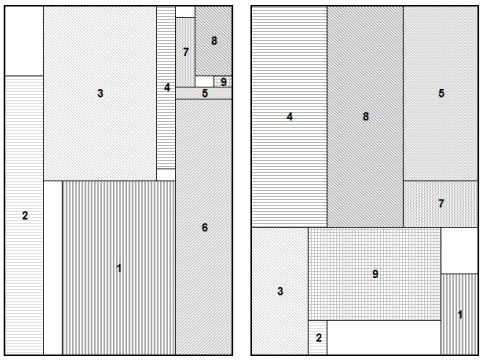

Figure 2. DRP solution for a problem with 10 users packed over 2 frames.

It can be observed from the figure that, 9 of the 10 users are selected for packing in 17 partitions (rectangles) over the two frames, with a total over-allocation of only 10 slots. The complete demand of all selected users are packed into the two frames in a rectangular fashion. It can be observed from the solution that the demands of all selected users are partitioned into the two frames, except for user 6. The complete demand of user 6 is packed in the first frame. All QoS constraints are satisfied by the model. The average area of the resulting user rectangles is 39 in the solution, whereas the average demand per user is 74.

In order to improve solution performance, the developed lower and upper bounds were incorporated into the model as it was explained in the previous section. Then, a second set of computational results were obtained for the same problem instances using only the SCIP solver, as it was found to be significantly more efficient than BARON during the first set of runs. The results are presented in Table 3.

Table 3. Results employing upper and lower bounds.

n	$A : W$xH	m	Average initial gap (%)	Average final gap (%)
10	12x30	2	7.8	4.6
20	12x30	2	13.8	13.1
		4	10.7	10.6
40	12x30	2	14.3	14.3
		4	19.4	19.4
		8	24.2	24.0
	20x60	2	12.2	12.2
80	12x30	2	17.3	17.3
		4	26.5	26.5
		8	38.9	38.9
	20x60	2	18.9	18.9
		4	24.7	24.7

When the lower bounding procedure in Algorithm 2 is employed together with Algorithm 1, initial feasible solutions can be obtained instantly as well as initial upper bounds for all problem instances. Therefore, we report the gap percentages between the initial lower and upper bounds in Table 3. The gaps are averaged over 10 instances and presented in the *Average initial gap (%)* column of the table. The formula for percentage initial gap is:

$$\% \; Initial \; Gap = 100(\frac{UB_{DRP} - LB_{DRP}}{UB_{DRP}}).$$

Average final gaps are also presented in Table 3. It can be observed that the average gap values are far superior in the presence of bounds, when compared to the ones presented in Table 2. The effectiveness of the developed lower and upper bounding procedures, and their contribution in obtaining faster optimal solutions are therefore apparent. For illustrative purposes, the optimal solution corresponding to the same problem instance used in Fig.2 is shown in Fig.3. The optimal solution is obtained in just 6 seconds for the same instance when the bounding procedures are employed.

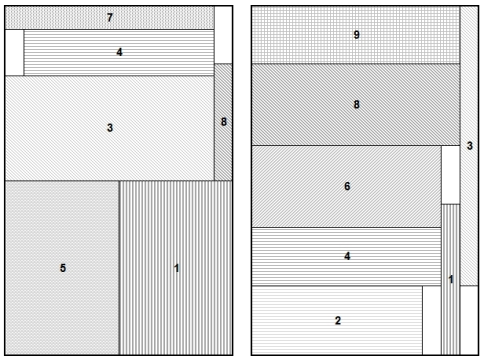

Figure 3. DRP solution for the same problem employing bounds.

It can be observed from Fig.3 that the partitioning of user demands is almost completely different than Fig.2. The new solution has a fewer number of rectangles; the total number is 13 as opposed to the earlier 17 rectangles. Consequently, the average area of the resulting rectangles is also larger (53). This result is due to the execution of the bounding mechanism in Algorithm 2. With the bounds in action, less partitioning is obtained in the optimal solution. It can be seen that the demand of users 2, 5, 6, 7 and 9 are not partitioned, and therefore packed in a single frame.

5.1. **Discussion.** It can be seen from the experimental results that the dominant factor determining the solution time and quality of the developed model is the problem size, which is defined by the number of users and frames involved. Therefore, it can be stated that the DRP model is robust in terms of the demand density parameter.

The computational results illustrate the complex nature of the problem. For the decision makers, the difficulty of obtaining repeated exact solutions of the problem in short periods of time remains to be challenging under dynamic conditions. However, this does not decrease the value of the exact solutions obtained by the developed model with upper and lower bounds. The introduced problem and the developed nonlinear integer programming model has a large variety of applications in L-IoT and other fields of logistics, and brings a novel and flexible 2D approach to a 3D problem.

Our computational analysis suggests that the complexity of the problem naturally might force the utilization of approximate approaches instead of exact solutions in some practical situations where quick solutions are crucial. One of the aims of this study is to present a novel general and representative model that includes important features of the problem, and to give some

applicable insights for solving new problems efficiently. Due to the problem's inherent complexity, it seems that real-time computation of average-size problems use significant computational resources. On the other hand, forthcoming advances in microprocessor technologies will allow high-quality solutions in very small times even for larger instances of the defined problem. With the use of such technology, and together with the efficient bounding procedures developed in this study, our mathematical model can generate optimal or near-optimal solutions in shorter times in time-pressed environments.

6. Conclusion

The bin packing problem with its wide range of applications has attracted the attention of many researchers, and many variants of the problem are yet to be investigated. In some application areas, it is observed that some basic assumptions of the classical packing problems do not apply, such as definite dimensions or rigidity of item shapes. We introduce an exact nonlinear integer programming model for an L-IoT problem that can find its implementations in many fields. By utilizing a versatile two-dimensional packing approach, our model simultaneously handles multiple packing problems with a dynamic time dimension. Simple and effective bounding mechanisms are proposed, as well.

As a result of the modeling approach employed in this study for DRP where simultaneous partitioning and packing of items are involved, an imminent idea for tackling the problem is decomposition into its two natural components. The first subproblem is the assignment of users to frames using maximum delay constraints and partitioning the user demands over the assigned frames. The second subproblem in this case becomes the packing of selected user partitions (rectangles) on their assigned frames. This decomposition might be modeled on the same solvers. Hence, in order to reduce initial problem sizes for obtaining acceptable solutions, a recursive loop between the two subproblems might be employed to solve the DRP problem.

Besides the decomposition approach that could be implemented on commercial or other solvers, similar two-phased heuristics might be designed and coded, as well. For user selection process in profit maximization, knapsack solution algorithms from literature can be employed. Demands of selected users can be randomly distributed with respect to the same QoS constraints employed in DRP. For the packing stage, it can be better to incorporate and adapt some efficient and well-known packing or frame allocation algorithms from literature.

Instead of a single objective such as minimization of the number of partitions or the maximization of utilization, one could also employ a quality definition introducing fuzzy constraints and classifications of user demands with respect to their sizes, and integrate different objectives such as QoS priorities in future studies.

References

[1] Achterberg, T., (2009), SCIP: solving constraint integer programs, Math. Program. Comput., 1(1), pp.1-41.
[2] Achterberg, T., Berthold, T., Koch, T., Wolter, K., (2008), Constraint Integer Programming: A New Approach to Integrate CP and MIP. In: CPAIOR 2008, Lecture Notes in Computer Science, 5015, pp.6-20.
[3] Akyildiz, I.F., Nie, S., Lin, S., Chandrasekaran, M., (2016), 5G roadmap: 10 key enabling technologies, Comput. Netw., 106, pp.17-48.
[4] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., Ayyash, M., (2015), Internet of things: a survey on enabling technologies, protocols, and applications, IEEE Commun. Surv. Tutor., 17(4), pp.2347-2376.
[5] Aswale, P., Shukla, A., Bharati, P., Bharambe, S., Palve, S., (2019), An overview of Internet of Things: architecture, protocols and challenges, In: Satapathy, S., Joshi, A. (eds), Information and Communication Technology for Intelligent Systems, 106, Springer, Singapore, pp.299-308.

[6] Atzori, L., Iera, A., Morabito, G., (2010), The Internet of Things: A Survey, Comput. Netw., 54(15), pp.2787-2805.
[7] Bacioccola, A., Cicconetti, C., Lenzini, L., Mingozzi, E., Erta, A., (2007), A downlink data region allocation algorithm for IEEE 802.16 e ofdma. In: 2007 6th International conference on information, communications & signal processing, IEEE, pp.1-5.
[8] Ben-Shimol, Y., Kitroser, I., Dinitz, Y., (2006), Two-dimensional mapping for wireless OFDMA systems, IEEE Trans. Broadcast., 52(3), pp.388-396.
[9] Bhatia, M., Sood, S.K., (2020), Quantum computing-inspired network optimization for IoT applications, IEEE Internet Things J., 7(6), pp.5590-5598.
[10] Caprara, A., Monaci, M., (2004), On the two-dimensional knapsack problem, Oper. Res. Lett., 32(1), pp.5-14.
[11] Cohen, R., Katzir, L., (2009), Computational analysis and efficient algorithms for micro and macro OFDMA downlink scheduling, IEEE ACM Trans. Netw., 18(1), pp.15-26.
[12] Dantzig, G., (1957), Discrete-variable extremum problems, Oper. Res., 5(2), pp.266-288.
[13] Eliiyi, U., Eliiyi, D.T., (2009), Applications of bin packing models through the supply chain, Int. J. Bus. Manag. Stud., 1(1), pp.11-19.
[14] Eliiyi, U., Nasibov, E., (2010), A fuzzy perspective for two-dimensional packing of variable-sized items. In: 24th Mini EURO Conference Selected Papers, VGTU Technika, pp.177-182.
[15] Garey, M.R., Johnson, D.S., (1979), Computers and Intractability, Freeman, San Francisco, 338p.
[16] Golpara, H., (2020), Optimal integration of the facility location problem into the multiproject multi-supplier multi-resource Construction Supply Chain network design under the vendor managed inventory strategy, Expert Syst. Appl., 139, pp.112841.
[17] Golpra, H. Bahramara, S., (2020), Internet-of-things-based optimal smart city energy management considering shiftable loads and energy storage, J. Clean. Prod., 264, pp.121620.
[18] Golpra, H., Khan, S.A.R., Safaeipour, S., (2021), A review of logistics Internet-of-Things: Current trends and scope for future research, J. Ind. Inf. Integr., 22, pp.100-194.
[19] Gutierrez, J.M., Jensen, M., Henius, M., Riaz, T., (2015), Smart waste collection system based on location intelligence, Procedia Comput. Sci., 61, pp.120-127.
[20] Han, D., Liu, T., Qi, Y., (2020), Optimization of mixed energy supply of IoT network based on matching game and convex optimization, Sensors, 20(19), pp.54-58.
[21] Israeli, A., Rawitz, D., Sharon, O., (2008), On the complexity of sequential rectangle placement in IEEE 802.16/WiMAX systems, Inf. Comput., 206(11), pp.1334-1345.
[22] Iwendi, C., Maddikunta, P.K.R., Gadekallu, T.R., Lakshmanna, K., Bashir, A.K., Piran, M.J., (2021), A metaheuristic optimization approach for energy efficiency in the IoT networks, Softw: Pract Exper., 51(12), pp.2558-2571.
[23] Jiang, S., Cao, J., Wu, H., Yang, Y., (2020), Fairness-based packing of industrial IoT data in permissioned blockchains, IEEE Trans. Industr. Inform., 17(11), pp.7639-7649.
[24] Jiang, N., Deng, Y., Nallanathan, A., Yuan, J., (2020), A decoupled learning strategy for massive access optimization in cellular IoT networks, IEEE J. Sel. Areas Commun., 39(3), pp.668-685.
[25] Kazmi, S., Javaid, N., Mughal, M.J., Akbar, M., Ahmed, S.H., Alrajeh, N., (2019), Towards Optimization of metaheuristic algorithms for IoT enabled smart homes targeting balanced demand and supply of energy, IEEE Access, 7, pp.24267-24281.
[26] Kellerer, H., Pferschy, U., Pisinger, D., (2005), Knapsack Problems, Springer-Verlag, Berlin Heidelberg, 548p.
[27] Kenyon, C., Rmila, E., (2000), A near-optimal solution to a two-dimensional cutting stock problem, Math. Oper. Res., 25(4), pp.645-656.
[28] Kong, F.C., Sakthivel, R., (2021), Uncertain external perturbation and mixed time delay impact on fixed-time synchronization of discontinuous neutral-type neural networks, Appl. Comput. Math., 20(2), pp.290-312.
[29] Korte, B., Vygen, J., (2008), Combinatorial Optimization: Theory and Algorithms, Springer-Verlag, Berlin Heidelberg, 627p.
[30] Li, X., Zhang, K., (2018), Single batch processing machine scheduling with two-dimensional bin packing constraints, Int. J. Prod. Econ., 196(1), pp.113-121.
[31] Lodi, A., Martello, S., Monaci, M., (2002), Two-dimensional packing problems: A survey, Eur. J. Oper. Res., 141(2), pp.241-252.
[32] Lodi, A., Martello, S., Monaci, M., Cicconetti, C., Lenzini, L., Mingozzi, E.C., Eklund, C., Moilanen, J., (2011), Efficient two-dimensional packing algorithms for mobile WiMAX, Manage. Sci., 57(12), pp.2130-2144.

[33] Lodi, A., Martello, S., Vigo, D., (2002), Recent advances on two-dimensional bin packing problems, Discret. Appl. Math., 123(1-3), pp.379-396.
[34] Maddikunta, P.K.R., Gadekallu, T.R., Kaluri, R., Srivastava, G., Parizi, R.M., Khan, M.S., (2020), Green communication in IoT networks using a hybrid optimization algorithm, Comput. Commun., 159, pp.97-107.
[35] Marmolejo-Saucedo, J.A., (2021), Digital twin framework for large-scale optimization problems in supply chains: A case of packing problem, Mob. Netw. Appl., pp.1-17, https://doi.org/10.1007/s11036-021-01856-9.
[36] Martello, S., Toth, P., (1990), Knapsack Problems: Algorithms and Computer Implementations, John Wiley & Sons, New York, 296p.
[37] Martello, S., Vigo, D., (1998), Exact solution of the two-dimensional finite bin packing problem, Manage. Sci., 44(3), pp.388-399.
[38] Martello, S., Pisinger, D., Vigo, D., (2000), The three-dimensional bin packing problem, Oper. Res., 48(2), pp.256-267.
[39] Martin, M., Morabito, R., Munari, P., (2020), A bottom-up packing approach for modeling the constrained two-dimensional guillotine placement problem, Comput. Oper. Res., 115, pp.104851,
[40] Motlagh H.N., Mohammadrezaei, M., Hunt, J., Zakeri, B., (2020), Internet of Things (IoT) and the energy sector, Energies, 13(2), pp.494.
[41] Musaddiq, A., Zikria, Y.B., Hahm, O., Yu, H., Bashir, A.K., Kim, S.W., (2018), A survey on resource management in IoT operating systems, IEEE Access, 6, pp.8459-8482.
[42] Necker, M. C., Kohn, M., Reifert, A., Scharf, J., Sommer, J., (2008), Optimized frame packing for OFDMA systems. In: VTC Spring 2008-IEEE Vehicular Technology Conference, IEEE, pp.1483-1488.
[43] Ohseki, T., Morita, M., Inoue, T., (2008), Burst construction and packet mapping scheme for OFDMA downlinks in IEEE 802.16 systems. In: IEEE GLOBECOM 2007-IEEE Global Telecommunications Conference, IEEE, pp.4307-4311.
[44] Oliveira, J.F., Neuenfeldt Junior, A., Silva, E., Carravilla, M.A., (2016), A survey on heuristics for the two-dimensional rectangular strip packing problem, Pesqui. Operacional, 36(2), pp.197-226.
[45] Omer, D., Aburukba, R., Landolsi, T., (2019), Optimization model for time sensitive IoT requests, In 2019 International Conference on Communications, Signal Processing, and their Applications (ICCSPA), IEEE, pp.1-4.
[46] Pisinger, D., (2002), Heuristics for the container loading problem, Eur. J. Oper. Res., 141(2), pp.382-392.
[47] Prasanth, A., Jayachitra, S., (2020), A novel multi-objective optimization strategy for enhancing quality of service in IoT-enabled WSN applications, Peer-to-Peer Netw. Appl., 13, pp.1905-1920.
[48] Ranjha, A., Kaddoum, G., (2020), Quasi-optimization of uplink power for enabling green URLLC in mobile UAV-assisted IoT networks: A perturbation-based approach, IEEE Internet Things J., 8(3), pp.1674-1686.
[49] Rodoplu, V., Nakip, M., Eliiyi, D.T., Guzelis, C., (2020), A multiscale algorithm for joint forecastingscheduling to solve the massive access problem of IoT, IEEE Internet Things J., 7(9), pp.8572-8589.
[50] Salih, K.O.M., Rashid, T.A., Radovanovic, D., Bacanin, N., (2022), A Comprehensive survey on the internet of things with the industrial marketplace, Sensors, 22(3), pp.730.
[51] So-In, C., Jain, R., Al-Tamimi, A.K., (2009), Scheduling in IEEE 802.16 e mobile WiMAX networks: key issues and a survey, IEEE J. Sel. Areas Commun., 27(2), pp.156-171.
[52] So-In, C., Jain, R., Al-Tamimi, A.K., (2009), OCSA: An algorithm for burst mapping in IEEE 802.16 e mobile WiMAX networks. In: 2009 15th Asia-Pacific Conference on Communications, IEEE, pp.52-58.
[53] So-In, C., Jain, R., Al-Tamimi, A.K., (2009), eOCSA: an algorithm for burst mapping with strict QoS requirements in IEEE 802.16 e mobile WiMAX networks, In: 2009 2nd IFIP Wireless Days (WD), IEEE, pp.1-5.
[54] Tawarmalani, M., Sahinidis, N.V., (2005), A polyhedral branch-and-cut approach to global optimization, Math. Program., 103(2), pp.225-249.
[55] Toffolo, T.A., Esprit, E., Wauters, T., Berghe, G.V., (2017), A two-dimensional heuristic decomposition approach to a three-dimensional multiple container loading problem, Eur. J. Oper. Res.,257(2), pp.526-538.
[56] Tu, M., Lim, M.K., Yang, M.F., (2018), IoT-based production logistics and supply chain system-Part 1: Modeling IoT-based manufacturing supply chain, Ind. Manag. Data Syst., 118(1), pp.65-95.
[57] Tu, M., Lim, M.K., Yang, M.F., (2018), IoT-based production logistics and supply chain system-Part 2: ioT-based cyber-physical system: a framework and evaluation, Ind. Manag. Data Syst., 118(1), pp.96-125.
[58] Wang, T., Feng, H., Hu, B., (2008), Two-dimensional resource allocation for OFDMA system, In: ICC Workshops-2008 IEEE International Conference on Communications Workshops, IEEE, pp.1-5.

[59] Wascher, G., Hauner, H., Schumann, H., (2007), An improved typology of cutting and packing problems, Eur. J. Oper. Res., 183(3), pp.1109-1130.
[60] Wei, L., Oon, W.C., Zhu, W., Lim, A., (2011), A skyline heuristic for the 2D rectangular packing and strip packing problems, Eur. J. Oper. Res., 215(2), pp.337-346.
[61] Wei, L., Zhang, Z., Zhang, D., Leung, S.C., (2018), A simulated annealing algorithm for the capacitated vehicle routing problem with two-dimensional loading constraints, Eur. J. Oper. Res., 265(3), pp.843-859.
[62] Wongthavarawat, K., Ganz, A., (2003), Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems, Int. J. Commun. Syst., 16(1), pp.81-96.
[63] Wu, Y.L., Huang, W., Lau, S.C., Wong, C.K., Young, G.H., (2002), An effective quasi-human based heuristic for solving the rectangle packing problem, Eur. J. Oper. Res., 141(2), pp.341-358.
[64] Yaghmaee, M.H., Moghaddassian, M., Leon-Garcia, A., (2016), Autonomous two-tier cloud-based demand side management approach with microgrid, IEEE Trans. Ind. Inf., 13(3), pp.1109-1120.
[65] Zedadra, O., Guerrieri, A., Jouandeau, N., Spezzano, G., Seridi, H., Fortino, G., (2018), Swarm intelligence-based algorithms within IoT-based systems: A review, J. Parallel Distrib. Comput., 122, pp.173-187.
[66] Zhang, D., Kang, Y., Deng, A., (2006), A new heuristic recursive algorithm for the strip rectangular packing problem, Comput. Oper. Res., 33(8), pp.2209-2217.



Uğur Eliiyi - is an assistant professor of Quantitative Methods at the Department of Business, Izmir Bakircay University, Turkey. He has received a B.S. in Mathematics and an M.S. in Industrial Engineering from Middle East Technical University, Ankara, Turkey. He obtained his Ph.D. degree in Statistics from Dokuz Eylul University, Izmir, Turkey in 2013. His main research interests include combinatorial optimization, smart transportation systems, mathematical modeling and computational intelligence.



Efendi Nasibov - received the B.Sc. and M.Sc. degrees in Applied Mathematics from Baku State University, Baku, Azerbaijan, in 1983, and the Ph.D. degree in Mathematical Cybernetics and the Dr.Sc. degree in Computer Science both from the Institute of Cybernetics, Azerbaijan National Academy of Science, Baku, in 1987 and 2003, respectively. He is currently working as a Full Professor at the Department of Computer Science with the Dokuz Eylul University, Izmir, Turkey. His research interests include the application of fuzzy sets theory and data mining techniques in medicine, optimization and decision-making problems.