



Scheduling in an Emergency Department, Linear Formulation and Heuristic Approach

Lahcene Mezouari, Lucas Wicher, Jean-Paul Boufflet, Aziz Moukrim

► To cite this version:

Lahcene Mezouari, Lucas Wicher, Jean-Paul Boufflet, Aziz Moukrim. Scheduling in an Emergency Department, Linear Formulation and Heuristic Approach. 9th International Conference on Control, Decision and Information Technologies (CoDIT 2023), Jul 2023, Rome, France. pp.1547-1552, 10.1109/CoDIT58514.2023.10284409 . hal-04365844

HAL Id: hal-04365844

<https://hal.science/hal-04365844>

Submitted on 28 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Scheduling in an Emergency Department, linear formulation and heuristic approach *

Lahcene Mezouari, Lucas Wicher, Jean-Paul Boufflet and Aziz Moukrim
Université de technologie de Compiègne, CNRS, Heudiasyc (Heuristics and Diagnosis of Complex Systems) CS 60 319 - 60 203 Compiègne Cedex

December 27, 2023

Abstract

In this study, we investigate the real-world scheduling problem of the Medical Day Unit of the Emergency Department (MDU-ED) of Jeanne de Flandres University Hospital in Lille (France). We implemented a heuristic (PRH) based on hospital practitioners' rules we collected by observing the operation of MDU-ED. We propose an Integer Linear Programming (ILP) formulation that makes it feasible to solve small instances. We propose an Adaptive Iterative Destruction Construction Heuristic (IDCH) solution approach. The IDCH obtains better solutions than the PRH within reasonable processing times. We report on experiments performed on instances generated using real-world patient pathways of the MDU-ED.

1 Introduction

Emergency departments (ED) are currently subject to an increasing demand. Optimizing processes in the ED may help in facing this increase by better use of resources. Improving the ED functioning can be approached from different views such as

staffing problems, rostering problems, task scheduling problems and several approaches have been investigated to address the optimization of such processes [1, 2, 3, 4, 5, 6, 7].

There are relatively few studies in the literature dealing with task scheduling problems in an emergency department. Kiriş et al. [8] presented a model for a real-time response. The proposed approach is based on a parallel machine scheduling model. The authors consider stochastic patient arrivals and deterministic care task processing times. Patients are grouped into six categories in the triage phase. A constructive heuristic is proposed to assign patients to doctors based on arrival time, patient priority, stay time, and doctor charge. The heuristic dynamically updates the assignment of patients to doctors as new patients are checked into triage. Ajmi et al. [9] proposed an approach for patient assignment using dynamic planning and considering treatment times to be stochastic. The scheduling problem is solved by a genetic algorithm. Azdazeh et al. [10] found that the timing of follow-up analyses in emergency labs affects the length of stay of patients in the ED. The planning of additional patient analyzes was modeled as a flexible *open shop* problem. The objective is to minimize waiting times for analyses and a genetic algorithm is proposed.

Lee and Lee [11] proposed a deep reinforcement learning algorithm. Each patient is categorized by triage method and assigned a level of urgency. For each resource, a list of patients is defined. The learning

*This work was carried out within the framework of the Labex MS2T, which was funded by the French Government and the French National Research Agency (ANR) as part of the OIILH project (Ref. ANR-18-CE19-0019). This work also benefited from an internship grant obtained as part of the PIA3 SFRI Real SU project of the Sorbonne University Alliance.

algorithm determines the best method of choosing patients for each list among several defined methods. The objective is to minimize the patient's stay in the ED. Patient are assigned to an appropriate medical resource, taking into consideration the level of urgency. The waiting time value may be different depending on the severity level of the patient. Luscombe and Kozan [12] proposed a combination of a parallel machine model to handle bed allocation (patient/bed) and a flexible job shop model to schedule resources (i.e. medical staff) for the patients. In this ED organization, once a patient is assigned to a bed, he/she does not move from it, the place is definitely assigned. Patients were seen as jobs and their care tasks (e.g. medical assessment) as operations that were processed by the resources.

In these previous studies, one can observe that ED operating rules differ from one country to another. It depends also on the organization of each ED which is itself immersed in the general organization of the hospital. Therefore, no proposed solution can be directly used and the operation of the service must be observed first and then modelled.

In the MDU-ED the patient care tasks may require to be performed at different places inside or outside the MDU-ED. From their admission until their departure, the patients' pathways at MDU-ED can be viewed as a continuous succession of places. Care tasks are performed, for example, in a box inside the MDU-ED or in the radiology department outside the MDU-ED. Waiting tasks are performed in a dedicated room, but also in places where care tasks are performed. This involves that for a patient there can be no break in the succession of places the patient occupies.

The processing time of a care task is less than or equal to the occupancy time of its allocated place. This should be minimized. The patient can also wait in the waiting room of the MDU-ED (inside the MDU-ED). No medical resource is allocated, but we have to respect the waiting room's capacity. For a patient, the period of this allocation is unknown beforehand as well and should be minimized.

We observed the operation of the MDU and its interaction with the other departments of the hospital. In collaboration with emergency physicians, we

have determined a set of the most common patient pathways, which detail the sequences of tasks. We have developed a heuristic based on observed hospital practitioners' rules, which also serves as a basis for comparison.

The operation of an ED is a dynamic process (admissions and departure of patients). It is however necessary to first consider the static case for modeling purposes and to obtain reference solutions which will serve as a basis for comparison. We propose an ILP model for this purpose. An ILP solving approach may face difficulties in computing solutions in times compatible with the necessary reactivity in the dynamic case. In this study, we propose an IDCH-type heuristic approach and configure it to obtain a good trade-off between quality of results and computation time.

2 Problem description

On arrival at the MDU-ED, a patient is first ranked based on the degree of urgency from among the set {1: medical resuscitation, 2: high level of urgency, 3: urgent, 4: less urgent, 5: non-urgent}. An acceptable timeframe for the first medical assessment performed by a doctor corresponds to each level of urgency, from zero for the first (immediate care) to two hours for the last. The MDU-ED daily takes care (from 8 a.m. to 9 p.m.) of patients with level of urgency 3, 4 and 5.

From their admission until their departure, every patient is always in a place that can either be inside or outside the MDU-ED area (e.g. the radiology department). Each human resource at MDU-ED has specific skills related to the tasks they can perform. Medical staff consists of an emergency doctor, a medical intern, a traumatology intern, a dentistry intern, two nurses, four caregivers and two hostesses. Each of them is a unit resource. Inside MDU-ED area, the medical boxes and the waiting room are the two types of places. They are four consultation boxes, a suture box, a plaster box, an Ear-Nose-Throat (ENT) box and a dentistry box. Each of them is a unit resource. The waiting room has a limited capacity. We also consider a place of infinite capacity for the moving

of patients (transition). The MDU-ED neither has control on the other medical services it needs for certain medical tasks, nor has it on the associated places it does not manage. Thus, external services can be considered as black boxes and regrouped in one place of infinite capacity. The external medical staff can as well be considered as one resource that can be shared between several patients (i.e. free at all times).

A patient's pathway can be represented as a directed acyclic graph (DAG) where vertices correspond to tasks and directed edges to precedence constraints between tasks. The pathways are the same until the first medical consultation, next they vary.

As an illustrative example, a schedule of two patient pathways is depicted on Fig. (1). There are a Gantt chart for places and a Gantt chart for medical staff. The Gantt charts show resource occupancy, not the individual tasks. As can be shown on the Gantt chart for the places, there is no idle time between the places for a patient pathway. For each patient, there is a continuous and successive occupation of places.

There are three types of tasks: care tasks, moving tasks and waiting tasks. Care tasks use some health staff resources. Most of them require the patient's attendance and require an appropriate place that fits with the task, others do not. Care tasks can be performed inside or outside the MDU-ED. A staff member is required to move a patient from one place to another. Waiting tasks are used to model the waiting times of a patient in the dedicated waiting room of the MDU-ED. They do not require medical resources.

Care tasks can be performed sequentially in a same place or not. For example, consider a medical consultation followed by a blood sampling. On the Gantt chart of medical staff, a medical consultation (performed by "ResidentEnt1") ends at 08:25, then it is followed by a blood sampling (performed by "Nurse2") that begins at 08:25. In such a case, we must ensure that the same place is allocated to both because there is no moving tasks between them (for this patient pathway, the 'ORLBox' was used).

For tasks that would require multiple resources in the real world (e.g. a doctor and a nurse) we decomposed them into multiple unit tasks (one for each resource). Hence, these tasks must start at the same time but can have different lengths. On the Gantt

chart for medical staff, see resources "ResidentTraumatolo1" and "Nurse2" (08h10).

Each care task and each moving task has a non-null length. There are explicit waiting tasks that can be modelled in a patient pathway. Each of them has an unknown length to be minimized when computing the schedule (minimum zero).

However, this is not sufficient to take into account hidden waiting times that occur in real situation. For the sake of explanation, consider a single care task that needs a human resource and this care task needs to be performed in a box. Assume the patient be moved from one place to the box with the aim of performing the care task. The human resource may or may not be available when the patient arrives in the box. In this later case, the patient waits inside the box before the arrival of the resource. Consider next, that the care task is performed, so the patient needs to be moved from the box to the waiting room, this requires another human resource that may or may not be available. In this later case, the patient waits inside the box after the departure of the resource.

Each care task with the patient's attendance has a length and needs an appropriate place. Hence, the length of occupancy of the place must be greater than or equal to this length. Length of a task must be equal to length of a place occupancy in the following cases.

For the tasks performed outside the MDU-ED with the patient's attendance, MDU-ED has no control and the processing time of such tasks has been estimated to include waiting periods that occur outside the MDU-ED. For the tasks that do not need the patient's attendance because the patient is not present for these tasks.

The objective is to shorten the waiting periods by taking into account the priority of each patient so as to minimize the Length of Stay (LOS) and also to shorten the Door-to-Doctor-Time (DTDT) waiting time while considering the place capacities and all the other medical resources.

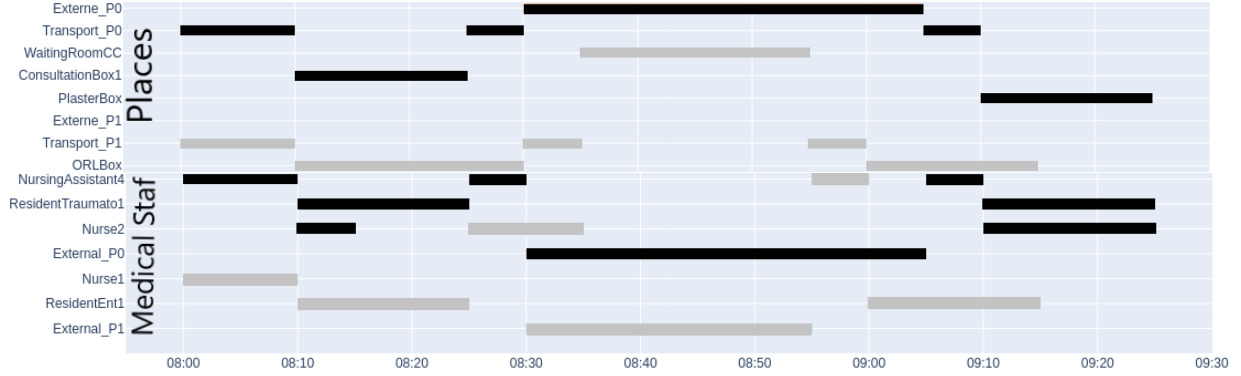


Figure 1: A schedule of the tasks of two patient pathways, places usage on top, medical staff resources used on bottom

3 ILP Formulation

The planning horizon is composed of T periods t of five minutes each. We denote the set of patients as J . For a patient, j the set of tasks which require their attendance is denoted as $I_j = I_j^{in} \cup I_j^{ou}$, where I_j^{in} is the set of tasks that are performed inside the MDU-ED whereas I_j^{ou} is the set of tasks that are performed outside the MDU-ED. We denote similarly, the set of waiting tasks as $W_j = W_j^{in} \cup W_j^{ou}$, the set of care task as $C_j = C_j^{in} \cup C_j^{ou}$ and the set of moving tasks as $D_j = D_j^{in} \cup D_j^{ou}$. Hence, we also have $I_j = W_j \cup C_j \cup D_j$. The set of tasks that do not require the j patient attendance is denoted as E_j . We denote the set of places as L and the set of types of resources as G . The number of resources of type g is denoted as R_g . We specifically denote as g_{ext} the type corresponding to external resources.

Given a task i of a patient j , we denote the set of places where i can be performed as L_{ij} and the set of types of resource that can perform i as G_{ij} . The set of precedence tasks of i is denoted Γ_{ij}^{-1} , and the set of successor tasks as S_{ij} . Some tasks may require to be performed in the same place as i and others may need to start at the same time as i . Hence, we denote the set of tasks that should be performed in the same place as i as Λ_{ij} , and the set of tasks that need to start at the same time as i as Ω_{ij} .

The primary decision variables are as follows:

x_{ijlt} one if patient j enters in place l to perform task i at time t , zero otherwise ;

z_{ijgrt} one if resource r of type g is used to begin task i of patient j at time t , zero otherwise.

Excluding waiting tasks, each task is processed within the occupancy time of its allocated place and needs a resource to be performed. The beginning of the occupancy time is expressed by the x_{ijlt} decision variables and the beginning of the processing of a task i by the z_{ijgrt} decision variables.

$u_{ijj'j'l}$ one if place l is used to perform task i of patient j before patient j' for task i' , zero otherwise ;

δ_{ij} one if the length of task i of patient j in a place is greater than zero, zero otherwise.

The $u_{ijj'j'l}$ variables are used to manage the disjunction of the place usages. For a patient j , the occupancy time of waiting tasks W_j are to be minimized. The δ_{ij} variables are used to manage the following case: for a patient j , a waiting task i of null time may use any place regardless of its occupation by other patients. Thus, the patient enters a place and immediately leaves it.

The problem can be formulated as the following integer linear program.

$$\begin{aligned}
& \text{Min} \\
& \sum_{j \in J} \left(\sum_{t=1}^T \sum_{l \in L_{\varphi_j j}} t \cdot x_{\varphi_j j l t} - \sum_{t=1}^T \sum_{l \in L_{1j}} t \cdot x_{1j l t} \right) \quad (1) \\
& + \sum_{j \in J} \omega_j \left(\sum_{t=1}^T \sum_{g \in G_{d_j j}} \sum_{r=1}^{R_g} t \cdot z_{d_j j r t} - a_j \right) \quad (2) \\
& \text{s.t.} \\
& x_{ij l t}, z_{ij g r t}, u_{ij i' j' l}, \delta_{ij} \in \{0, 1\} \\
& \sum_{t=\beta_{1j}}^{\gamma_{1j}} \sum_{l \in L_{1j}} (t \cdot x_{1j l t}) = a_j \quad \forall j \in J \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} x_{ij l t} = 1 \quad \forall j \in J, \forall i \in I_j \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{g \in G_{ij}} \sum_{r=1}^{R_g} z_{ij g r t} = 1 \quad \forall j \in J, \forall i \in D_j \cup C_j \cup E_j \\
& \sum_{j \in J} \sum_{i \in D_j \cup C_j \cup E_j} \sum_{\tau=\max\{\beta_{ij}, t-p_{ij}+1\}}^{\min\{\gamma_{ij}, t\}} z_{ij g r \tau} \leq 1 \\
& \forall g \in G \setminus \{g_{ext}\}, \forall r \in \{1, \dots, R_g\}, \forall t \in \{1, \dots, T\} \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) - \sum_{t=\beta_{i'j}}^{\gamma_{i'j}} \sum_{l \in L_{i'j}} (t \cdot x_{i'j l t}) = 0 \\
& \forall j \in J, \forall i \in I_j, \forall i' \in \Lambda_{ij} \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{g \in G_{ij}} \sum_{r=1}^{R_g} (t \cdot z_{ij g r t}) - \sum_{t=\beta_{i'j}}^{\gamma_{i'j}} \sum_{g \in G_{i'j}} \sum_{r=1}^{R_g} (t \cdot z_{i'j g r t}) = 0 \\
& \forall j \in J, \forall i \in C_j, \forall i' \in \Omega_{ij} \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) - \sum_{t=\beta_{i'j}}^{\gamma_{i'j}} \sum_{l \in L_{i'j}} (t \cdot x_{i'j l t}) = 0 \\
& \forall j \in J, \forall i \in I_j, \forall i' \in \Omega_{ij} \\
& \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) - \sum_{t \in L_{i'j}} (t \cdot x_{i'j l t}) = 0 \\
& \forall t \in \{\max(\beta_{ij}, \beta_{i'j}), \dots, \min(\gamma_{ij}, \gamma_{i'j})\}, \\
& \forall j \in J, \forall i \in I_j, \forall i' \in \Omega_{ij} \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) - \sum_{t=\beta_{i'j}}^{\gamma_{i'j}} \sum_{l \in L_{i'j}} (t \cdot x_{i'j l t}) \geq 0 \\
& \forall j \in J, \forall i \in I_j, \forall i' \in \Gamma_{ij}^{-1} \cap W_j \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) - \sum_{t=\beta_{i'j}}^{\gamma_{i'j}} \sum_{g \in G_{i'j}} \sum_{r=1}^{R_g} (t \cdot z_{i'j g r t}) \geq p_{i'j} \\
& \forall j, \forall i \in I_j, \forall i' \in \Gamma_{ij}^{-1} \cap (C_j^{in} \cup E_j) \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) - \sum_{t=\beta_{i'j}}^{\gamma_{i'j}} \sum_{g \in G_{i'j}} \sum_{r=1}^{R_g} (t \cdot z_{i'j g r t}) = p_{i'j} \\
& \forall j \in J, \forall i \in I_j, \forall i' \in \Gamma_{ij}^{-1} \cap (D_j \cup C_j^{ou}) \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{g \in G_{ij}} \sum_{r=1}^{R_g} (t \cdot z_{ij g r t}) - \sum_{t=\beta_{i'j}}^{\gamma_{i'j}} \sum_{g \in G_{i'j}} \sum_{r=1}^{R_g} (t \cdot z_{i'j g r t}) \geq p_{i'j} \\
& \forall j \in J, \forall i \in E_j, \forall i' \in \Gamma_{i'j}^{-1} \cap (C_j \cup D_j \cup E_j) \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{g \in G_{ij}} \sum_{r=1}^{R_g} (t \cdot z_{ij g r t}) - \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) \geq 0 \\
& \forall j \in J, \forall i \in C_j^{in} \\
& \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{g \in G_{ij}} \sum_{r=1}^{R_g} (t \cdot z_{ij g r t}) - \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) = 0 \\
& \forall j \in J, \forall i \in D_j \cup C_j^{ou} \\
& \sum_{t=\beta_{sj}}^{\gamma_{sj}} \sum_{l \in L_{sj}} (t \cdot x_{sj l t}) - \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) \geq \delta_{ij} \\
& \forall j \in J, \forall i \in I_j, s \in S_{ij} \cap I_j \\
& \sum_{t=\beta_{sj}}^{\gamma_{sj}} \sum_{l \in L_{sj}} (t \cdot x_{sj l t}) - \sum_{t=\beta_{ij}}^{\gamma_{ij}} \sum_{l \in L_{ij}} (t \cdot x_{ij l t}) \leq \delta_{ij} \cdot D \\
& \forall j \in J, \forall i \in I_j, s \in S_{ij} \cap I_j \\
& u_{ij i' j' l} + u_{i' j' i j l} \geq \sum_{t=\beta_{ij}}^{\gamma_{ij}} x_{ij l t} + \sum_{t=\beta_{i'j'}}^{\gamma_{i'j'}} x_{i' j' l t} + \delta_{ij} + \delta_{i' j'} - 3 \\
& \forall j \in J, \forall i \in I_j^{in}, \forall j' \in J \setminus \{j\}, \\
& \forall i' \in I_{j'}^{in}, \forall l \in L_{ij} \cap L_{i' j'} \\
& u_{ij i' j' l} + u_{i' j' i j l} \leq 1 \\
& \forall j \in J, \forall i \in I_j^{in}, \forall j' \in J \setminus \{j\},
\end{aligned}$$

$$\forall i' \in I_{j'}^{in}, \forall l \in L_{ij} \cap L_{i'l'}$$

$$(21) \quad 4 \quad \text{PRH and IDCH heuristics}$$

$$\sum_{t=\beta_{i'l'}}^{\gamma_{i'l'}} (t \cdot x_{i'l't}) - \sum_{t=\beta_{sj}}^{\gamma_{sj}} \sum_{l \in L_{sj}} t \cdot x_{sjlt} \geq (u_{ij i' j' l'} - 1) \cdot M$$

$$\begin{aligned} &\forall j \in J, \forall i \in I_j^{in}, \forall j' \in J \setminus \{j\}, \forall i' \in I_{j'}^{in}, \\ &\forall l' \in L_{ij} \cap L_{i'l'}, s \in S_{ij} \cap I_j \end{aligned} \quad (22)$$

The objective function minimizes the LOS (1) and the DTTD (2). For a patient j , the last task is φ_j , the priority penalty that relates to the degree of urgency is ω_j , the first task requiring a doctor is d_j , and the arrival time is a_j . Constraints (3) require the variables to be either zero or one. Constraints (4) ensure that each patient j is in a place at the a_j arrival time. Constraints (5) ensure that each task is processed in one place. Constraints (6) ensure that one resource is used by the care and moving tasks. Thus, constraints (5) and (6) ensure that each task is performed only once for a patient's pathway. Except for external resources, constraints (7) ensure that a resource allocated for a task i of a patient j cannot be allocated for any other patient during the processing time of the i task.

Constraints (8) ensure that a same place is used by the tasks which need it. Constraints (9) ensure that a same start time is used by the tasks which need it.

A patient j can enter a place for a task i when all the Γ_{ij}^{-1} precedence tasks are completed. Constraints (12)-(14) ensure the precedence constraints according to each type of tasks that precedes task i (waiting, care, moving). For the tasks that do not require the patient's attendance, constraints (15) ensure the precedence relations. Constraints (16) ensure that a care task can begin after the patient is in one of the convenient places. For moving tasks, constraints (17) make sure that the patient is immediately handled by the resource. Constraints (18)-(19) ensure that a δ_{ij} is equal to zero if the i task has a null length. Constraints (20)-(22) ensure the disjunction constraints. We denote $Obj(S)$ as the function that assesses a solution S as presented in equations (1)-(2).

We spent several days in the MDU-ED to observe its operation and we discussed at length with the medical staff. From these field observations, we have seen that some hospital practitioners' rules are used to manage patients, their pathway and the care tasks within the pathway. These rules are used whatever the cases are that makes it possible to mimic the overall strategy and we denoted as PRH the constructive heuristic we implemented. Given an initial solution computed by PRH, we propose an Iterative Destruction/Construction Heuristic (IDCH) to improve the schedule. Algorithm 1 shows the IDCH proposed approach.

Algorithm 1 IDCH for MDU-ED

Input: S_{cur} computed by PRH

Parameters: $MaxIter, D_{limit}$

```

Iter ← 0; Dmax ← Dinit; Sbest ← Scur;
while Iter < MaxIter do
    k ← rand(1, Dmax);
    Ldes ← PatientRandomDestruction(Scur, k);
    PatientPriorityAndSort(Ldes);
    Scur ← BestInsertionPatient(Scur, Ldes);
    if Obj(Scur) < Obj(Sbest) then
        Sbest ← Scur; Dmax ← Dinit;
        Iter ← 0;
    else
        Dmax ← min(Dmax + 1, Dlimit);
        Iter ++;
return Sbest;

```

Provided a feasible solution S_{cur} , at each iteration, we first remove at random a number k of patients and place them into a list L_{des} . At most, D_{max} patients are removed at each overall iteration ($k \leq D_{max}$). D_{max} acts as a degree of diversification. It is incremented after each non-improving overall iteration up to D_{limit} . We make use of this adaptive diversification mechanism to explore the neighborhood of the new solution as soon as an improvement is found and to explore more distant zones by increasing D_{max} whenever the search is trapped in a local optimum. This aims at better exploring the solutions space toward a good solution. Then,

the *PatientPriorityAndSort*(\mathcal{L}_{des}) sorts \mathcal{L}_{des} , in ascending order of the patients' contribution to the objective function. Next, the patients of \mathcal{L}_{des} are scheduled using *BestInsertionPatient* to obtain a new solution S_{cur} . When a better solution is found, we update S_{best} , and we reset the iteration counter $Iter$ to zero and D_{max} to its initial value. Whenever we perform a non-improving iteration, we increment D_{max} if it has not yet reach D_{limit} . The algorithm stops when $MaxIter$ iterations have been performed without any improvement.

Algorithm 2 (BIP) shows the *BestInsertionPatient*. Given a partial solution S_{cur} , we aim at scheduling in turn the tasks i of each patient $p \in \mathcal{L}_{des}$. For the task i of a patient p , we compute $\mathcal{S}_{i,p,t6}$ the set of 6-tuplets indexed by $t6$. One tuple can be used to schedule the task i of patient p . A tuple $t6$ contains a place, a resource, an entry time in the place, an exit time from the place, a treatment start time and a treatment end time. The BIP selects the earliest feasible tuple to schedule i , next the resources and the places' availability are updated.

The directed acyclic graph that corresponds to the pathway of a patient p should be strictly respected to schedule the tasks.

We chose to experiment with an IDCH because such a destruction/construction approach will be easier to adapt in the dynamic case. It will be necessary to destroy an ongoing schedule at each occurrence of an event at time t that forces to recalculate a task scheduling in an efficient way. The computing time should be reasonable to fit with the high level of reactivity required in an emergency department.

5 Computational Experiments

In our experiments, our objectives were: (i) to provide an overview of how to adjust IDCH parameter for best performance, (ii) to compare performances between the ILP model and the IDCH approach within a 1800 seconds time limit. Tests were done using C++ and Standard Template Library (STL), compiled with GCC under Linux, on a machine with an Intel(R) Xeon(R) Gold 6138 CPU @ 2.00 GHz.

Algorithm 2 Best Insertion Patient (BIP)

Input: S_{cur} , \mathcal{L}_{des} ordered list of patients to be scheduled

```

while ( $\mathcal{L}_{des} \neq nil$ ) do
   $p \leftarrow \mathcal{L}_{des}.pop\_front()$ ;
  //  $\mathcal{L}_{des}^{(p)}$ , for a patient  $p$ , the list of tasks to be scheduled
  while ( $\mathcal{L}_{des}^{(p)} \neq nil$ ) do
    //  $\mathcal{S}_{i,p,t6}$  for task  $i$  of the patient  $p$ , set of 6-uplets
    // (place, resource, entry time in the place,
    // exit time from the place,
    // treatment start time, treatment end time)
     $Compute(\mathcal{S}_{i,p,t6})$ ;
    Schedule the task  $i$  of patient  $p$  in  $S_{cur}$  in the
    earliest possible time using  $\mathcal{S}_{i,p,t6}$ ;
    Update places and resources availability;
  return  $S_{cur}$ 

```

We used CPLEX 12.10 [13] solver.

Datasets overview

We were given at our disposal data from the ED of the Jeanne de Flandres University Hospital in Lille. The level of urgency and the code diagnostic filled in after the first medical assessment were used. We modeled the most representative patient pathways (about twenty) and fixed the lengths of the care tasks in accordance with the medical staff. We decided to generate instances based on these pathways. To generate an instance with a set of patients, we draw up patients arrivals according to a nonstationary Poisson process and the patients pathways are chosen at random.

Parameter tuning

The IDCH solution approach uses the parameter $MaxIter$, the maximum number of iterations without any improvements. This influences the quality of solutions and CPU times. To make adaptive this parameter with an instance, we formulate $MaxIter$ as $\alpha \cdot \left\lceil \frac{\sum_j |I_j|}{\max(|L|, \sum_g |R_g|)} \right\rceil$, $\alpha \in \mathbb{N}$. Given an instance, the ratio increases as the total amount of tasks and decreases as the amount of places or the amount of

type of resources for care tasks. To tune α , we performed preliminary experiments on a subset of randomly chosen instances from among the generated ones. As IDCH is a randomized search method, the experiments were repeated five times with a different random seed. We evaluate the results using the Relative Percentage Error (RPE) defined as follows: $RPE = 100 \cdot \frac{Z_{best} - Z_{min}}{Z_{best}}$. For an instance, Z_{min} is $\min(Z_{AllRuns}, Z_{ILP})$ where $Z_{AllRuns}$ is the best solution obtained on all the executions carried out during tuning and Z_{ILP} is the optimal solution attained by the ILP, if any. Then, we calculate the average RPE for the runs for the ten chosen instances for the tuning. As shown in Figure 2, the average RPE decreases as α increases and the CPU time increases with the α value. The average RPE stabilizes when α is greater than or equal to 130, we chose $\alpha = 130$, which is a good trade-off between solution quality and CPU time.

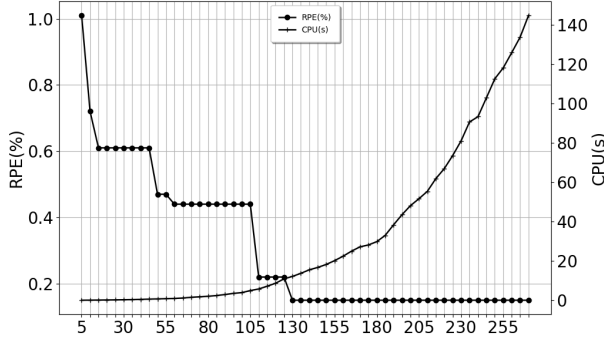


Figure 2: Tuning of α , effect of α on RPE (in %) and CPU time (in s).

Results

The instances in datasets are ranged according to the number of patients. So, instances are denoted as $a_XX(a/b/c)$, the $(a/b/c)$ notation is used whether two distinct instances have the same number of patients. In Table 1, for each instance, column $P(T)$ shows the number of patients and tasks and column PRH shows the result of the constructive heuristic that mimics the practitioners' rules. Under the ILP heading, Obj , $t(s)$ and $g\%$ columns indicate the objective function value, the computing time and the

gap recorded by the CPLEX solver. Under the IDCH heading, we show the *Min*, *Max* and *Avg* values that we obtained within the 10 runs. Column $t(s)$ shows the average processing times in second. We use $(Min - Obj)/Obj$ reported in column headings as $Ig(\%)$ to compare results of IDCH with those of ILP, if any.

The ILP approach finds optimal solutions and finds feasible solutions for some instances, but faces difficulties to obtain feasible solutions for some other within the 1800s time limit. The IDCH approach found a feasible solution for each instance within less than 30s. All the optimal solutions are obtained and strictly improves all the other solutions compared to PRH and ILP.

Instance	P (T)	PRH	ILP			IDCH				
			Obj	t(s)	g(%)	Min	Max	Avg	Ig(%)	t(s)
u_06	6(97)	2505	2615	1800	27.18	2175	2430	2291	-16.83	18.57
u_07*	7 (91)	1955	1850	16.41	0.00	1850	1850	1850	0.00	0.28
u_08*	8 (82)	2380	1940	183.38	0.00	1940	1950	1945	0.00	0.51
u_09*	9 (92)	6500	3205	526.08	0.00	3205	3205	3205	0.00	0.81
u_10(a)	10 (87)	3625	2835	379.80	0.00	2835	2835	2835	0.00	0.36
u_10(b)	10 (155)	20560	—	1800	—	7715	8030	7855	—	23.59
u_10(c)*	10 (76)	4385	4085	184.22	0.00	4085	4085	4085	0.00	0.28
u_11(a)	11 (132)	6260	—	1800	—	3800	4080	3911	—	2.49
u_11(b)	11 (100)	7275	7360	1800	1915.76	6020	6350	6170	-35.64	1.79
u_11(c)	11 (110)	7190	5190	1800	1666.67	4335	4595	4485	-16.47	1.37
u_12(a)	12 (169)	12960	—	1800	—	5095	5215	5170	—	22.81
u_12(b)	12 (133)	5450	—	1800	—	3990	4050	4030	—	2.61
u_12(c)*	12 (88)	5405	4310	125,95	0.00	4310	4515	4380	0.00	0.41

Table 1: Results, PRH, ILP and IDCH

6 Conclusion and future work

In this paper, we have addressed the scheduling problem of care tasks of the Medical Day Unit of the Emergency Department (MDU-ED) of Jeanne de Flandres University Hospital in Lille (France). We proposed an ILP model and an IDCH approach. Computational experiments are performed on generated instances based on actual patients' pathways. We compared the results obtained between the proposed mathematical model and the IDCH approach.

The results show that IDCH obtained strictly better results within short processing time. The tuning permits to achieve a good trade-off between the qual-

ity of the solution and the processing time. In collaboration with the practitioners of the MDU-ED of Jeanne de Flandres University Hospital, we are currently in the process of adapting this approach to the dynamic context that requires high reactivity.

A future research direction would be to consider a broader problem by considering non-standard capacity such as stretchers that can exceptionally be used for a short time as a place for a few patients, or extra capacity that can be mobilized to face a longer term stress situation as for the Covid pandemic.

References

- [1] J. F. Bard and H. W. Purnomo, "Hospital-wide reactive scheduling of nurses with preference considerations," IIE Transactions (Institute of Industrial Engineers), vol. 37, no. 7, pp. 589–608, jul 2005.
- [2] L. V. Green, J. Soares, J. F. Giglio, and R. A. Green, "Using queueing theory to increase the effectiveness of emergency department provider staffing," Academic Emergency Medicine, vol. 13, no. 1, pp. 61–68, 2006.
- [3] E. K. Burke, J. Li, and R. Qu, "A hybrid model of integer programming and variable neighbourhood search for highly-constrained nurse rostering problems," European Journal of Operational Research, vol. 203, pp. 484–493, 2009.
- [4] C. F. F. Costa Filho, D. A. R. Rocha, M. G. F. Costa, and W. C. de Albuquerque Pereira, "Using constraint satisfaction problem approach to solve human resource allocation problems in co-operative health services," Expert Systems with Applications, vol. 39, no. 1, pp. 385–394, 2012.
- [5] N. Izady and D. Worthington, "Setting staffing requirements for time dependent queueing networks: The case of accident and emergency departments," European Journal of Operational Research, vol. 219, no. 3, pp. 531–540, jun 2012.
- [6] D. Sinreich, O. Jabali, and N. P. Dellaert, "Reducing emergency department waiting times by adjusting work shifts considering patient visits to multiple care providers," IIE Transactions, vol. 44, no. 3, pp. 163–180, mar 2012.
- [7] A. Elalouf and G. Wachtel, "Queueing problems in emergency departments: A review of practical approaches and research methodologies," Operations Research Forum, vol. 3, no. 1, 2022.
- [8] Şafak Kırış, N. Yüzügüllü, N. Ergün, and A. Alper Çevik, "A knowledge-based scheduling system for emergency departments," Knowledge-Based Systems, vol. 23, no. 8, pp. 890–900, 2010.
- [9] F. Ajmi, H. Zgaya, S. B. Othman, and S. Hammadi, "Agent-based dynamic optimization for managing the workflow of the patient's pathway," Simulation Modelling Practice and Theory, vol. 96, nov 2019.
- [10] A. Azadeh, M. Hosseinabadi Farahani, S. Torabzadeh, and M. Baghersad, "Scheduling prioritized patients in emergency department laboratories," Computer Methods and Programs in Biomedicine, vol. 117, no. 2, pp. 61–70, nov 2014.
- [11] S. Lee and Y. H. Lee, "Improving Emergency Department Efficiency by Patient Scheduling Using Deep Reinforcement Learning," Healthcare, vol. 8, no. 2, pp. 77–78, mar 2020.
- [12] R. Luscombe and E. Kozan, "Dynamic resource allocation to improve emergency department efficiency in real time," European Journal of Operational Research, vol. 255, no. 2, pp. 593–603, 2016.
- [13] I. I. Cplex, "IBM ILOG CPLEX Optimization Studio: CPLEX User's Manual," 2020. [Online]. Available: <https://www.ibm.com>