The Cutting Stock Problem with Diameter Conversion in the Construction Industry

Deniz Altinpulluk, Nur Banu Altinpulluk
Department of Industrial and Systems Engineering
Wayne State University
4815 4th St, Detroit, MI 48201, USA
deniz@wayne.edu, nurbanu@wayne.edu

Haldun Sural
Department of Industrial Engineering
Middle East Technical University
Universiteler Mahallesi, Dumlupınar Bulvari No: 1
06800 Cankaya, Ankara, Turkey
hsural@metu.edu.tr

Abstract

Solving the one-dimensional cutting stock problem (1-D CSP) has been widely integrated in construction industry for better managing cutting reinforcement steel bars (rebars). To provide the required tensile strength to the structure, the structural designer determines the diameter sizes of rebars, which can be adjusted as long as the rebar to concrete area ratio remains constant. The decision-maker has the option to alter the diameter size to optimize cutting patterns, thereby reducing raw material usage. In our study, we address the cutting stock problem by considering not only the cutting patterns but also the diameter sizes, with the objective of minimizing raw material usage. Unlike the classical cutting stock problem, where the number of pieces to be cut is known in advance, in this case, the number of pieces is not known until the diameter sizes are selected. To tackle this challenge, we propose a solution that employs a pseudo-polynomial formulation. Our computational study shows that converting diameters could enhance the 1-D cutting stock problem's solution quality by up to 6%.

Keywords
Construction industry, 1-5 D cutting stock, diameter conversion, reflect formulation, open dimension.

1. Introduction
The construction industry is a significant contributor to the global economy, with many countries relying on construction to account for a significant percentage of their GDP. Estimates suggest that the construction industry accounts for approximately 6% of the global economy (World Economic Forum, 2016), but its contribution is relatively more significant in the GDP of developing countries. In addition to its economic impact, the construction industry can also have a significant impact on the environment, with construction waste being a major issue. As a result, there is a growing need for cost reduction measures that also promote sustainability. Despite technological advancements and cost-saving measures, there is still considerable potential for further improvements in the construction industry through planning of construction activities. Such improvements could lead to both economic benefits, including cost reduction, and environmental benefits, such as a reduction in construction waste.

Reinforcement steel bars, commonly referred to as rebars, are crucial components in the construction industry and provide tensile strength to the structures. Efficient utilization of reinforcement steel bars is vital for reducing the cost of building structures and conserving natural resources. Rebars are custom-cut to meet specific needs, and one-dimensional cutting stock problem (1-D CSP) is often employed to cut rebars in an economical manner for utilization of raw materials. Suppose that there are m item types of rebars, each of them has length w_j and demand d_j, k different diameter sizes, and a sufficiently large number of stocks ($j = 1, 2,...,m$). The objective is to cut d_j copies for each item type j using the minimum number of stocks so that the total length of items for each stock does not exceed the capacity.

However, the application of the 1-D CSP necessitates categorizing items according to their diameter sizes, leading to a separate solution for each diameter group k. This approach sacrifices the advantage of the convertibility of diameter sizes in the design of structural elements, potentially resulting in waste of resources.

The diameter size and number of rebars required are determined to achieve the necessary tensile strength for each structural element. The designer can alter the number of rebars and diameter sizes by following conversion rules. Basically, opting for larger diameter sizes can lead to a reduction in the required number of rebars, while selecting smaller diameter sizes can increase the required number of rebars. This enables decision maker to group different sizes of diameters and change d_j to generate more efficient cutting patterns.

The diameter dimension, in addition to the length dimension, distinguishes the problem from the classical 1-D CSP. Determining the proper diameter size is a crucial aspect of the decision-making process, making diameter size a decision variable in the problem and distinct from the classical 2-D CSP. The 1-D and 2-D cutting stock problems have received significant attention in the literature. Nevertheless, only a few studies have addressed problems similar to ours, which is referred to as the 1.5-dimensional cutting stock problem (1.5-D CSP). In this paper, we propose a pseudo-polynomial arc-flow formulation, specifically the reflect formulation of Delorme and Iori (2020), to solve the 1.5-D CSP in the construction industry. It is quite a powerful technique even for large size problem instances because it uses the half capacity of the stock size.

The main contributions of the paper can be summarized as follows. First, we present a pseudo-polynomial formulation that efficiently addresses the 1.5-D CSP in the construction industry. Second, to the best of our knowledge, this is the first study to consider the convertibility of diameter sizes for reinforcement steel bars together with the cutting stock problem. Third, we develop computational experiments to demonstrate the value of the proposed decision framework based on the real projects in the construction industry such as hospitals, apartments, business centers, and public buildings.

The organization of the remaining sections is as follows. Section 2 provides a summary of related work on cutting stock problem and its extensions. Section 3 defines our problem, presents background information on it, and summarizes the reflect formulation and graph generation. Section 4 presents the results of our computational experiments and provides a detailed analysis of those results. Finally, Section 5 offers concluding remarks and presents further research opportunities.

2. Literature Review

Cutting and packing problems are classified based on five main characteristics, as proposed by Wäscher et al. (2007). These characteristics include dimensionality, kind of assignment, assortment of small items, assortment of large objects, and shape of small items. The first criterion, dimensionality, classifies problems as one, two, three, or higher dimensional categories. The second criterion, type of assignment, categorizes problems as either input minimization or output maximization. The third criterion, assortment of small items, classifies problems into three main groups: identical small items, weakly heterogeneous assortment, and strongly heterogeneous assortment. The fourth criterion, large object assortment, categorizes problems as having one large object or several large objects. The final criterion, the shape of small items, classifies problems based on their geometric shapes. By considering these five characteristics, cutting and packing problems can be systematically categorized and addressed using appropriate techniques.

In open dimension problem smaller items are fitted into one or multiple larger objects, with at least one dimension being treated as a variable in the problem. Our problem can be classified as an input minimization, open dimension with more than one large object, and cutting stock problem according to typology of Wäscher et al. (2007). Moreover, this class of problems is also defined as the 1.5-dimensional cutting stock problem (1.5-D CSP) in the literature. Haessler (1978) first studied the 1.5-D cutting stock problem for coil slitting in the metal industry. Han and Chang (2015) selected the proper large objects to minimize slitting loss and overproduction with given cutting patterns. They also addressed the issue of overproduction with penalties and developed a pseudo-polynomial time algorithm to solve the problem. Gasimov et al. (2007) studied the 1.5-D assortment problem in the production of corrugated boxes in the paper industry. Paper orders are typically cut from rectangular-shaped rolls, with large objects assumed to be sufficiently long sheets for practical purposes, and infinite in length. Furthermore, the problem involves multiple stock sizes, with several options available in the market. Maintaining an inventory of various stock sizes can be advantageous in achieving lower trim loss levels. They leverage the trade-off between inventory holding cost and trim loss. Song et al. (2006) tackled a real-life 1.5-D CSP in the plastic industry, considering the waste of material and production time under limitations of cutter knife changes, machine restrictions, and due dates.

Minimizing waste of reinforcement steel bar has taken significant attention in the literature due to its significant cost impact on the construction industry. Nadoushani et al. (2018) dealt with minimizing cutting waste of reinforcing steel bars by considering designing issue, lap splicing, which affects ordered lengths of reinforcement steel bar. The concept of flexibility in the length of rebars for specific structural elements enables decision makers to achieve more efficient material utilization during the process of generating cutting patterns. Zheng and Lu (2016) considered rebar material costs related to trim loss and rebar installation costs including labor hours used in rebar stock processing, delivering, placing, and tying. Benjaoran et al. (2019) studied the effect of demand variations on steel bars cutting loss and experimentally showed how the distribution of pieces of length ordered affects material utilization.

There are many different mathematical formulations and solution methodologies employed on cutting stock and bin packing problems including Branch and Price (B&P), column generation, heuristics, and metaheuristics (Vance et al. (1994), Vance (1998), Belov and Scheithauer (2006), Wei et al. (2020)). We refer to the review paper published by Delorme et al. (2016) and Iori et al. (2021) for solution methods related to the cutting stock problems. Delorme and Iori (2020) suggested the reflect formulation, an enhanced version of the arc-flow formulation of Valério de Carvalho (1999) that is formulated as a minimum flow network problem. These are pseudo-polynomial formulations. They become weak because an increase in stock length capacity leads to a substantial increase in network size. However, the reflect formulation uses half the stock length capacity, resulting in a reduction of the number of nodes and arcs.

This study makes a significant contribution by leveraging Operations Research techniques to address the cutting stock problem in the construction industry. Specifically, we propose a solution approach for the 1.5-D CSP that accounts for the conversion of reinforcement steel bar diameters into different sizes and the optimization of cutting patterns in the construction industry. This problem is novel and unique, and to the best of our knowledge, it has not been previously addressed in the literature. Moreover, we demonstrate the practical relevance of our approach by considering a real-world case study in the construction industry, which adds further value to the research.

3. Problem Formulation

Cutting stock problems have been widely studied in the literature and can be categorized as 1-D and 2-D CSPs. The former involves finding optimal cutting patterns for a single dimension, typically length, to minimize raw material waste. This is common in the construction industry for cutting reinforcement steel bars to meet demand. The latter extends the problem to two dimensions, width and length, to generate optimal cutting patterns for pieces. In this study, we propose a novel approach that considers the convertibility of diameter sizes and cutting patterns simultaneously, which we refer to as a 1.5-D CSP. Unlike the classical 1-D CSP, where the sizes of all dimensions are fixed, the 1.5-D CSP allows for the diameter sizes to be changed by the decision maker. This means that the number of pieces being cut is not known until the sizes of the variable dimensions are fixed. The goal is to minimize material usage by determining cutting patterns and selecting the proper sizes of variable dimensions. While the problem is distinct from both the 1-D and 2-D CSPs, it shares similarities with both. The early definition of the problem was made by Haessler (1978) on the coil slitting problem in the industry. The cutting stock problem with diameter conversion in the construction industry can be classified as a 1.5-D CSP due to the variable order sizes of diameter that can be converted by adjusting the required number of rebars.

This section provides essential background information on diameter conversion, which is explored in detail in Section 3.1. Subsequently, we present the mathematical formulations for reflect formulation and graph generation in Sections 3.2 and 3.3, respectively.

3.1. Diameter Conversion

Reinforcement steel bars are essential components in the construction industry, providing the necessary tensile strength to structures when placed in concrete. Given their importance, managing their usage is crucial for construction companies since they constitute a significant cost. In practice, rebars are used in various diameters ranging from 8 to 40 millimeters for different applications such as stirrups, shear walls, slabs, columns, and beams. Typically, rebars come in lengths of 12 meters, the maximum amount that trucks can carry, and cutting processes are performed by workers on a workbench in the construction zone. During the cutting process, the 1-D CSP occurs, and generating cutting patterns that minimize the usage of materials becomes vital.

The diameter size and number of reinforcement steel bars used in construction are determined by structural designers to provide the necessary tensile strength. Once the load of concrete is calculated, the required cross-sectional area between the rebar and concrete is determined for each structural element. The diameter size and number of required rebars can be converted, as long as the necessary ratio of the rebar cross-sectional area to the concrete area remains constant. This allows for greater flexibility in the selection of diameter sizes and reduces the amount of waste generated during the cutting process. The conversion formula is shown below:

$$Required\ number\ of\ rebars = \frac{Necessary\ cross\ section\ area}{\pi(radius)^2}$$

When determining the size and number of rebars for a concrete structure, there is some flexibility as long as a constant ratio between the rebar cross-section area and the concrete area is maintained. This means that the total area of the rebars in the concrete should remain constant. Consequently, a decision maker can adjust the number of rebars by changing the diameter size using the following conversion formula:

(diameter)^2 (required number of rebar) = (new diameter)^2 (new required number of rebar)

The number of required rebars can be reduced by increasing their diameter size, and vice versa, as long as the required ratio of rebar cross-section area to concrete area remains constant. This conversion formula allows decision makers to explore different options and achieve more efficient use of materials. For example, instead of using 100 rebars of Ø16, the decision maker could choose to use 64 rebars of Ø20 while maintaining the necessary cross-section area, as shown below.

$$(Ø16)^2 * 100 = (Ø20)^2 * 64$$

It is important to consider the conversion of diameter sizes before generating cutting patterns, as it can lead to more efficient material utilization. For example, suppose that the orders are as follows: 6 pieces of 3 meters from Ø12, 6 pieces of 7 meters from Ø12, and 3 pieces of 2 meters from Ø14, and the stocks are 12 meters long. If the conversion is not taken into account during pattern generation, the best patterns would be six times 3-7 and a 2-2-2. However, if we convert the last order to diameter 12 using the conversion formula 3x14x14$=$Rx12x12 (where R is the conversion ratio), we get R=4.08, which we can round up to 5 for safety reasons. As a result of changing the diameter from Ø14 to Ø12, we now need to cut 5 pieces of 2 meters instead of the original 3. By doing so, we can generate cutting patterns more efficiently, resulting in five times 3-7-2 and a 3-7 with 2 m loss, thereby reducing material consumption by using leftover parts of rebars from diameter 12. Thus, considering the conversion of diameter sizes before generating cutting patterns can lead to more optimal and cost-effective solutions.

Our problem involves the conversion of diameter sizes using specific rules. To minimize material usage, the decision maker must determine which diameter size to use for each demand item and generate cutting patterns accordingly. Since the selection of diameter sizes and cutting patterns are interrelated, the decision should be made concurrently. Table 1 presents the original diameter requirements for each item and the possible conversions to different diameters. For instance, the fourth item requires 48 pieces of Ø12, which can be converted to 108, 70, 36, and 27 in quantity if a diameter size of Ø8, Ø10, Ø14, and Ø16 is chosen, respectively. Therefore, there is flexibility in the diameter sizes and corresponding quantities for each item, as specified by the designer. Note that the range of convertibility is usually limited to two size steps up or down, but it can be tighter or non-convertible in some cases. By converting these requirements, the decision maker can generate more effective cutting patterns and achieve a higher level of material utilization (Table 1).

Table 1. Conversion Table

Items	Dia.	Length	Quant.	Ø8	Ø10	Ø12	Ø14	Ø16	Ø18	Ø20
1	8	8.50	808	**808**	518	360	-	-	-	-
2	8	3.45	202	**202**	130	90	-	-	-	-
3	10	6.65	265	415	**265**	185	136	-	-	-
4	12	7.00	48	108	70	**48**	36	27	-	-
5	12	9.15	105	237	152	**105**	78	60	-	-

6	14	8.05	203	-	398	277	**203**	156	123	-
7	14	3.15	1685	-	3303	2294	**1685**	1291	1020	-
8	16	1.50	24	-	-	43	32	**24**	19	16

* Dia: Diameter (mm), Length: Stock length (m), Quant: Quantity (units)

3.2. Reflect Formulation

To formulate the problem, we draw upon existing knowledge in literature. Valério de Carvalho's arc-flow formulation (1999) is a useful starting point, but its pseudo-polynomial nature, with O(mc) variables and O(m + c) constraints, limits its effectiveness as the stock capacity increases. To address this issue, Delorme and Iori (2020) introduced the reflect formulation, which utilizes only half of the stock capacity, leading to a substantial reduction in the number of arcs and nodes. This approach results in a more powerful formulation with fewer constraints and variables, making it a more efficient solution method for cutting stock problems. In addition, Delorme and Iori's (2020) reflect formulation also improves computation times, making it a highly effective tool for tackling these problems.

The reflect formulation has properties as listed below:

1. It uses vertices as in the normal patterns but from 0 to $\frac{W}{2}$ and extra vertex, called *R*, whose corresponding size is $\frac{W}{2}$.
2. The formulation converts each item arc *(d, e)* in the arc-flow formulation whose $d < \frac{W}{2}$ and $e > \frac{W}{2}$ whose $d > \frac{W}{2}$ into arc *(d, W-e)*.
3. It eliminates all items and loss arcs *(d, e)* whose $d < \frac{W}{2}$.
4. It adds a last loss arc between the right most vertex before R with R.

A cutting pattern can be represented as a pair of two intersecting paths. Both paths start from 0 and reach the same vertex, but only one of them can pass through the R. This means that only one of the paths can include reflected arcs. Figure 1 illustrates the reflection of these paths (Figure 1).

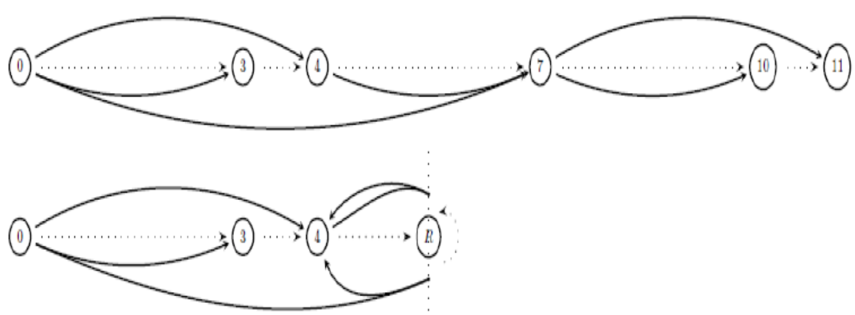

Figure 1. Example for network representation of the reflect formulation (Delorme and Iori (2020))

The Multi Graph $G = (V,A)$ utilized in the reflect formulation consists of a set vertices denoted by $V=0 \cup e \in N$, $0<e<\frac{C}{2} \cup \frac{C}{2}$. The set of arcs, A, comprises two different types of arcs: standard arcs, A_s, and reflected arcs, A_r. A_r include arcs that are reflected from arc (*d, e*) where $d < \frac{W}{2}$ and $e > \frac{W}{2}$ to (*d, W-e*). The notation (*d, e, r*) and (*d, e, s*) represent arcs from *d* to *e* reflected and standard respectively, whereas (*d, e, k*) used for generic arcs from either A_s or A_r.

We modified reflect formulation proposed by Delorme and Iori (2020) by adding diameter dimension according to the structure of the 1.5-D CSP based on sets, parameters and decision variables summarized in Table 2. The reflect formulation for our problem is also provided below (Table 2).

Table 2. Notation of the reflect formulation for 1.5-D CSP

	Sets:
I	Items
J	Diameters
d,e,f	Nodes
A_r	Set of reflected arcs
A_s	Set of standard arcs
A_i	Set of arcs, include both reflected and standard arcs, whose sizes correspond to the length of item *i*
r,s,k	Arc type
$\delta^-_s(e)$	Denotes the set of standard arcs entering *e*
$\delta^-_r(e)$	Denotes the set of reflected arcs entering *e*
$\delta+(e)$	Denotes the set of arcs emanating from *e*
	Parameters:
b_{ij}	Demand of item *i* in diameter *j*
c_j	Unit cost of stock with diameter *j*
	Decision Variables:
ξ_{dekj}	Arc between d and e with arc type k for diameter *j*
m_{ij}	$\begin{cases} 1, \text{ If diameter j is chosen for item } i \\ 0, \text{ Otherwise} \end{cases}$

Mathematical Formulation:

$$Min \sum_{j} \sum_{(d,e,r) \in A_r} c_j \, \xi_{derj} \quad (1)$$

s.t.

$$\sum_{(d,e,s) \in \delta_s^-(\mathrm{e})} \xi_{desj} = \sum_{(d,e,r) \in \delta_r^-(\mathrm{e})} \xi_{derj} + \sum_{(e,f,k) \in \delta^+(\mathrm{e})} \xi_{efkj} \qquad e \in V - \{0\},\ \forall j \in J \quad (2)$$

$$\sum_{(0,e,k) \in \delta^+(0)} \xi_{0ekj} = 2 \sum_{(d,e,r) \in A_r} \xi_{derj} \qquad \forall j \in J \quad (3)$$

$$\sum_{(d,e,k) \in A_i} \xi_{dekj} \geq b_{ij} \, m_{ij} \qquad \forall i \in I,\ \forall j \in J \quad (4)$$

$$\sum_{j} m_{ij} = 1 \qquad \forall i \in I \quad (5)$$

$$\xi_{dekj} : integer \quad \forall (d, e, k) \in A,\ \forall j \in J;\ \ m_{ij} \in \{0,1\}\ \ \forall i \in I,\ \forall j \in J \quad (6)$$

The objective function (1) is designed to minimize the reflected arcs with unit cost of stock c_j from diameter *j*. Since each pattern consists of two colliding paths, where the first path includes only standard arcs and the second path includes reflected arc, the total number of bars utilized is equal to the number of reflected arcs. Thus, minimizing the number of reflected arcs is same as minimizing the total bars used. Constraints (2) ensure that the flow balance is maintained, and they guarantee that amount of flow on standard arcs entering node *e* equals the sum of flow on every arc emanating from node *e*, and flow of reflected arcs entering node *e*. Constraints (3) ensure that the flow balance is maintained for the flow emanating from node 0, which must be equal to twice the amount of reflected arcs to ensure that each pattern is composed of two colliding paths. Constraints (4) guarantee that demand for items is met based on the selection of diameter type *j*. Constraints (5) ensure that each item is cut from only one diameter type. Lastly, constraints (6) represent binary and integrality restrictions for the decision variables.

3.3. Graph Generation for Reflect Formulation

The reflect formulation, developed by Delorme and Iori (2020), offers an improved approach to solving cutting stock problems compared to the traditional arc-flow formulation. As the stock length capacity increases, the arc-flow formulation becomes weaker, but the reflect formulation uses only half of the length capacity, reducing the number of nodes and arcs in the network and making it more powerful for solving large instances quickly. However, like the arc-flow formulation, it is still a pseudo-polynomial formulation and can become weak if the capacity of the stock length increases too much. Furthermore, if the network used in the formulation is not generated efficiently, it can contain many symmetries, which can hinder the solution time. To overcome this challenge, we employed a graph generation algorithm based on the concepts of Delorme and Iori (2020) to generate a network without these symmetries within a reasonable amount of time. Algorithm 1 provides the pseudocode for the graph generation algorithm used in the reflect formulation.

Sorting items in non-increasing order of size during graph generation is a crucial step to eliminate redundant arcs and vertices, as well as to break possible symmetries. Efficient network generation is essential for obtaining solutions within a reasonable time. Implementing these formulations with inefficient networks can significantly increase solution time. Additionally, the reflect formulation requires reflecting arcs from the middle of the stock length capacity, which exists in the range between 0 and capacity/2. If an arc has a head within this range and a tail beyond capacity/2, it should be reflected from the middle of the capacity, changing its tail to capacity minus its original tail. For instance, if the arc is (i, j) and capacity is 11, then (i, 11-j) should replace it. If capacity is not an even integer, both the items' length and capacity should be expanded by multiplying them by two. Finally, after generating the network using Algorithm 1, the problem parameters are provided to the model to be solved by CPLEX.

Algorithm 1. Graph Generation Algorithm for Reflect Formulation

Step 1: Sort items in descending order

Step 2: Initialize paths

Take items whose lengths $\geq C/2$

Generate reflected arcs as (0, C - $length_i$) for those items

Remove these items from set of items

Initialize paths as (0, $length_i$) set them generation 1

Keep tail points of paths and last added items

Take generation 1 as current

Do Step 3, Until no generation of path added

Step 3: Generate Graph

For each tail point of current generation path:

For each item i:

If $length_i \leq$ last added item to tail point:

If $tail\ point + length_i \leq C/2$:

Generate standard arc as ($tail\ point,\ tail\ point + length_i,\ s$)

Add this arc to the network

Keep item length as last added item for this path

Keep new tail point of path of next generation

Else:

Generate reflected arc as ($tail\ point,\ C - tail\ point - length_i,\ r$)

Else:

Go to next tail point of current generation

If there is no new tail point,

Remove duplicated arcs in the network

Take this new generation as current and return step 3

If no generation added at last iteration, then stop.

Step 4: Create loss arcs

Take all vertex in the network and sort them in ascending order

Create loss arcs by connecting them each other consecutively

Step 5: Expand graph into multi-graph by adding diameter dimension

Take network created, set of arcs = *(d,e,k)*

For each arc in the set convert it *(d,e,k,j)* for all *j in J* (set of diameters)

4. Computational Experiments

To provide a realistic analysis of our solution approaches, we utilized real-world construction projects from different application areas, such as hospitals, apartments, business centers, and public buildings. For this purpose, we selected five different projects, each having distinct properties in terms of size and complexity. The main properties that differentiate these projects are the number of items, the number of diameter types, the minimum item length, and the number of different lengths. Table 3 illustrates the properties of these projects. We anticipate that problems with a higher number of items and diameter types are more challenging to solve. Similarly, the number of different lengths and the minimum item length of a project significantly affect the complexity of the problem, as they directly impact the size of the network. Specifically, an increase in the number of different lengths leads to a larger network size, resulting in longer solution times and lower solution quality. Similarly, a decrease in the minimum item length also leads to a larger network size, as it results in more vertices and arcs. For instance, two identical instances with the exception of their minimum item length of 50 cm and 100 cm, respectively, will have different network sizes due to the variation in minimum item length (Table 3).

Table 3. Properties of projects

Project Name	Number of Items	Number of Different Diameter sizes	Minimum Item Length	Number of Different Lengths
Project-1	1835	6	20 cm	268
Project-2	1940	12	40 cm	254
Project-3	1540	7	62 cm	126
Project-4	11313	8	60 cm	457
Project-5	2900	5	43 cm	218

To conduct experiments, we developed an implementation in C++ and used CPLEX 12.8.0 Concert Technology with Intel (R) Core (TM) i7-4790 CPU @3.10 GHz and 16GB RAM. The experiments were performed with a time limit of five hours. We aimed to investigate the impact of converting diameter sizes on the efficiency of our solution approaches. To this end, we solved five real construction projects without considering diameter conversion and then solved the same instances after converting the diameters. Without diameter conversion, each item requires cutting from its original diameter, creating one-dimensional cutting stock problems. We also solved Project-4 by rounding their length sizes to reduce the problem's complexity, which is presented in the reflect-rounded column of Table 4. When the problem cannot be solved in a reasonable amount of time, length sizes can be rounded by one decimal. This allows us to reduce the network size and problem complexity. The computational results indicate that considering diameter conversion when preparing a cutting plan of rebars provides significant advantages. For instance, our approach can save up to 6% on rebar usage in Project-3. Table 4 provides further details on our computational results (Table 4).

Table 4. 1.5-D CSP % gain comparison to 1-D CSP

	% Gap		Gain Compared to 1-D CSP (%)
Project Name	**Reflect**	**Reflect-Rounded**	
Project-1	0.1	-	5.6
Project-2	0.1	-	5.9
Project-3	0.1	-	6.0
Project-4	5.5	0.5	4.3
Project-5	0.1	-	0.9

5.Conclusion

The construction industry has many improvement opportunities by leveraging Operations Research methodologies. One of the most significant cost components in construction is reinforcement steel bars. Although the 1-D CSP has been extensively studied in literature and can provide efficient cutting patterns for rebars, incorporating diameter convertibility into the decision-making process can lead to even better patterns and reduced trim loss. In this paper, we introduce a new problem in the construction industry that considers both diameter selection and cutting pattern generation. Our computational experiments, which utilize real data from construction projects, show that project properties such as item count and minimum length have a significant impact on solution quality. Additionally, we propose a solution approach that involves rounding up length sizes to reduce network size and increase the effectiveness of the reflect formulation in attacking the 1.5-D CSP. Our results demonstrate that incorporating diameter convertibility can lead to cost savings of up to 6% for the 1.5-D CSP.

References

Belov, G. and Scheithauer, G., A branch-and-cut-and-price algorithm for one-dimensional stock cutting and two-dimensional two-stage cutting. *European Journal of Operational Research,* 171(1), 85-106, 2006.

Benjaoran, V., Sooksil, N. and Metham, M., Effect of demand variations on steel bars cutting loss, *International Journal of Construction Management*, 19(2), 137-148, 2019.

Delorme, M., Iori, M. and Martello, S., Bin packing and cutting stock problems: Mathematical models and exact algorithms, *European Journal of Operational Research*, 255(1), 1-20, 2016.

Delorme, M. and Iori, M., Enhanced pseudo-polynomial formulations for bin packing and cutting stock problems, *INFORMS Journal on Computing*, *32*(1), 101-119, 2020.

Gasimov, R. N., Sipahioglu, A. and Saraç, T., A multi-objective programming approach to 1.5-dimensional assortment problem, *European Journal of Operational Research*, 179(1), 64-79, 2007.

Haessler, R. W., A procedure for solving the 1.5-dimensional coil slitting problem, *AIIE Transactions*, 10(1), 70-75, 1978.

Han, Y. T. and Chang, S. Y., A subset sum approach to coil selection for slitting, *International Journal of Industrial Engineering*, 22(3), 2015.

Iori, M., De Lima, V. L., Martello, S., Miyazawa, F. K. and Monaci, M., Exact solution techniques for two-dimensional cutting and packing. *European Journal of Operational Research*, *289*(2), 399-415, 2021.

Nadoushani, Z. S. M., Hammad, A. W., Xiao, J. and Akbarnezhad, A., Minimizing cutting wastes of reinforcing steel bars through optimizing lap splicing within reinforced concrete elements*, Construction and Building Materials*, 185, 600-608, 2018.

Shaping the future of construction a breakthrough in mindset and technology, In *World Economic Forum*. Available: https://www3.weforum.org/docs/WEF_Shaping_the_Future_of_Construction_full_report__.pdf, May, 2016.

Song, X., Chu, C. B., Nie, Y. Y. and Bennell, J. A., An iterative sequential heuristic procedure to a real-life 1.5-dimensional cutting stock problem, *European Journal of Operational Research*, 175(3), 1870-1889, 2006.

Valério de Carvalho, J. M., Exact solution of bin-packing problems using column generation and branch-and-bound, *Annals of Operations Research*, 86(0), 629-659, 1999.

Vance, P. H., Barnhart, C., Johnson, E. L. and Nemhauser, G. L., Solving binary cutting stock problems by column generation and branch-and-bound*.* *Computational Optimization and Applications*, 3, 111-130, 1994.

Vance, P. H., Branch-and-price algorithms for the one-dimensional cutting stock problem. *Computational Optimization and Applications*, *9*, 211-228, 1998.

Wäscher, G., Haußner, H. and Schumann, H., An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183(3), 1109-1130, 2007.

Wei, L., Luo, Z., Baldacci, R. and Lim, A., A new branch-and-price-and-cut algorithm for one-dimensional bin-packing problems. *INFORMS Journal on Computing*, *32*(2), 428-443, 2020.
Zheng, C. and Lu, M., Optimized reinforcement detailing design for sustainable construction: Slab case study, *Procedia Engineering*, 145, 1478-1485, 2016.

Biographies

Deniz Altinpulluk currently pursues his Ph.D. studies at the Industrial Engineering Department of Wayne State University. He received his BS and MS degrees from the Industrial Engineering Department of Middle East Technical University in Turkey. His research focuses on robust optimization models for prognostics-driven operations & maintenance and real-time predictions of remaining useful life distributions for components subjected to dynamic degradation processes. In addition, he leverages machine learning and federated learning methodologies into energy and manufacturing systems areas.

Haldun Sural is an emeritus professor in the IE Department of Middle East Technical University (METU), where he earned his Bachelor's, Master's, and PhD degrees. He has held teaching and research positions at various institutions, including METU, INSEAD (France), and UW and WLU in Canada. Dr. Sural's research portfolio includes a diverse range of applied works, such as the design of supply/production/distribution networks, restructuring a large humanitarian aid organization, and waste management system design. He has also contributed to the academic community as a member of the editorial board of the Turkish Journal of Industrial Engineering and as past president of the OR Society of Turkey. Dr. Sural's research interests span methodology, modeling, and applications in OR, as well as the history of OR in Turkey.

Nur Banu Altinpulluk is a current Ph.D. candidate in the Industrial and Systems Engineering Department at Wayne State University. She earned both her master's and bachelor's degrees from Middle East Technical University in Turkey. Her research lies in advancing the integration of machine & reinforcement learning and stochastic optimization in the energy systems domain. In addition, she actively participated in working on projects in the areas of condition monitoring, prognosis, federated learning, and distributed computing.