DOI: 10.2478/ijasitels-2024-0012

Logistics Optimization for Resource Allocation and Scheduling Using Time Slots

Cezar-Marian Papară [1], *Ștefan-Horia Schirliu* [2]

[1]*PhD student at "Alexandru Ioan Cuza" University, Faculty of Computer Science, 16 Berthelot St., Iasi, 700506, România, cezarmarian98@gmail.com*

[2]*Master's student at Faculty of Sciences, "Vasile Alecsandri" University of Bacău, 157 Cal. Mărășești, Bacău, 600115, România, stefanschirliu@gmail.com*

Abstract

In the field of logistics, efficient scheduling and resource allocation are essential for ensuring the seamless flow of goods through transportation networks. This paper addresses the Interval Scheduling Problem, a combinatorial optimization challenge, in the context of logistics planning for goods transportation. The study examines how optimized appointment scheduling and resource allocation can enhance the performance of transportation networks. By combining theoretical insights, algorithmic solutions, and practical applications, this work proposes a comprehensive approach grounded in mathematical models that account for time, resource, and capacity constraints, alongside a computational implementation. Utilizing advanced computational techniques and real-time data integration, the proposed solutions aim to increase operational effectiveness and competitiveness while reducing costs in transportation logistics.

Keywords: Interval Scheduling Problem, Time Slots, Resource Allocation, Combinatorial Optimization, Logistics, Transportation Networks.

1 Introduction

In today's interconnected world, large-scale transportation networks play a critical role in facilitating the efficient movement of goods and services. These networks form the backbone of global trade, supply chains, and local distribution systems [1]. Improving the efficiency of such networks is essential, as it directly impacts economic growth, operational costs, and service quality. Consequently, optimizing logistics operations within transportation networks has become a central focus of both academic research and practical development efforts [2].

While our previous research has predominantly addressed external aspects of transportation networks—such as routing and path optimization, including the well-known Traveling Salesman Problem (TSP) and Vehicle Routing Problem (VRP) [14], [15]—a less explored area is the internal logistics within network nodes. This paper shifts the focus to understanding how goods are processed once they enter a node, such as a warehouse or distribution center, and how constraints and optimization decisions within these nodes—related to resource allocation, timing, and capacity—affect the overall flow and efficiency. Although past work has utilized heuristics, greedy algorithms, and dynamic programming for optimization [16], [17], [18], [19], there is still much to be explored regarding the internal logistics operations of these nodes.

This study seeks to address this research gap by investigating the Interval Scheduling Problem (ISP) [3] as it applies to logistics within transportation networks. The ISP focuses on allocating tasks or activities within specific time intervals to maximize operational efficiency and minimize conflicts or overlaps in resource usage. In the logistics context, this involves scheduling deliveries, transport operations, and inventory management tasks across different logistical points in a way that ensures smooth and uninterrupted operations [4].

Classified as NP-hard due to its computational complexity [5], ISP necessitates advanced algorithmic solutions for effective resolution. Researchers and practitioners rely on dynamic programming techniques to decompose the scheduling problem into smaller, manageable subproblems. By recursively solving these subproblems and storing their solutions, dynamic programming optimizes resource allocation and minimizes scheduling conflicts within the constraints of time availability and resource capacity.

The research question at the core of this study is: How can the Interval Scheduling Problem be effectively applied to optimize internal logistics within transportation network nodes, accounting for temporal, resource-based, and capacity constraints? By addressing this question, the study aims to contribute to the development of more efficient and reliable scheduling strategies that improve both operational performance and service quality in logistics operations. The findings are expected to provide practical insights into the management of internal logistics challenges and offer robust solutions for complex scheduling problems in this domain.

The structure of the paper is as follows: Section 1 provides a review of both classical and contemporary literature on scheduling problems. Section 2 presents a mathematical model that addresses key constraints, such as operating hours, national holidays, team holidays, and the prevention of overlapping appointments within the same time intervals and available resources. Section 3 introduces the implemented solution using state-of-the-art programming frameworks. Finally, Section 4 analyses data sets and results, culminating in a set of conclusions based on the study's findings.

2 Literature Review

2.1 Challenges of Uncertainty in ISP

One of the key challenges in implementing ISP in logistics is managing uncertainty in processing and setup times. Article [6] highlights these challenges, emphasizing the need for resilient scheduling techniques such as robust optimization, stochastic programming, and scenario-based approaches to handle uncertainties effectively. These techniques are essential for minimizing disruptions and ensuring effective resource allocation in logistics operations.

2.2 Mathematical Models for Optimization in Logistics

A substantial body of literature focuses on the development of mathematical models to optimize logistics operations. Article [7] introduces several crucial models such as linear programming, integer programming, and constraint programming, which are widely used for optimizing deliveries, vehicle routes, and work shifts. These models maximize resource utilization and minimize operational costs, playing a pivotal role in ISP implementation across various logistical nodes.

2.3 Real-Time Decision Making in ISP

The integration of real-time decision-making techniques into ISP is another significant theme in the literature. Article [8] explores the use of predictive models, including machine learning, to enhance scheduling accuracy. By incorporating online algorithms with predictive analytics, schedules can be dynamically adjusted based on real-time data, allowing logistics operations to respond swiftly to unexpected changes and improve overall operational efficiency.

2.4 Heterogeneous Entities in ISP

Another critical aspect of ISP in logistics is the consideration of heterogeneous entities—both in terms of agents and machines. Article [9] focuses on how heterogeneous agents with diverse preferences and constraints can complicate scheduling, necessitating robust allocation mechanisms that ensure fairness and efficiency without overlap. Similarly, article [10] addresses the complexities introduced by nonidentical machines with varying processing capacities, demonstrating how task allocation must be optimized to fully utilize these resources and enhance scheduling efficiency.

2.5 Task Sequencing in ISP

The sequencing of tasks is crucial in optimizing logistics operations (Fig. 1). Article [11] examines the permutation flow-shop scheduling problem, where tasks must be processed in a specific order across multiple machines. By applying constrained programming techniques, the study improves the sequencing of tasks to reduce delays and maximize throughput, which is directly applicable to ISP challenges in logistics.

2.6 Resource-Related Constraints in ISP

Scheduling in logistics is also influenced by various resource-related constraints, including team capacity, availability, and holidays. Article [12] investigates these variables using integer linear optimization to develop flexible scheduling solutions. These solutions accommodate both employee needs and operational requirements, ensuring sufficient staffing levels while preventing conflicts in ISP.

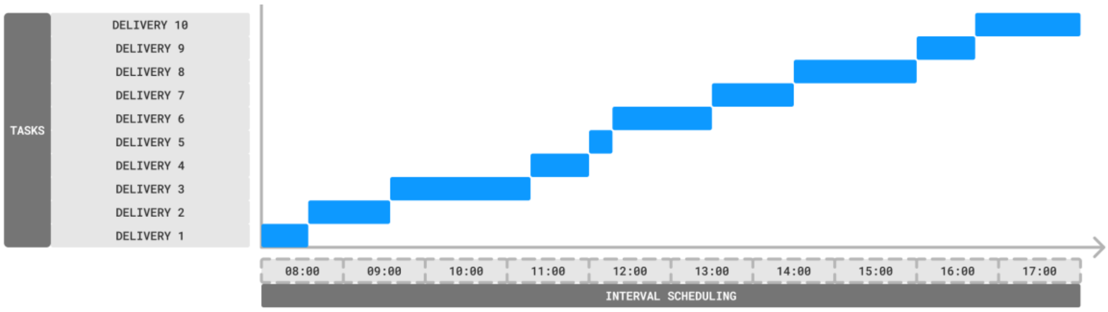

Figure 1 - A solution to an Interval scheduling problem

2.7 Heuristic Methods for ISP Optimization

Given the complexity of scheduling problems, heuristic methods play a critical role in providing efficient solutions. Article [13] demonstrates the effectiveness of heuristics in addressing NP-complete problems, such as those encountered in ISP. By using heuristic techniques, logistics operations can achieve near-optimal scheduling solutions, even in highly dynamic environments, ensuring efficient resource allocation and minimizing scheduling conflicts.

The literature review has outlined the primary challenges and existing solutions in applying ISP to logistics, including uncertainty, mathematical modeling, real-time decision-making, heterogeneous entities, and heuristic methods. Building on these findings, this paper will focus on developing a mathematical model that addresses key constraints—such as timing, resource capacity, and availability—applied to the logistics component at the entry point of a node within a transportation network. This model aims to optimize internal logistics operations, ensuring efficient resource allocation and scheduling.

3 Mathematical Model Overview and Contribution

3.1 Key Contribution and Innovation

The primary contribution of this paper is the development of a new, custom mathematical model that addresses specific constraints related to logistics within large-scale transportation networks. While existing studies on the ISP have addressed certain constraints, the novelty of this model lies in its ability to serve as a reference framework for managing the inbound flow of goods at network nodes, considering real-time constraints such as working hours, team availability, holidays, and overlapping operations.

This model is particularly useful for second-mile or middle-mile logistics operations, such as transporting goods from logistics centres to local warehouses and can be adapted for use by both private and public entities, including retail companies and courier services. A key feature of the proposed framework is its ability to dynamically assess the availability of a node based on the constraints, assigning time slots that ensure efficient processing of goods.

3.2 Future Application and Hybrid Integration

Beyond its immediate utility, this model offers significant potential as a reference for future studies and hybrid implementations. Specifically, it can be integrated with dynamic routing methods, such as those used to solve TSP. By combining the scheduling framework with routing algorithms, future systems can construct dynamic transportation routes based on real-time node availability, evaluated for each specific time slot. This integration would enable logistics operators to select optimal routes that consider not only the shortest or most cost-effective path but also the operational readiness of each node at a given moment in time.

In this way, the proposed model paves the way for hybrid scheduling and routing solutions that can improve the overall efficiency of transportation networks, allowing for more intelligent decision-making and resource allocation in logistics operations.

3.3 Mathematical Model

3.3.1 Definition of the transportation network

To address the complexities of logistics scheduling within transportation networks, we must first define the transportation network itself and structure it as a graph. In this graph, nodes represent logistical points such as warehouses, distribution centres, or retail stores, while edges depict the connections or routes between these nodes (Fig. 2). For the ISP, as applied to logistics units within this network, we will construct concepts such as time slots, which refer to the flow of logistic arrivals at these points, considering various time, capacity, and resource constraints. We will analyse the problem by considering diverse constraints, detailed further, and develop mathematical models that we will use to build our solution.

When considering these constraints, it is crucial to consider a wide array of heterogeneous factors that influence operational efficiency. These factors include varied working hours depending on the logistics node, differing team sizes and availability, and distinct types of merchandise, each requiring customized processing times. Additionally, operator preferences for scheduling arrivals at logistics points at different hours or frequencies, rather than in continuous succession, must be accounted for. Our proposed solution will integrate these diverse aspects and will be applied to a large-scale transportation network.

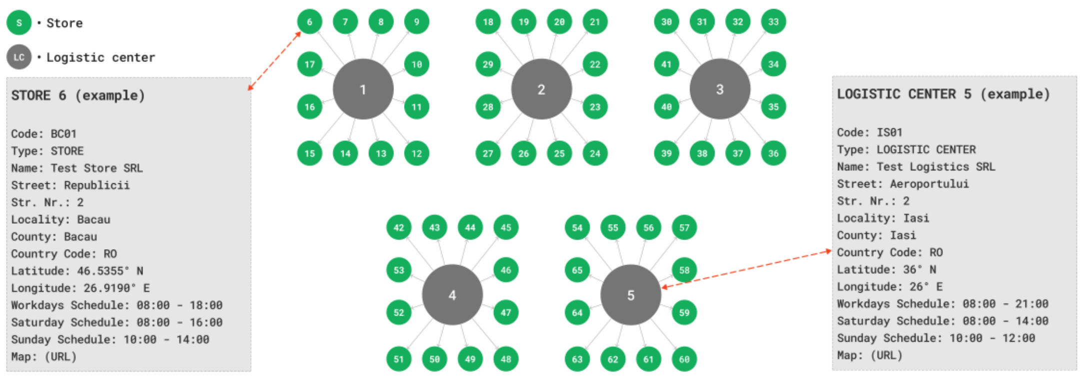

Figure 2 - Representation of Transportation network

To enumerate all the constraints identified for our problem, they are as follows:

- Availability only between Logistic Point Working Hours
- Existence or absence of team holidays or national holidays set by the government
- Overlapping with other operations or deliveries already scheduled in the same slots.
- Type of merchandise involved in the transportation process (palletized, non-palletized, mixed, oversized, etc.), and different processing times for each
- Continuous system verification to ensure that time slots do not become invalid by transitioning into the past
- Real-time calculation of availability for scheduling

By addressing these factors through advanced mathematical models, we aim to develop a robust scheduling solution that enhances the overall performance of transportation networks, ensuring that goods are handled efficiently, and resources are optimally utilized.

To define the transportation network for the interval scheduling problem applied to logistics, we consider a network consisting of a total of a work points (nodes), where b represents the number of logistics centres and (a – b) represents the number of stores, forming a complete graph. This means each node is connected to every other node, allowing for logistics arrivals from one point to another. Mathematically, the transportation network is represented as a graph G = (V, E), where:

- V is the set of vertices (nodes), representing the logistics centres and stores.
- E is the set of edges (connections) between the nodes

$V = \{v_1, v_2, \ldots, v_a\}$ where $v_1, v_2, \ldots, v_b$ represent the logistics centers and $v_{b+1}, v_{b+2}, \ldots, v_a$ represent the stores.

The set of edges E is defined as: $E = \{(v_i, v_j) \mid v_i, v_j \in V, i \neq j\}$, this means that for any two distinct nodes v_i and v_j: $(v_i, v_j) \in E$ and $(v_j, v_i) \in E$. Network Properties:

- The number of nodes (vertices) in the graph G is $|V| = a$
- The number of edges (connections) in the complete graph G is: $|E| = a\times(a - 1)$.

By defining the transportation network this way, we ensure a fully connected structure that supports robust and efficient scheduling solutions. This comprehensive framework allows us to address the interval scheduling problem in logistics effectively, accommodating the diverse and interconnected nature of logistics operations.

3.3.2 Definition of Logistics Scheduling at a Point

To define logistics scheduling mathematically, we represent it as a set named SCHEDULE with specific properties. A logistics schedule at a point involves the allocation of time slots for the arrival and departure of goods, considering various constraints such as working hours, equipment availability, and processing times for different types of merchandise. We define the logistics SCHEDULE as follows:

$$SCHEDULE = \{(O, D, T_s, T_e, Ct, St) \mid O \in V_{centers}, D \in V_{stores}, (v_O, v_D) \in E, T_s, T_e \in T, Ct \in C, St \in STATUSES\} \quad (1)$$

, where:

- O is the origin point (logistics center) from the subset $V_{centers} \subset V$ where $V_{centers} = \{v_1, v_2, \ldots, v_b\}$

- D is the destination point (store) from the subset $V_{stores} \subset V$ where $V_{stores} = \{v_{b+1}, v_{b+2}, \ldots, v_a\}$
- T_s is the start time of the logistics schedule, from the set of time slots T
- T_e is the end time of the logistics schedule, from the set of time slots T
- Ct is the cargo type from the set of cargo types C
- St is the schedule status from the types of STATUSES = {"New", "Processing", "Canceled", "Completed" etc.}

3.3.3 Dependencies (indirect constraints)

3.3.3.1 Dependency on Cargo Type

The logistics SCHEDULE depends on the cargo type Ct, which influences the processing time. Different types of merchandise require different handling times, affecting the start and end times of the logistics schedule. $Ct \in C = \{C_1, C_2, \ldots, C_m\}$, where each C_i represents a different type of cargo (e.g., palletized, non-palletized, mixed, oversized) with specific processing times $\tau(C_i)$.

$$\tau(Ct) = T_e - T_s \qquad (2)$$

3.3.3.2 Dependency on Inventory Source Teams

The logistics SCHEDULE also depends on the availability of teams at inventory sources. Let Lt be the set of all teams at logistics nodes V. $Lt = \{Lt_1, Lt_2, \ldots, Lt_n\}$, where each Lt_j is associated with a specific node $v \in V$ and has its own availability. It's necessary to have at least one team defined for the destination node (point) ($\forall v_j \in V, \exists Lt_k$, with $Lt_k(\text{node}) = v_j$), and not be on holiday. For checking an existing team availability, we assume that team is available every day, except when there are team holidays. Team holidays are defined as a set of associations between teams and specific days intervals:

$LtH = \{(LtH_k, x_l) \mid LtH_k \in LtH, x_l \in X\}$, where x_l represents a team holiday period
$X = \{(startTime, endTime) \mid startTime, endTime \in T$ (time slots)
$, endTime > startTime\}$ (3)

3.3.3.3 National Holidays

National government holidays can also impact the scheduling process. Let H be the set of national holidays for a given year. $H = \{h_1, h_2, \ldots, h_m\}$. Similarly, as team holidays, we define a set of associations between national holidays and specific days intervals:

$H = \{(H_n, x_o) \mid H_n \in H, x_o \in X\}$, where x_o represents a national holiday period (4)

3.3.4 Logistic constraints

Availability only between Logistic Point Working Hours:
$(T_s, T_e) \cap (Lwh_s, Lwh_e) \mathrel{!=} \emptyset, \forall T_s, T_e \in T$
, where $Lwh_s, Lwh_e \in \{hours\}$ represents the (start and end) working hours
(for a specific weekday) of (logistic) point Lt_j (5)
Overlapping with Other Unfinished Operations:
$(T_s, T_e) \cap (T_{s'}, T_{e'}) = \emptyset$

$, \forall (T_s, T_e), (T_{s'}, T_{e'}) \in T$ and $\forall$ St, St' $\in$ STATUSES \ {"Finished", "Canceled"} (6)

Continuous System Verification:

$T_s \geq$ current time, $\forall\, T_s \in T$ (7)

Real-Time Calculation of Availability:

$A\,(T_s, T_e) \in \{true, false\}$, availability status is based on real-time data (8)

3.3.5 Comprehensive Scheduling Equation

Finally, we define the comprehensive scheduling equation, incorporating all constraints ((1), (2), (3), (4), (5), (6), (7), (8)) for a specific logistics SCHEDULE:

$$\begin{aligned}
&\text{SCHEDULE} = \{(O, D, T_s, T_e, Ct, St) \mid (v_o, v_d) \in E, (T_s, T_e) \cap (Lwh_s, Lwh_e) \neq \emptyset \\
&, (T_s, T_e) \cap H = \emptyset, \text{count}(Lt) \geq 1, \text{ with } Lt(node) = D, (T_s, T_e) \cap LtH = \emptyset \\
&, (T_s, T_e) \cap (T_{s'}, T_{e'}) = \emptyset, \tau(Ct) = T_e - T_s, T_s \geq \text{current time}, A(T_s, T_e) = true \\
&, \forall\, O \in V_{centers} \subset V \text{ where } V_{centers} = \{v_1, v_2, \ldots, v_b\}, \forall\, D \in V_{stores} \subset V \\
&, \text{where } V_{stores} = \{v_{b+1}, v_{b+2}, \ldots, v_a\}, \forall (T_s, T_e), (T_{s'}, T_{e'}) \in T, T_e > T_s, T_{e'} > T_{s'} \\
&, \forall\, Ct \in C, \forall\, St, St' \in \text{STATUSES} \setminus \{\text{"Finished", "Canceled"}\}\}.
\end{aligned} \tag{9}$$

This comprehensive mathematical framework (represented visually in Fig. 3) ensures that logistics scheduling is efficient, considering all necessary constraints and dependencies.

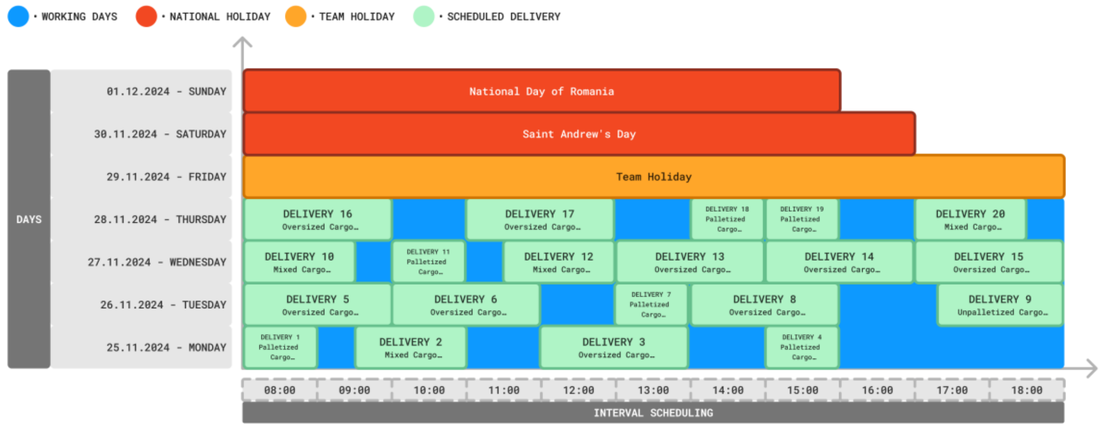

Figure 3 - Visual representation of proposed framework for restricted Interval Scheduling Problem

4 Implementation of the Proposed Solution

In this section, we will discuss the implementation of the solution for the interval scheduling problem described mathematically in the previous section. The application we have developed leverages well-known and widely used frameworks to ensure robustness and efficiency. The back-end processing is managed using PHP Symfony, a powerful framework that facilitates scalable and maintainable web applications. For data storage, we utilize a well-structured MySQL database, ensuring reliable and efficient data management. At the front-end, our application employs a variety of technologies to enhance the user interface and user experience. These include Node.js, JavaScript, jQuery, AJAX, Twig, Bulma and CSS. These technologies collectively contribute to a responsive and interactive user interface. To ensure portability and streamline development and deployment processes, the application architecture is containerized using Docker. This approach not only enhances portability but also simplifies the management of dependencies and the deployment process across different environments. The application's functionality relies on the seamless communication between the frontend and backend via RESTful APIs. This architecture allows for efficient data exchange and ensures that the application can dynamically respond to user interactions and real-time updates. By integrating these advanced frameworks and technologies, our solution provides a comprehensive and effective tool for optimizing appointment scheduling and resource allocation in transportation networks, ultimately enhancing operational efficiency and reducing costs.

The database has been populated with all streets, localities, and counties in Romania using a public governmental data source [20]. Subsequently, we constructed a transportation network that spans the entire country, encompassing 5 regional logistics centres and 55 stores, each dynamically linked to an address. The transportation network is inspired by the distribution network of Dedeman stores [21]. Therefore, the variables from the mathematical model Section, have the following values: a = 60, b = 5. Major components of the applications are presented in Figure 4.

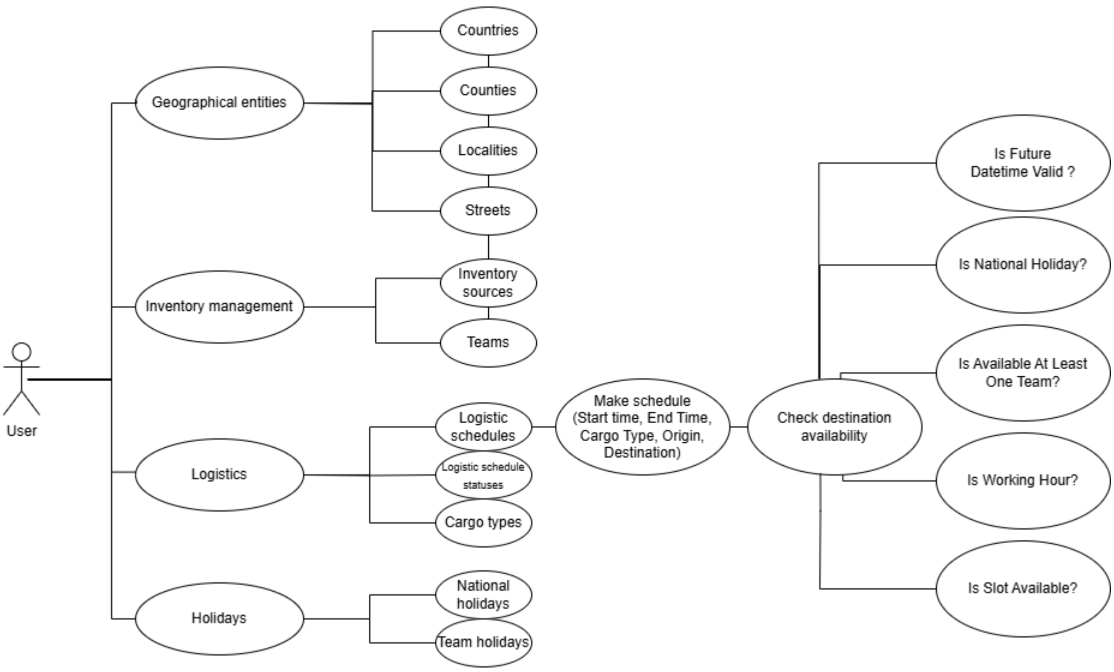

Figure 4 - Major components of proposed framework

The relationships between entities in the database are established using foreign keys to ensure dynamic connections and to enforce constraints that allow only valid values to be inserted. We have defined the LogisticSchedule entity, which serves as the main implementation of our problem. This entity is linked to the Status entity, as schedules can be in various processing states. It is also connected to the CargoType entity, where custom processing times are defined for each type of cargo. The LogisticSchedule entity is further related to the InventorySource entity, which defines the working schedule for weekdays and weekends, and is associated with teams (InventorySourceTeam) that may have specified holidays (TeamHoliday). Moreover, there is an entity for national holidays provided by the government. All these relationships and entities in the database will help us validate when a new logistic schedule can be recorded in a specific slot, that is, for a certain logistics point, on a particular day, at a specific hour, and for a defined type of cargo.

As discussed, the core implementation of our solution revolves around the LogisticSchedule entity, which will store all scheduled plans for a logistics point, only after all validations defined mathematically in the previous section have been verified. The proposed solution is based on a validation function that checks the availability status for scheduling a logistic appointment within a certain time interval, on a specific day, at a particular logistics point (the destination), which has assigned teams that may or may not have holidays.

These teams can also be affected by national holidays. The scheduling is influenced by the type of cargo (considering that each specific type of cargo has different processing times and thus non-uniform slot occupancy), as well as the existence of other schedules with the condition that they are not in the final status. Additionally, it is impacted by the working hours of the store and the validity of the time moment.

As mentioned, we have developed a function that validates the eligibility of creating a new schedule. This function is depicted in Figure 5. It returns a Boolean value. This function in turn calls five different singular functions, each of which also returns a Boolean value. We will present each of these functions immediately to highlight their significance and importance. If, in the end, each of these functions returns an appropriate result, then our main function will return true, indicating that we can record the schedule for the specified parameters at this time.

```
    Throws: NonUniqueResultException
            Exception

3 usages   cezar.papara +1
public function canMakeSchedule(string $day, string $hour, string $processingTime, InventorySource $inventorySource): bool
{
    if ($this->isFutureDateTimeValid($day, $hour) === false) {
        return false;
    }

    if ($this->isNationalHoliday($day) === true) {
        return false;
    }

    if ($this->isAvailableAtLeastOneTeam($day, $inventorySource) === false) {
        return false;
    }

    if ($this->isWorkingHour($day, $hour, $processingTime, $inventorySource) === false) {
        return false;
    }

    if ($this->isSlotAvailable($day, $hour, $processingTime, $inventorySource) === false) {
        return false;
    }

    return true;
}
```

Figure 5 - Validation function for presented solution of Logistic Scheduling Interval Problem

The **Function isFutureDateTimeValid** ensures that the scheduled date and time for the logistics appointment are set in the future. It validates that the provided date and time are not in the past by comparing them with the current date and time. This is essential to avoid scheduling appointments for times that have already passed.

The **Function isNationalHoliday** checks if the proposed scheduling date falls on a national holiday. This validation is crucial because national holidays typically affect the availability of logistics services. The function queries the database to determine if the date lies within the range of any national holiday period. If it does, scheduling on that day is not permitted.

The **Function isAvailableAtLeastOneTeam** ensures that at least one logistics team is available on the specified date at the given logistics point. It checks the availability of teams by considering their assigned holidays and other constraints. This function is fundamental to guarantee that there is sufficient human resource capacity to handle the logistics operations on the chosen date.

The **Function isWorkingHour** verifies that the scheduling falls within the working hours of the specified logistics point. It retrieves the working schedule for the day (weekdays or weekends) and confirms that the logistics appointment can be accommodated within these hours. This ensures that the logistics operations are planned within the operational hours, avoiding conflicts with non-working periods.

The **Function isSlotAvailable** checks the availability of the time slot for the new logistics appointment. It evaluates whether the proposed time slot overlaps with any existing schedules that are still active and not finalized. This function is crucial for preventing double-booking and ensuring that each time slot is dedicated to a single logistics operation, maintaining an organized and conflict-free scheduling system.

By systematically applying these validation functions, the canMakeSchedule method ensures that all logistical appointments adhere to defined constraints and are feasible within the operational framework. Each function plays a vital role in maintaining the integrity and efficiency of the scheduling system.

5 Results Assessment

This section evaluates the performance and effectiveness of our application in solving logistics scheduling problems. We assess how well the solution adheres to defined constraints and improves the efficiency and reliability of the transportation network.

Validation of Theoretical Models: We validated our theoretical models against real-world logistics scenarios by constructing a graph representing logistics centers and stores, incorporating constraints such as working hours, team availability, and cargo types. This approach, using real data, confirmed the accuracy and practicality of our models.

Performance of the Scheduling Algorithm: The scheduling algorithm was tested across various logistics scenarios. Key performance metrics include:
Accuracy: The algorithm consistently generated valid schedules, respecting all constraints.
Efficiency: It handled numerous scheduling requests promptly, suitable for high-demand environments.

Scalability and Performance Analysis: The application demonstrated exceptional scalability and performance, maintaining reliability even as the complexity of the logistics network increased. Initially tested on a transport network with 60 nodes, the system was further validated experimentally with up to 500 nodes, representing logistics centers and stores across Romania. Under these expanded conditions, it continued to perform real-time validation for creating new schedules while accounting for existing ones.

Performance metrics confirm the system's efficiency, with response times consistently under 10 seconds, even when handling up to 50 simultaneous user requests generated using automated testing tools. Frontend AJAX requests ensured seamless real-time validation, effectively preventing invalid data entries.

Practical Applicability: Our solution was assessed for practical applicability in real-world logistics operations. It effectively integrated diverse constraints, including team holidays and national holidays, demonstrating adaptability across different regions and industries. Real-time validation functions ensured schedules remained feasible and up to date.

Analysis with Numerical Data: To illustrate, we consider the existing schedules for destination Store SRL Alba Iulia from Figure 6.

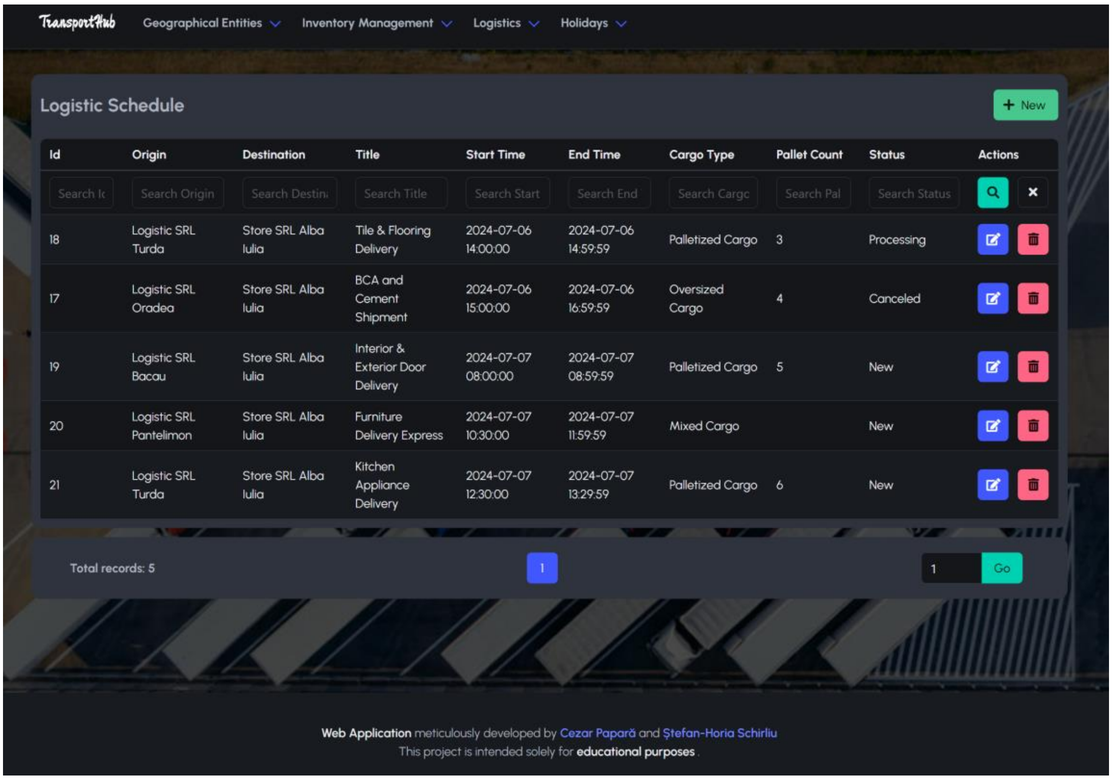

Figure 6 - Example of existing schedules for specific Logistic Point

On July 6, 2024, at 13:55:00, we attempted to schedule deliveries to destination Store SRL Alba Iulia with various start times on different days and hours, considering the cargo type as Oversized Cargo, which requires a processing time of 2 hours. The table below presents the scheduling attempts and their outcomes. In the column "Available Time Slot" we noted the system's response: "TRUE" if the scheduling was possible and "FALSE" otherwise, accompanied by the reason for the invalid scheduling.

The operating schedule for the Alba Iulia store is Monday to Friday from 08:00 to 18:00, Saturday from 08:00 to 17:00, and Sunday from 08:00 to 16:00. Additionally, according to the national holidays table, November 30, 2024, is Saint Andrew's Day. Furthermore, a vacation period for the Alba Iulia team has been set from August 15 to August 20, 2024. Table 1 presents the attempts and outcomes.

Table 1 - Validation of different schedule attempts

Nr.	Start Time	End Time	Cargo Type	Available Time Slot	Reason
1	2024-07-06 08:00:00	2024-07-06 09:59:59	Oversized Cargo	FALSE	Invalid moment (past moment)
2	2024-07-06 13:55:00	2024-07-06 15:54:59	Oversized Cargo	FALSE	Invalid moment (past moment)
3	2024-07-06 14:00:00	2024-07-06 15:59:59	Oversized Cargo	FALSE	Slot unavailable, overlapping with another active schedule
4	2024-07-06 15:00:00	2024-07-06 16:59:59	Oversized Cargo	TRUE	-
5	2024-07-06 16:00:00	2024-07-06 17:59:59	Oversized Cargo	FALSE	Slot unavailable, outside working hours
6	2024-07-07 08:30:00	2024-07-07 10:29:59	Oversized Cargo	FALSE	Slot unavailable, overlapping with another active schedule
7	2024-07-07 13:30:00	2024-07-07 15:29:59	Oversized Cargo	TRUE	-
8	2024-07-07 13:45:00	2024-07-07 15:44:59	Oversized Cargo	TRUE	-
9	2024-07-07 14:00:00	2024-07-07 15:59:59	Oversized Cargo	TRUE	-
10	2024-07-06 14:15:00	2024-07-06 16:14:59	Oversized Cargo	FALSE	Slot unavailable, end time will be outside working hours
11	Any time between 2024-08-15 and 2024-08-20	Any start time + processing time	Oversized Cargo	FALSE	Team is on holiday
12	2024-11-30 (any time)	Any start time + processing time	Oversized Cargo	FALSE	National holiday (Saint Andrew's Day)

6 Discussion: Comparison Between Our Method and TSP with Time Windows

Advantages of Our Interval Scheduling Problem (ISP) Approach with Custom Constraints:

- **Dynamic Constraints Handling**: Our method efficiently integrates various real-world constraints, such as working hours, holidays, overlapping operations, and merchandise types. These are highly relevant to logistics nodes and are checked in real-time, ensuring that the availability of time slots is accurately reflected.
- **Customized to Logistic Needs**: By considering node-specific factors, such as team holidays and merchandise types, our solution is more tailored to the internal operations of logistic points compared to generic TSP with time windows. This allows for more precise planning and resource allocation within nodes.

- **Real-Time System Monitoring**: Our approach continuously verifies that time slots remain valid, preventing issues caused by scheduling into past time slots. This real-time verification offers a higher level of accuracy and responsiveness than traditional TSP models.

Disadvantages or Challenges:

- **Complexity**: The introduction of multiple custom constraints increases computational complexity, especially when compared to TSP with time windows, which typically only accounts for a few constraints, such as travel times and delivery windows.
- **Scalability**: Handling dynamic constraints in real-time, such as continuously monitoring holiday schedules or overlapping operations, may reduce scalability for larger networks. TSP with time windows is a more established solution that tends to scale better with larger networks due to its simplicity.

Comparison with TSP with Time Windows:

- **Flexibility**: While TSP with time windows primarily focuses on optimizing routes and delivery schedules within fixed windows, it does not account for the internal dynamics of logistics nodes, such as staff availability or different processing times for goods. Our ISP-based approach is more flexible in adapting to these internal constraints.
- **Real-Time Decision Making**: TSP with time windows often works with pre-set schedules, whereas our method incorporates real-time availability checks, offering more dynamic decision-making capabilities. This can improve efficiency in unpredictable environments but comes at the cost of increased system complexity.
- **Scope**: TSP with time windows is primarily focused on external routing optimization. In contrast, our approach shifts focus to internal node operations, making it more suitable for environments where bottlenecks within nodes, like warehouses, are a significant factor in overall performance.

7 Conclusions

In conclusion, our analysis of the logistic interval scheduling problem highlights that the more detailed and real-world constraints we incorporate into our transportation network, the more complex the problem becomes. However, this increased complexity results in more adaptable and realistic solutions. By integrating custom constraints such as working hours, holidays, overlapping operations, and merchandise-specific processing times, our method creates solutions that are not only theoretically sound but also highly practical and robust in real-world logistics.

Comparing our approach to the traditional Traveling Salesman Problem (TSP) with time windows, we observe that while TSP provides simpler, scalable solutions focused on external routing, it does not address the internal logistical complexities within network nodes. Our Interval Scheduling Problem (ISP)-based approach, on the other hand, captures these internal dynamics and offers real-time adaptability, though at the cost of increased computational overhead.

Looking ahead, we plan to explore further the connections between TSP, Vehicle Routing Problem (VRP), and Logistic Interval Scheduling. Investigating how logistical constraints—such as capacity, resource availability, and time limitations—affect the selection of optimal routes will be central to our future research. We aim to develop heuristic solutions that account for all relevant constraints and variables, ensuring these theoretical ideas can be applied in practice. This work will focus on addressing the inherent uncertainty and complexity in transportation networks, with the goal of creating dynamic and scalable solutions for modern logistics.

Acknowledgment
The research performed Ștefan-Horia Schirliu was funded by the Ministry of Education, through the National Council for the Financing of Higher Education, Romania, grant number CNFIS-FDI-2024-F-0492: Integrated actions to increase the quality of the research process at "Vasile Alecsandri" University of Bacau (acronym: UBc4Research).

References

[1] Wang, Yu & Liu, Haoxiang & Fan, Yinchao & Ding, Jianxun & Long, Jiancheng. *Large-scale multimodal transportation network models and algorithms-Part II: Network capacity and network design problem*, Transportation Research Part E: Logistics and Transportation Review, 167, 102918, Pages 1-7, 2022, 10.1016/j.tre.2022.102918

[2] Lumír Pečený, Pavol Meško, Rudolf Kampf, Jozef Gašparík. *Optimisation in Transport and Logistic Processes*, Transportation Research Procedia, Volume 44, Pages 15-22, ISSN 2352-1465, 2020

[3] James R. Jackson. *Scheduling a Production Line To Minimize Maximum Tardiness*, Research Report, Issue 43, Pages 1-72, Management Science Research Project, University of California, Los Angeles, 1955

[4] Min Ji, Ji Fang, Wenya Zhang, Lijuan Liao, T.C.E. Cheng, Yuanyuan Tan. *Logistics scheduling to minimize the sum of total weighted inventory cost and transport cost*, Computers & Industrial Engineering, Volume 120, Pages 206-215, ISSN 0360-8352, 2018

[5] Mallek, Amin & Boudhar, Mourad. *Scheduling on uniform machines with a conflict graph: complexity and resolution*, International Transactions in Operational Research. 31, Pages 1-4, 2024, 10.1111/itor.13170

[6] Allahverd, Ali. *A survey of scheduling problems with uncertain interval/bounded processing/setup times*, Journal of Project Management, Volume 7, Pages 255-264, 2022, 10.5267/j.jpm.2022.3.003

[7] Niraj Ramesh, D., Krishnamoorthy, M., Ernst, A.T. *Efficient models, formulations and algorithms for some variants of fixed interval scheduling problems*, in: Sarker, R., Abbass, H.A., Dunstall, S., Kilby, P., Davis, R., Young, L. (eds.) Data and Decision Sciences in Action. LNMIE, Pages 43–69, Springer, Cham, 2018, https://doi.org/10.1007/978-3-319-55914-8_4

[8] Boyar, J., Favrholdt, L.M., Kamali, S., Larsen, K.S. *Online interval scheduling with predictions*, in: 18th International symposium on algorithms and data structures (WADS), Lecture Notes in Computer Science, Volume 14079, Pages 193–207, Springer, Berlin, Heidelberg, 2023

[9] Xu, Xinping & Zhang, Jingwen & Li, Minming & Duan, Lingjie & Xie, Lihua. *A Family of Strategyproof Mechanisms for Activity Scheduling **, Autonomous Agents and Multi-Agent Systems, Volume 37, 2, Pages 1-8, 2023, 10.1007/s10458-023-09624-7

[10] Zhou, Haohao & Bai, Guanghan & Deng, Su. *Optimal interval scheduling with nonidentical given machines*, Cluster Computing, Volume 22, 5, Pages 1-6, 2019, 10.1007/s10586-018-02892-z

[11] Gogos, Christos. *Solving the Distributed Permutation Flow-Shop Scheduling Problem Using Constrained Programming*, Applied Sciences, Volume 13, 12562, Pages 1-26, 2023, 10.3390/app132312562

[12] Kolker, Alexander. *Staffing and Scheduling Problems with Sliding Days Off: Solutions Using Integer Linear Optimization*, in book: Healthcare Management Engineering In Action, Pages 1-15, 2024, 10.1007/978-3-031-53663-2_6

[13] A. Sri Harsha, P. Vijaya Kumar, Arumalla Nagaraju, K. Prakash Babu, N. R. Medikondu, B. V. Dharmendra. *Mining knowledge for NP-complete scheduling problems*, AIP Conference Proceedings, At: Nandyal City, Volume 2821 (1), Pages 1-5, 070014, 2023, https://doi.org/10.1063/5.0158474

[14] G. Reinelt. *The Traveling Salesman Problem: Computational Solution for TSP Applications*, LNCS, Springer-Verlag, vol. 840, Pages 4-30, 1994

[15]G. B. Dantzig and J. H. Ramser. *The Truck Dispatching Problem*, Management Science, Vol. 6, No. 1, Pages 80-91, 1959
[16]Cezar-Marian Papară. *An empirical study of large transportation networks and solutions for the cost optimization*, International Journal of Advanced Statistics and IT&C for Economics and Life Sciences, Vol. XII, no. 2, Pages 41-52, 2022
[17]Cezar-Marian Papară. *Enhancing Education Quality in Rural Settings through Steiner-Based Societal Partitioning Optimization*, Journal of Innovation in Psychology, Education and Didactics ISSN: 2247-4579, E-ISSN: 2392-7127, 2023, Volume 27, Issue 2, Pages: 241-248, 2023, doi:10.29081/JIPED.2023.27.2.08
[18]Cezar-Marian Papară. *Digital frontiers in logistics: A scalable approach to wide-area transportation network optimization*, "Vasile Alecsandri" University of Bacău, Faculty of Sciences, Series Mathematics and Informatics, Vol. 33, No 2, Pages 63 – 80, 2024
[19]Cezar-Marian Papară, Alex-Andrei Rîpan. *Strategies for Optimizing Pharmaceutical Logistics: A Focus on Transportation Efficiency*, Eighth International Conference on Applied Informatics, Imagination, Creativity, Design, Development - ICDD, Sibiu, Romania, https://conferences.ulbsibiu.ro/icdd/2024/program.php, 2024
[20][Online] https://data.gov.ro/dataset?tags=nomenclator, [Accessed 01.04.2024]
[21][Online] https://www.dedeman.ro/ro/suport-clienti/magazine-dedeman, [Accessed 03.04.2024]