A model-based approach to the optimization of an Emergency Department

F. Amato[2], F. Basile[1], E. Calce[1], R. Granata[2], M. Romano[2] and A. M. Ponsiglione[2]

***Abstract*— Emergency Departments (EDs) are highly complex systems that frequently experience congestion due to the increasing demand for healthcare services and the continuous expansion of clinical capabilities. Improving quality of care typically requires additional resources, which are inherently limited. Consequently, optimal resource allocation represents a critical strategy to address one of the main operational challenges in EDs: the reduction of Length of Stay (LOS). This study contributes to this research area by formulating a constrained optimization problem aimed at identifying the optimal configuration of medical staff. The healthcare workflow is first obtained using Process Mining (PM) techniques. The resulting model is then employed in a simulation environment, where LOS is evaluated and minimized through a Genetic Algorithm (GA)-based optimization problem.**

Index Terms—Discrete event systems, Petri nets, Process mining, Emergency Department, Genetic Algorithm.

I. Introduction

Healthcare systems are currently facing unprecedented challenges, particularly in terms of economic sustainability and resource constraints [16]. Healthcare engineers and managers continuously investigate the efficiency of existing systems while exploring opportunities for operational improvement [2]. Evaluating proposed interventions prior to implementation is essential; however, this task is inherently complex due to several factors: the high degree of human involvement at both patient and resource levels (e.g., physicians and nurses), limited budgets and infrastructure, and the large number of interdependent decision variables such as medical staff scheduling and bed capacity. At the same time, patients demand high standards of service quality and safety.

Among hospital units, the ED operates under particularly intense pressure, as it must provide timely and appropriate care to patients with urgent medical needs [3]. EDs frequently experience overcrowding, resource shortages, skill mismatches, inadequate equipment, inefficient workflows, and treatment delays. A significant contributing factor is the steady increase in elderly patients, who require more frequent admissions and consume more resources compared to younger populations. Although EDs are designed to serve heterogeneous patient groups, demographic shifts substantially impact operational performance, particularly the LOS.

LOS is widely recognized as a key performance indicator of ED efficiency and quality of care [7]. It is typically defined as the total time a patient spends in the ED and is often interpreted as an indicator of system congestion and operational stress. Efficient patient flow is therefore essential not only for performance improvement, but also for patient safety, satisfaction, and overall care optimization.

Each patient traverses multiple process stages, whose performance depends on both human and non-human resources, as well as on organizational procedures and staff expertise. Consequently, maximizing ED performance under resource constraints requires systematic resource optimization. Various modeling approaches have been proposed in the literature to represent ED services and support performance improvement.

In recent years, metaheuristic algorithms have emerged as powerful tools for addressing complex optimization problems in healthcare management [13]. These algorithms can efficiently search through large solution spaces and locate optimal or near-optimal responses to complex issues. Among them, GAs have been successfully applied to healthcare resource allocation problems [18]. GAs iteratively evolve candidate solutions inspired by evolutionary principles. Their robustness to stochastic variability makes them particularly suitable when system performance is evaluated through simulation.

In parallel, PM has gained increasing relevance as a event-driven methodology for discovering, monitoring, and improving real-world processes from event logs [1]. In modern healthcare environments, integrated information systems provide structured event data that enable accurate workflow reconstruction. By applying PM techniques, *Petri net* (PN) models of ED workflows can be derived. These models are well suited to represent concurrency, decision points, and resource sharing, which are intrinsic characteristics of healthcare systems. Furthermore, they can be simulated to obtain quantitative performance indicators such as mean LOS.

The main contribution of this paper is the optimization of an ED through the strategic allocation of medical staff resources. First, the healthcare workflow is discovered via PM and used for the simulation. Subsequently, a GA is used to identify the optimal staff configuration aimed at minimizing mean LOS.

II. PRELIMINARIES

A. Petri Net

In the following section we assume that the reader is well acquainted to basic notation on PNs [19].

A *PN* is defined as 4-tuple: $N = (P, T, \mathbf{Pre}, \mathbf{Post})$, where P is the set of m places (represented by circles), T is the set of n transitions (represented by boxes), and $\mathbf{Pre}$ and $\mathbf{Post}$ are the pre- and post-incidence matrices, respectively. $\mathbf{C} = \mathbf{Post} - \mathbf{Pre}$ is the incidence matrix. We denote as $\mathbf{m} \in \mathbb{N}^m$ the PN marking, while $\mathbf{m_0}$ is the initial one.

A transition t is enabled under the marking $\mathbf{m}$ if and only if $\mathbf{m} \geq \mathbf{Pre}(\cdot, t)$; the transition enabling is denoted as $\mathbf{m}[t\rangle$. An enabled transition t may fire, yielding the marking $\mathbf{m}' = \mathbf{m} + \mathbf{C}(\cdot, t)$, i.e., $\mathbf{m}[t\rangle\mathbf{m}'$. The *enabling*

[1] F. Basile and E. Calce are with DIEM, University of Salerno, Fisciano 84084, Italy. {fbasile, encalce}@unisa.it. Enrica Calce is the corresponding author.

[2]F. Amato, R. Granata, M. Romano and A.M. Ponsiglione are with Department of Electrical Engineering and Information Technology, University of Naples "Federico II", Naples, Italy. {framato, rita.granata, mariarom, alfonsomaria.ponsiglione}@unina.it

degree of a transition t_i, enabled under the marking $\mathbf{m}$ is the largest integer k such that $\mathbf{m} \geq k \cdot \mathbf{Pre}(\cdot, t)$. A net system $\mathcal{S} = (N, \mathbf{m}_0)$ is given by a net and its initial marking. An *admissible firing sequence* from $\mathbf{m}$ is a sequence of transitions $\sigma = t_1 t_2 \ldots t_k$ such that $\mathbf{m}[t_1\rangle \mathbf{m}_1 [t_2\rangle \mathbf{m}_2 \ldots [t_k\rangle \mathbf{m}_k$, i.e., $\mathbf{m}[\sigma\rangle \mathbf{m}_k$. The notation $\mathbf{m}[\sigma\rangle$ denotes an enabled sequence under the marking $\mathbf{m}$. The length of a sequence σ is denoted by $|\sigma|$. The notation $\sigma = \pi(\sigma)$ is used to denote that σ is the *firing count vector* (FCV) of σ. If $\mathbf{m}_0[\sigma\rangle \mathbf{m}$, then it is possible to write $\mathbf{m} = \mathbf{m}_0 + \mathbf{C} \cdot \sigma$ which is called the *state equation* of the net system. We denote as $T_e(\mathbf{m}) = \{t \in T \mid \mathbf{m} \geq \mathbf{Pre}(\cdot, t)\}$ the set of transitions enabled at $\mathbf{m}$.

A *conflict* is a situation where not all that is enabled can occur at once [4]. More formally, t, $t' \in T$ are in *effective* conflict relation at marking $\mathbf{m}$ if there exist k, $k' \in \mathbb{N}$ such that $\mathbf{m} \geq k \cdot \mathbf{Pre}(\cdot, t) \wedge \mathbf{m} \geq k' \cdot \mathbf{Pre}(\cdot, t')$, but $\mathbf{m} \ngeq k \cdot \mathbf{Pre}(\cdot, t) + k' \cdot \mathbf{Pre}(\cdot, t')$. A necessary condition for the effective conflict is that $\bullet t \bigcap \bullet t' \neq \emptyset$, and in this case t and t' are said to be *structural* conflict relation. The structural conflict relation (or *choice*) is a structural pre-requisite for the behavioral property of conflict.

In this paper, timed net models are used. In a timed net model a transition t can be fired at time τ if the time elapsed since its enabling is $G(t)$. Depending on how $G(t)$ is characterized, different timed net classes can be defined [21]. An *Arbitrary Stochastic Petri Nets* (ASPNs) [21] is defined as a couple: $N_s = (N, G)$, where:

- $N = (P, T, \mathbf{Pre}, \mathbf{Post})$ is a PN;
- $G : T \rightarrow \mathbb{R}$ is a firing function that associates each transition with an arbitrarily distributed firing times.

In practice each duration has an own statistical characterization. The firing of a transition will be considered *atomic*, meaning that token consumption and production occur as an indivisible event. During an atomic firing, all tokens required from input places are consumed simultaneously, and all tokens are deposited in output places without any intermediate state.

B. Genetic Algorithm

We first summarize the general concepts behind GAs [8].

Assume we have a discrete search space $\mathcal{X}$ and a function $f : \mathcal{X} \rightarrow \mathcal{R}$. The general problem is to find: $\min_{x \in \mathcal{X}} f(x)$.

Here, x is a vector of *decision variables* and f is the *objective function*. That is, the vector x is represented by a string s, of length l, made up of symbols drawn from an alphabet $\mathcal{A}$, using a mapping $c : \mathcal{A}^l \rightarrow \mathcal{X}$.

In practice, we may need to use a search space $\mathcal{S} \subseteq \mathcal{A}^l$, to reflect the fact that some strings in the image of $\mathcal{A}^l$ under c may represent invalid solutions to the original problem. The string length l depends on the dimensions of both $\mathcal{X}$ and $\mathcal{A}$.

Thus the optimization problem becomes one of finding $\min_{s \in \mathcal{S}} g$ where the function $g(s) = f(c(s))$. It is usually desiderable that c should be a bijection.

In the case of GAs, a *population* of strings is used, and these strings are often referred to in the GA literature as *chromosomes*. The recombination of strings is carried out using simple analogies of genetic *crossover* and *mutation*, and the search is guided by the results of evaluating the objective function g for each string in the population. Based on this evaluation, strings that have lower objective function values (i.e., represent better solutions) can be identified, and these are given more opportunity to breed. A basic template for the generation of a GA is despicated in Algorithm 1.

According to [10], a common convergence criterion in GAs is to stop the evolution when the objective function shows no significant improvement over several consecutive generations.

Algorithm 1: Genetic Algorithm Generation

```
Choose an initial population of chromosomes;
while termination condition not satisfied do
    repeat
        if crossover condition satisfied then
            select parent chromosomes;
            choose crossover parameters;
            perform crossover;
        if mutation condition satisfied then
            choose mutation points;
            perform mutation;
        evaluate objective function values of
          offspring;
    until sufficient offspring created;
    select new population;
```

When the objective function g does not have an analytical expression, it is evaluated using a simulator. In particular, if a timed net system needs to be simulated *infinite-server* semantic is assumed, i.e. all the enabled instances of a transition t are handled at the same time. In particular, the semantics used is the k-server one, where a transition t can be executed up to k times concurrently. This is modeled by adding a self-loop place with k tokens to each transition t. The firing of transitions at the i-th simulation step is managed by a logical structure, named *Scheduled Event List* (SEL) [6], which is a list of couples (T_i, τ_i), where T_i is the set of transition enabled and its scheduled firing time τ_i. According to Algorithm 2, the SEL is initialized as empty and at first step it contains all the enabled transitions at τ_0 and their scheduled firing times. The SEL sorts all the transitions based on the minimum scheduled firing time and chooses it as the current step time. The transitions associated to the minimum scheduled firing time can fire, updating the simulation state and they are deleted from the SEL. In the case of structural conflicts, or rather when more than one transition share the same input place, the transition to fire is chosen with a probabilistic rule. At the last simulation step, named MS, the SEL is updated with the new enabled transitions and the transitions that are no longer enabled are deleted.

III. MAIN RESULT

We propose a methodological framework for the discovery, analysis, and optimization of a healthcare workflow. First, the logical process model is obtained from an event log using PM algorithm, which transform raw healthcare event log into a structured business process representation. To enable performance analysis, the temporal information contained in the event log is also exploited. Specifically, a statistical analysis is conducted to estimate the duration distributions associated with each transition, thereby enriching the model with time-related parameters. Once both logical consistency and temporal characterization are ensured, the resulting model is employed in a simulation environment. The simulation framework is then used to evaluate different

Algorithm 2: Simulation algorithm of a net system

Input: $\mathbf{m}_0$, τ_0, MS
Output: SEL
Initialize: $i = 0$, $SEL = [\,]$, $\mathbf{m}_{curr} = \mathbf{m}_0$, $\sigma_{curr} = zeros(|T|)$
for *all* $t \in T_e(\mathbf{m}_{curr})$ **do**
  $\tau_e = \tau_0$
  Append (t, τ_e) to SEL
while $i \leq MS$ ***and*** $SEL \neq \emptyset$ **do**
  $i = i + 1$
  Solve conflicts according to the chosen resolution policy
  Extract (t_i, τ_i) from SEL such that $\bar{\tau} = \min\{\tau_e | (t_i, \tau_e) \in SEL\}$
  $\tau_{curr} = \tau_i$
  $\sigma_{curr}(t_i) = 1$
  $\mathbf{m}_{curr} = \mathbf{m}_{curr} + \mathbf{C} \cdot \sigma_{curr}$
  Remove (t_i, τ_i) from SEL
  for *all* $t \in T_e(\mathbf{m}_{curr})$ **do**
    Append (t, τ_{curr}) to SEL
return SEL

resource configurations generated by a GA, with the aim of identifying performance improvements.

A. Process Discovery

Process discovery is one of the most challenging process mining tasks. A process model is discovered from the observed behavior, based on event logs.

Event logs serve as the starting point for the process mining and is a set of traces on T [1]. Sequences are the most natural way to represent traces in an event log. A trace can be represented with transitions, i.e. $\sigma \in T^*$[1].

A *process discovery algorithm* maps each event log onto a PN system. For this approach, we have chosen to apply the *Heuristic Miner* algorithm, which discovers a process model analyzing the causal dependencies in the event log, i. e., if an event is always followed by another event, it is likely that there is a dependency relation between both events. For more details, the reader can refer to [22].

B. Statistical Analysis

Taking into account the logical model from PM, we want to conduct a statistical analysis.

Considered a timed sequence $\sigma_T = (T_1, \tau_1) \dots (T_q, \tau_q) \dots (T_L, \tau_L)$ where T_q is the set of transitions fired at time τ_q, $\tau_1 < \tau_2 \dots < \tau_L$ denote firing time instants, is called timed firing sequence. The position q the couple (T_q, τ_q) occupies in the sequence is called step time, so (T_1, τ_1) is associated with step 1, (T_2, τ_2) is associated with step 2 and so on. The length of a sequence σ_T is denoted by $|\sigma_T|$. A timed event log can be represented with a set of timed sequences.

If a timed event log is available, as in this paper, the process discovery algorithm can be conducted by using its logical counterpart. Such a logical counterpart, can be obtained from a timed sequence by considering for each time instant τ_q any interleaving in T_q.

In healthcare workflows, the timing structure reflects the unpredictability of the real world. The duration of each activity can vary significantly from one patient to another or from one ward to another. For the sake of simplicity, the arrival time of a patient has a different timing distribution from a medical examination, which depends on the patient's health condition.

According to Algorithm 3, F is the number of transitions in an event log E_{L_F} and Dur is a data structure containing the activity durations for each transition. For each fired transition from E_{L_F}, the algorithm extracts its enabling timing from the SEL. It is so possible to compute the activity duration of each transition, update the state and remove the transition from the SEL. For the next transitions, the algorithm checks which transitions are enabled under the updated marking. Those transitions are appended to SEL.

Algorithm 3: Computation of activity durations for each transition

Input: E_{L_F}, $\mathbf{m}_0$, τ_0 and F
Output: Dur
Initialize: $i = 0$, $\mathbf{m}_{curr} = \mathbf{m}_0$, $SEL = [\,]$, $fired\_transitions = [\,]$, $\sigma_{curr} = [\,]$ and $D = [\,]$
for *all* $t \in T_e(\mathbf{m}_{curr})$ **do**
  $\tau_e = \tau_0$
  Append (t, τ_e) to SEL
$i = i + 1$
while $i \leq F$ **do**
  Extract (t_i, τ_i) from E_{L_F}
  Append (t_i, τ_i) to $fired\_transitions$
  Find $(t_i, \bar{\tau}) \in SEL$ such that $\bar{\tau} = \min\{\tau_e | (ti, \tau_e) \in SEL\}$
  $\sigma_{curr} = zeros(|T|)$
  $Dur(t_i) = \tau_i - \bar{\tau}$
  $\sigma_{curr}(t_i) = 1$
  $\mathbf{m}_{curr} = \mathbf{m}_{curr} + \mathbf{C} \cdot \sigma_{curr}$
  Remove $(t_i, \bar{\tau})$ from SEL
  $\tau_e = \bar{\tau}$
  $i = i + 1$
  for *all* $t \in T_e(\mathbf{m}_{curr})$ **do**
    Append (t, τ_e) to SEL
return Dur

We show the process duration distributions using boxplots for each transition and analyze which data distribution best fits the activity duration distribution, according to the Kolmogorov-Smirnov (KS) test [17], evaluated with *p-value*. Taking into account the empirical distributions, the activity duration of each transition is compared with them. The KS test returns a p-value, which indicates whether the observed difference between the distributions is statistically significant.

Once the statistical characterization is completed, the duration of each transition is available and the healthcare workflow results to be modeled by an ASPN system.

C. Optimization

LOS_i, associated with the i-th patient, is computed as follow:

$$LOS_i = \tau_{out_i} - \tau_{in_i} \tag{1}$$

where τ_{out_i} is the timestamp associated with the exit from the healthcare workflow, while τ_{in_i} is the timestamp associated with the entry into the healthcare workflow.

The mean LOS represents a fundamental variable in the formulation of the optimization problem because it is the objective function $g(x)$.

[1]The notation T^* denotes the Kleene closure of T (see [6, Ch. 2]).

We solve the following optimization problem:

$$\min_{\mathbf{x}} \ \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} LOS_i(\mathbf{x})\right] \tag{2}$$

subject to

$$\begin{cases} \displaystyle\sum_{j=1}^{M} c_j x_j \leq c_{\max} & \text{(3a)} \\ l_{b_j} \leq x_j \leq u_{b_j}, \quad j = 1, \ldots, M & \text{(3b)} \end{cases}$$

where the objective function (2) minimizes the expected mean LOS with respect to M resources across all the N patients in the considered time horizon.

Constraint (3a) ensures that the total cost of the resources does not exceed the daily budget $c_{\max}$. Constraint (3b) ensures that the number of each resource units is bounded between the minimum and maximum allowed values.

According to [14], although a larger number of replications would reduce the variance of the estimator, a trade-off between computational effort and estimation accuracy should be found.

LOS_i has not an analytical expression, so it is evaluated by simulation. To this aim the discovered model can be exploited. To perform optimization the net system devised in Sections III-A-III-B must be completed to solve conflicts, to enable resource allocation and to distinguish each patient.

In healthcare workflows, choices are common, such as the assignment of a triage code or the selection of a medical examination. These choices introduces conflicts between transitions. The same occurs to manage resources.

Assumption 1 (Conflicting transitions): It is assumed that all transitions that make up a choice – i.e., all transitions $t \in p^\bullet$ with $|p^\bullet| > 1$ – are immediate (i.e., their duration is null). Hence if $|p^\bullet| > 1 \quad \Rightarrow \quad dur(t) = 0 \, \forall \ t \in p^\bullet$, for all the places in P. This assumption is motivated by the consideration that, when a timed activity is associated with conflicting transitions, a conflict resolution policy may be a race between conflicting transitions, which is pointless for healthcare management systems. ◊

According to Assumption 1, each structural conflict is modified as shown in Fig.1. Each activity transition involved in the conflict is split into two transitions: an immediate transition $t_{i_{im}}$, associated with a probability weight, which resolves the conflict, and a timed t_i, representing the execution of the activity and characterized by its duration.

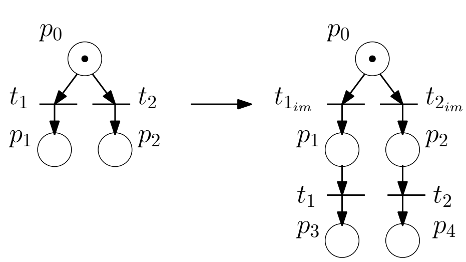

Fig. 1: A conflict between timed transition is replaced with a conflict between immediate transitions.

The conflict resolution between transitions can be obtained by using *routing rates*. Let $T' = \{t_1, \cdots, t_k\} \subset T$ be in structural conflict (i.e. having equal pre-incidence function: $\mathbf{Pre}(.,t_1) = \cdots = \mathbf{Pre}(.,t_k)$). The constants $r_1, \cdots, r_k \in \mathbb{N}^+$ (*routing rates*) are explicitly defined in PN interpretation in such a way that when $t_1, \cdots, t_k$ are enabled, transition $t_i \in T'$ fires with probability $r_i / (\sum_{j=1}^{k} r_j)$. When a transition t_i is involved in multiple structural conflicts simultaneously, the conflicts resolved by establishing firing priorities among the conflicting transitions.

Then, resources must be explicitly represented in the model. Resources are modeled as places. Each resource place has an outgoing arc toward the activity transition representing resource allocation, and an incoming arc from the activity transition representing resource release.

Finally, colored tokens are associated with the places of the discovered net (not for the places modelling the resources). Indeed, each patient has an ID, which is a sequential number used to be identified. Then, the discovered net becomes a colored Petri net (CPN) to perform the resource optimization. Precisely, such a CPN has the same skeleton and timed structure of the discovered one, but in this way the flow of tokens allows to follow the flow of each patient in the system, i.e. to evaluate LOS_i. For the sake of brevity, background material about CPN is not recalled here. For further details, the reader can refer to [4].

IV. CASE STUDY

In this section, we consider a case study to prove the proposed approach. The code has been developed in Python. In particular, for the Process Discovery, it has been solved with *PM4py* library [5] and, for the statistical analysis, it has been solved with *SciPy* library [20].

The considered case study is an ED workflow, under normal conditions, obtained from the simulated event log, named E_{L_F}. E_{L_F} contains data from 3000 patients, of which 15.4% are white codes, 56% green codes, 18.6% yellow codes, 6.67% orange codes, and 3.33% red codes.

Without loss of generality, each event is assumed to be associated with a transition, as follows:

- t_1: patient arrival and check-in at the ED;
- t_2, t_3, t_4, t_5, and t_6: the evaluation of white, green, yellow, orange, and red triage codes, respectively;
- t_7: physician examination of a white-code patient;
- t_8: discharge of a white-code patient;
- t_9: physician examination of a green-code patient;
- t_{10}: discharge of a green-code patient with prescribed therapy;
- t_{11}: physician decision on routine tests to be performed;
- t_{12}, t_{13}, and t_{14}: ultrasound, X-ray, and blood sampling, respectively;
- t_{15}–t_{20}: transfer and discharge related to each routine test, respectively;
- t_{21} and t_{22}: physician examination of a yellow-code and an orange-code patient, respectively;
- t_{23}: stabilization treatment;
- t_{24}, t_{25}, and t_{26}: emergency transfer, hospital admission, and discharge, respectively;
- t_{27}, t_{28}, t_{29}, and t_{30}: shock room, emergency surgery, transfer to another hospital, and intensive care unit admission, respectively;
- t_{31}: patient check-out from the ED.

By applying the *Heuristic Miner* algorithm, we obtain the PN shown in Fig.2. The healthcare workflow describes the patients flow from their arrival to their discharge. Upon arrival at the ED, patients may access care either as walk-ins or via ambulance transport, after which they undergo triage assessment. A team of qualified nurses evaluates the

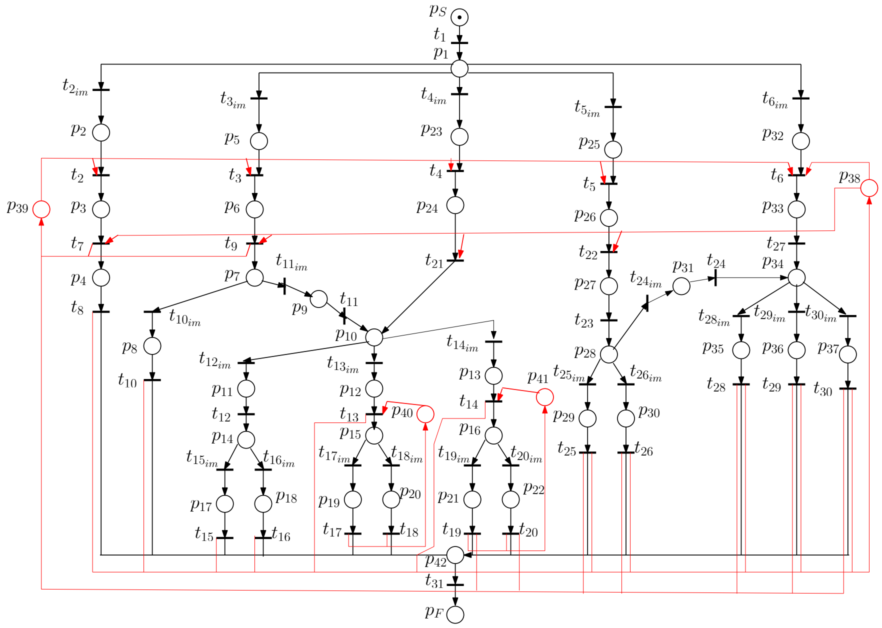

Fig. 2: Emergency Department PN

patient's clinical condition and assigns a triage code that determines the priority level for access to care. Based on the assigned triage category, patients follow different clinical pathways. For example, patients classified with a white code typically undergo a standard physician examination only, whereas those assigned a green or yellow code, reflecting a moderately critical condition, undergo both a medical examination and routine diagnostic tests. Following the clinical assessment and any required diagnostic procedures, the physician updates the patient's electronic health record with the prescribed therapy and clinical decisions. The patient is then discharged from the ED.

After discovering the logical PN model, its timed counterpart can be statistically characterized using the algorithm 2. We show all the transition durations using boxplots in Fig. 3.

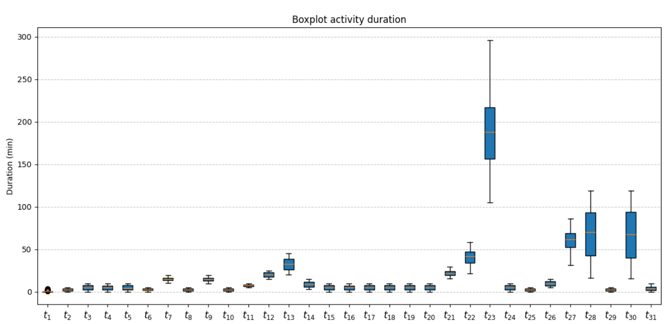

Fig. 3: Boxplot of activity duration for each transition.

For example, the transition t_1 results to be characterized by an exponential distribution to model patient arrivals in the ED and it is assumed to work under single-server semantic (one patient arrives at time). This is coherent with usual assumption in practice [11]. The other transitions result to be characterized by uniform and triangular distributions (see Table I) and they are assumed to work under infinite-server semantic. The *exponential* distribution is parameterized by the rate λ, representing the constant transition rate. The *uniform* distribution is specified by the lower and upper bounds a_u and b_u, respectively. The *triangular* distribution is characterized by three parameters, a_t, b_t, and c_t, corresponding to the lower bound, upper bound, and mode, respectively.

Transition	**Distribution**	**Parameters**
t_1	Exponential	$\lambda = 3.2$
t_2, t_6, t_8, t_{10}	Uniform	$a = 0$, $b = 5$
t_3, t_4, t_5, t_{15}, t_{16}, t_{17}, t_{18}, t_{19}, t_{20}, t_{24}, t_{25}, t_{29}, t_{31}	Uniform	$a_u = 0$, $b_u = 10$
t_7, t_9	Triangular	$a_t = 10$, $b_t = 20$, $c_t = 15$
t_{11}	Uniform	$a_u = 5$, $b_u = 10$
t_{12}	Uniform	$a_u = 15$, $b_u = 25$
t_{13}	Uniform	$a_u = 20$, $b_u = 45$
t_{14}	Uniform	$a_u = 3$, $b_u = 15$
t_{21}	Triangular	$a_t = 15$, $b_t = 30$, $c_t = 20$
t_{22}	Triangular	$a_t = 20$, $b_t = 60$, $c_t = 40$
t_{23}	Triangular	$a_t = 90$, $b_t = 300$, $c_t = 180$
t_{26}	Uniform	$a_u = 5$, $b_u = 15$
t_{27}	Triangular	$a_t = 30$, $b_t = 90$, $c_t = 60$
t_{28}, t_{30}	Uniform	$a_u = 15$, $b_u = 120$

TABLE I: Activity distribution parameters

The ED optimization is carried out with respect to the available human resources, i.e., the medical staff configuration. In this case, the human resources consist of physicians, nurses, radiologists, and laboratory technicians; therefore, $M = 4$. Physicians are responsible for conducting medical examinations and managing the clinical pathway of patients across all triage codes. Nurses perform triage assessment and support physicians in the management of patients with severe clinical conditions. Radiologists are in charge of X-ray examinations, whereas laboratory technicians perform blood sample analyses. These resources are modeled as follows: place p_{38} represents physicians, place p_{39} nurses, place p_{40} radiologists, and place p_{41} laboratory technicians. In Fig. 2, resource places are depicted as red circles, and the tokens therein denote the number of units of each resource available at the initial marking.

The starting place p_S, shown in blue in Fig. 2, does not affect the optimization process, since patient arrivals are generated according to an exponential distribution through transition t_1. Every time t_1 fires, a token with a color associated with a new patient ID is put in p_1. Conversely, the terminal place p_F collects all patients after they leave the ED during the simulation horizon.

The weights associated with immediate transitions $t_{i_{im}}$ are defined according to clinical practice (e.g., routing rates equal to 0.1 for white, 0.2 for green, 0.5 for yellow, 0.15 for orange and 0.05 for red codes).

When transitions are simultaneously involved in multiple conflicts, priorities are assigned based on the activities they enable. In this paper, this scenario occurs with transition t_6, which represents the evaluation of a red triage code. This transition requires both a physician (token in p_{38}) and a nurse (token in p_{39}). The latter is also in conflict with transitions t_2, t_3, t_4, and t_5, which represent the other triage codes. To resolve these conflicts, priorities are defined according to the triage codes: the red code (t_6) has the highest priority, followed in descending order by orange (t_5), yellow (t_4), green (t_3), and finally white (t_2).

Each simulation is performed over a 24-hour time horizon, in line with common practice in ED modeling [12]. Let x_1 be the number of physicians, x_2 the number of nurses, x_3 the number of radiologists and x_4 is the number of technicians. For each medical staff role, a lower and upper bound on the number of available units is imposed to reflect practical and budgetary constraints [9], as follows:

- Physicians: $l_{b_1} = 4$, $u_{b_1} = 8$;
- Nurses: $l_{b_2} = 8$, $u_{b_2} = 12$;
- Radiologists: $l_{b_3} = 2$, $u_{b_3} = 4$;
- Laboratory technicians: $l_{b_4} = 2$, $u_{b_4} = 4$.

Costs associated with human resources (e.g., physicians, nurses, radiologists, and laboratory technicians) are commonly included as input parameters in healthcare models in order to evaluate the economic implications of different staffing configurations [15]. In this paper, plausible cost values were assumed for each staff role (physician: €100, nurse: €80, radiologist: €100, laboratory technician: €80) and a maximum daily budget constraint of $c_{\max} =$€3000 was imposed accordingly.

According to Algorithm 1, an initial population of size PS individuals (configurations) is randomly generated, ensuring all constraints are met. For the successive generations, elitism is applied by preserving the best configuration (the one with the lowest LOS). To generate the remaining $(PS-1)$ individuals, tournament selection is used to select two parents. Each parent is obtained from the comparison of two random configurations from the previous population, the one with the lower LOS is selected. The offspring is obtained via uniform crossover, selecting 50% of the roles from each parent. Subsequently, mutation is applied by randomly modifying roles with a unitary variation. For each new configuration, we estimate the LOS. The best individual with respect to the initial population (8 physicians, 10 nurses, 3 radiologists and 2 technicians) requires a budget cost of 2060€ and returns a mean LOS value of 272 $\pm$ 3% minutes. Running Algorithm 1, the optimal solution is 8 physicians, 9 nurses, 3 radiologists and 4 technicians, has a budget cost of 2140€ and returns a mean LOS value of 264 $\pm$ 3% minutes. The mean value (2) has been evaluated over 100 replications achieving a 95% confidence interval.

V. CONCLUSION

An approach to discover, analyze, and optimize a healthcare workflow in the context of an ED has been proposed. By discovering a logical model through PM, we can statistically characterize its behavior. Effective timing management in healthcare workflows improves both cost efficiency for medical staff and patient satisfaction by reducing LOS. Future research will focus on the use of real event logs from ED healthcare information systems and on incorporating additional constraints to support the planning of alternative medical staff allocation strategies.

REFERENCES

[1] W. M. P. van der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Berlin, Heidelberg: Springer, 2011.

[2] W. Abo-Hamad and A. Arisha, "Simulation-based framework to improve patient experience in an emergency department," *European Journal of Operational Research*, vol. 224, no. 1, pp. 154–166, 2013.

[3] K. B. Ahsan, M. R. Alam, D. G. Morel, and M. A. Karim, "Emergency department resource optimisation for improved performance: a review," *Journal of Industrial Engineering International*, vol. 15, no. Suppl 1, pp. 253–266, 2019.

[4] F. Basile, C. Carbone, and P. Chiacchio, "Simulation and analysis of discrete-event control systems based on Petri nets using PNetLab," *Control Engineering Practice*, vol. 15, no. 2, pp. 241–259, 2007.

[5] A. Berti, S. van Zelst, and D. Schuster, "PM4Py: A process mining library for Python," *Software Impacts*, vol. 17, p. 100556, 2023.

[6] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, 3rd ed. Springer, 2021.

[7] D. Di Laura, L. D'Angiolella, L. Mantovani, et al., "Efficiency measures of emergency departments: an Italian systematic literature review," *BMJ Open Qual.*, vol. 10, no. 3, p. e001058, Sep. 2021.

[8] M. Gendreau and J.-Y. Potvin, *Handbook of Metaheuristics*, 2nd ed., International Series in Operations Research & Management Science, vol. XX. New York, NY, USA: Springer, 2010.

[9] K. Ghanes, M. Wargon, and S. Jouini, "Simulation-based optimization of staffing levels in an emergency department," *Simulation*, vol. 91, no. 10, pp. 942–953, 2015.

[10] D. Greenhalgh and S. Marshall, "Convergence criteria for genetic algorithms," *SIAM Journal on Computing*, vol. 30, no. 1, pp. 269–282, 2000.

[11] S. S. Jones and R. S. Evans, "An agent based simulation tool for scheduling emergency department physicians," in *AMIA Annual Symposium Proceedings*, vol. 2008, p. 338, 2008.

[12] L. Keshtkar, M. Zeraati, and M. J. Tarokh, "A simulation optimization approach for resource allocation in an emergency department," *QScience Connect*, vol. 2015, no. 8, 2015.

[13] A. Lameesa, M. Hoque, M. S. B. Alam, S. F. Ahmed, and A. H. Gandomi, "Role of metaheuristic algorithms in healthcare: a comprehensive investigation across clinical diagnosis, medical imaging, operations management, and public health," *Journal of Computational Design and Engineering*, vol. 11, no. 3, pp. 223–247, 2024.

[14] A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis*, 4th ed. New York, NY, USA: McGraw-Hill, 2007.

[15] S. Liu, Y. Li, K. P. Triantis, et al., "The diffusion of discrete event simulation approaches in health care management in the past four decades: a comprehensive review," *MDM Policy & Practice*, 2020.

[16] M. Masmoudi, B. Jarboui, and P. Siarry, *Operations Research and Simulation in Healthcare*, 1st ed. Cham, Switzerland: Springer, 2021.

[17] F. J. Massey, "The Kolmogorov-Smirnov test for goodness of fit," *Journal of the American Statistical Association*, vol. 46, no. 253, pp. 68–78, 1951.

[18] M. Mitchell, *An Introduction to Genetic Algorithms*. Cambridge, MA, USA: MIT Press, 1998.

[19] T. Murata, "Petri Nets: Properties, Analysis and Applications," *Proc. of IEEE*, vol. 77, no. 4, pp. 541–580, April 1989.

[20] P. Virtanen, R. Gommers, T. E. Oliphant, et al., "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, no. 3, pp. 261–272, 2020.

[21] J. Wang, "Stochastic timed Petri nets and stochastic Petri nets," in *Timed Petri Nets: Theory and Application*, Boston, MA, USA: Springer, 1998, pp. 125–153.

[22] A. Weijters, W. van der Aalst, and A. K. A. de Medeiros, "Process mining with the heuristics miner-algorithm," *CIRP Annals - Manufacturing Technology*, vol. 166, Jan. 2006.