



HAL
open science

Exploration of (de-)centralising scheduling in an Emergency Department

Jessica Florencia, Thierry Moyaux, Lorraine Trilling, Guillaume Bouleux,
Vincent Cheutet

► **To cite this version:**

Jessica Florencia, Thierry Moyaux, Lorraine Trilling, Guillaume Bouleux, Vincent Cheutet. Exploration of (de-)centralising scheduling in an Emergency Department. SOHOMA workshop 2023, IAE Annecy, Sep 2023, Annecy, France. hal-04252091

HAL Id: hal-04252091

<https://hal.science/hal-04252091>

Submitted on 20 Oct 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.

Exploration of (de-)centralising scheduling in an Emergency Department

Jessica FLORENCIA¹, Thierry MOYAUX¹, Lorraine TRILLING¹,
Guillaume BOULEUX¹, and Vincent CHEUTET¹

Abstract The choice of decision structure with a more or lesser degree of (de-)centralisation is important since it affects the operation and decision-making process of enterprises in uncertain situations. The Emergency Department (ED) is the critical and main part of the hospital. There are various crucial decisions to be taken quickly under uncertainty and constraints in EDs, including resource scheduling. The exploration of the decision structure for this decision is required to improve the patients' pathway. This article explores the (de-)centralisation of decision, *i.e.*, centralised and decentralised models of nurse-to-patient scheduling in EDs. We base our centralised scheduling on a Mixed Integer Linear Program (MILP), and our decentralised scheduling makes a multi-agent system run a Contract Net Protocol (CNP) in which the agents locally optimise a variant of this MILP. We assess and compare both models. The result shows that both models generate similar patients' schedules.

Keywords: Decentralisation/centralisation of decision; Scheduling; Contract Net Protocol (CNP); Emergency Department (ED)

1 Introduction

Enterprises nowadays face various challenges and are required to take decisions in an uncertain situation [3]. The operation and decision-making processes of such an enterprise are greatly affected by the choice of its decision structure or organisation [11]. An enterprise may have a centralised organisation capable of taking optimal decisions by optimising the entire system, or a decentralised organisation with local decision-making and good reactivity [5, 12]. The enterprise may also apply a hybrid structure that combines the

¹University Lyon, INSA Lyon, Univ. Jean Monnet Saint-Étienne, Univ. Claude Bernard Lyon 1, Univ. Lyon 2, DISP UR4570
e-mail: {firstname.lastname}@insa-lyon.fr

strengths of both organisations: the optimality of centralisation and the reactivity of decentralisation in order to operate in both normal and disturbed situations [3, 12].

We study this (de-)centralisation of decisions in the context of the control of an Emergency Department (ED). An ED is a main and complex part in the healthcare system as it provides first-level and immediate care to patients. The decision-making process in EDs for managing operations is complex due into limited resource availability, dynamic and uncertain patient entry flow to ED, and stochastic and multi-stage patient treatment [1, 7, 4]. The decision-makers have to make sure that the decisions guarantee good quality of care for patients and improve their pathway [10, 4].

The exploration of (de-)centralised decision structures to a more or lesser degree is needed in order to improve decision-making by making a trade-off between optimality (which favours centralisation) and reactivity (which is provided by decentralisation). Among all decisions in EDs, one of the most crucial is resource allocation or scheduling (we use the terms "allocation" and "scheduling" interchangeably in this paper) [9, 10]. The contribution of this article is to propose and compare two models of resource scheduling. The first model has a single decision maker who schedules the jobs (patients) to resources (nurses) with a Mixed Integer Linear Program (MILP). The decentralised model makes the nurses solve a decentralised version of this MILP in order to study the two models as closely as possible. We apply this comparison to the Paediatric ED of Saint-Étienne University Hospital, France.

The outline of the paper is as follows. Section 2 presents the related research on dynamic resource scheduling and centralised and decentralised organisation. Section 3 explains our two models with different levels of (de-)centralisation for resource scheduling. Section 4 discusses the experimental results of both models. Section 5 concludes.

2 Related research

2.1 Centralisation and decentralisation

The discussion on whether to implement more centralised or decentralised organisations has emerged in many fields [6, 11]. This decision on which organisation to be applied is important as it affects the efficiency of the enterprise [11]. A Centralised Organisation (CO) has and utilises global information to make optimal decisions and optimise the system globally [5]. It performs better in a deterministic environment and for predictive purposes. However, it lacks adaptability and reactivity to the changes or disturbances which happen in the system [3, 12]. In contrast, Decentralised Organisations (DOs) involve agents who use their local information in order to make locally applicable decisions [5]. A DO has good reactivity to adapt to disturbances. However,

it has minimum global present and future information which leads to a loss of global long-term performance [3]. Creating a hybrid of both CO and DO emerge to get advantages and minimize the drawbacks of both organisations. This approach combines CO and DO to achieve a globally optimised solution with good reactivity [3, 12].

Many previous studies compare CO and DO quantitatively [5, 11] or qualitatively to find which one performs the best in their fields of application. For example, such a comparison is based on measuring the efficiency of seven mechanisms with different levels of (de-)centralisation to solve a modified multiple travelling salesmen problem [11]. The results show that CO produces results with better quality as DO has a longer median total route length than CO.

The previous researches demonstrate that the performance of CO and DO depends on the problem. CO performs better in static and deterministic situations, while DO is the best in dynamic problems. In addition, the hybrid of both actually shows better results compared to pure CO or DO [3, 6, 12]. Another noticeable aspect is that most comparisons are done in the manufacturing or logistics field. To our knowledge, little to no comparative studies were performed in healthcare management, hence, the novelty of our work. Our research develops the centralised and decentralised decision-making models, similar to [11] for nurse-to-patient scheduling in an ED.

2.2 Resource scheduling in an ED

The ED is an important part of the healthcare system because it provides initial, immediate and essential medical care to patients. It has the difficult task of providing 24-hour emergency services, offering good quality service and ensuring that valuable resources are well-utilised [1, 4]. However, at the same time, it is the most overcrowded component because it faces a dynamic and large number of visits of patients, but has limited manpower [7, 9]. Therefore, manpower planning and scheduling is a crucial decision to be made in EDs. Resources such as physicians and nurses have to be properly allocated to patients such that these patients receive good quality care [4, 10].

Many previous studies explore various resource planning approaches for EDs, ranging from static and predictive to dynamic and online scheduling. Harzi *et al.* [8] develop predictive patient scheduling in ED using Mixed Integer Linear Programming (MILP) that significantly reduces the patients' waiting time. Luscombe *et al.* [10] propose a dynamic scheduling framework that provides real-time support for managing the scarce resources of an ED. The proposed framework has two layers of scheduling to minimize the patients' response and treatment times. It optimises patient-to-bed allocation with parallel machine scheduling and resource task allocation using flexible flow shop scheduling. Duma *et al.* [7] reduce the care process duration and make an ED less crowded by utilising real-time resource allocation.

Our CO model mostly relies on the model by Luscombe *et al.* [10] and Cao *et al.* [2] for the centralised dynamic nurse allocation model with MILP. The MILP model is mostly taken from the model of parallel machine scheduling because nurse scheduling has a similar model to parallel machine scheduling. Our DO model is mostly adapted from Moyaux and Marcon [11] that combines MILPs and a multi-agent system. Our DO also uses the principle of resource self-scheduling similar to Traub *et al.*'s [13]. This combination allows nurses in our DO to schedule their patients using a MILP similar to our CO, and interact with other nurses to exchange patients and find locally optimised scheduling of patients between agents.

3 Two (de-)centralised models of resource scheduling

Resource allocation, specifically nurse-to-patient scheduling, is one of the crucial decisions identified in EDs. It is a complicated decision because it needs to be made under several constraints, including limited resource availability and dynamic patients flowing into the ED. The patients' arrival time in ED is difficult to predict. Moreover, these arriving patients have different levels of criticality, which are characterised by the triage nurse on a scale from one to five, as shown in Table 1. On the other hand, the decision maker has to take the nurse-to-patient scheduling decision quickly to ensure that the patients receive high-quality care. Therefore, the optimisation of the resource scheduling decision is required to improve the performance of the ED and its patients' pathway.

Table 1: Triage levels of patients

Level	Status	Weight in our MILP	Colour in Figure 3
1	Resuscitation/acute	10^8	Red
2	Very urgent	10^6	Orange
3	Urgent	10^4	Yellow
4	Less urgent	10^2	Green
5	Not urgent	1	Blue

In order to optimise the scheduling in both normal and disturbed conditions, we propose two nurse-to-patient scheduling models with different (de-)centralisation levels, as briefly summarized in Figure 1.

3.1 Centralised model

The centralised model aims to optimise the nurse-to-patient scheduling decision globally by solving a dynamic version of a scheduling MILP. This model captures all patients' data and considers nurses' availability as input. Then, it solves the MILP presented in this subsection. This MILP of nurse allocation minimises the total waiting time of the patients as the objective function to generate the best scheduling. In order to cope with dynamic patient flow

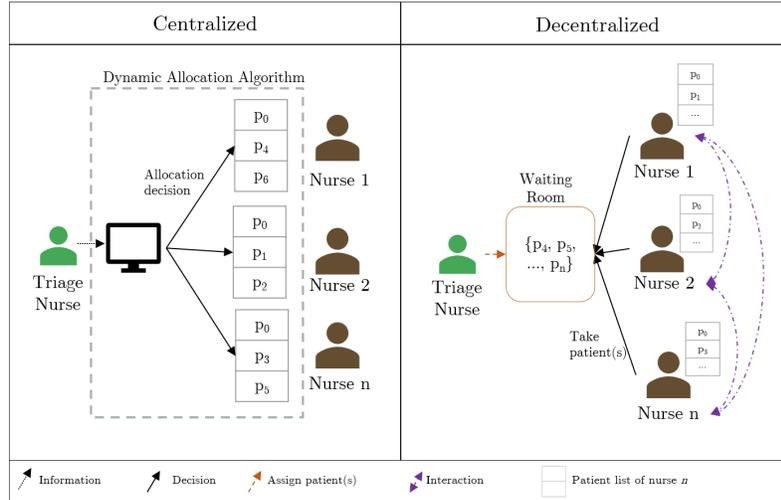


Fig. 1: Brief presentation of the (de-)centralisation models

in the ED, this MILP is re-optimised every time a new patient arrives. The patient schedule is adjusted dynamically with a constraint modelling the fact that the treatment process must not be stopped, that is, the patients currently receiving treatment cannot be rescheduled later.

The main part of the dynamic allocation algorithm is the MILP model of nurse scheduling adapted from [2, 10]. The followings are the mathematical model notations and equations (Eq. 1-13) for the dynamic nurse allocation MILP model:

Indices and sets:

- j patient $j = 0, 1, \dots, J$. Patient $j = 0$ is the "dummy" patient ($j \in J$)
- n nurse $n = 0, 1, \dots, N$ ($n \in N$)
- J Set of patients
- N Set of nurses

Decision variables:

- s_j Treatment start time of Patient j
- c_j Treatment completion time of Patient j
- x_{jn} 1 if Patient j treated by nurse n , 0 otherwise
- z_{jkn} 1 if Patient k treated immediately after Patient j by Nurse n , 0 otherwise

Parameters:

a_j	Arrival time of Patient j
p_j	Treatment duration of Patient j
u_j	Weight of triage level of Patient j (see Table 1)
zt_{jkn}	Treatment order decision variable of Patient j in the previous run of scheduling, <i>i.e.</i> , 1 if Patient k was treated immediately after Patient j by Nurse n in the previous run
ct_j	Completion time of Patient j in the previous run
t	Time of the current run of scheduling

$$\min \quad w = \sum_{j=1}^J u_j (s_j - a_j) \quad (1)$$

$$\text{s.t.:} \quad s_j + p_j = c_j \quad \forall j \in J \quad (2)$$

$$s_j \geq a_j \quad \forall j \in J \quad (3)$$

$$\sum_{n=0}^N x_{jn} = 1 \quad \forall j \in J \quad (4)$$

$$c_j - s_k \leq M(1 - z_{jkn}) \quad \forall j \in J, \forall k \in J, \forall n \in N \quad (5)$$

$$\sum_{j=0, j \neq k}^J z_{jkn} = x_{kn} \quad \forall k \in J, \forall n \in N \quad (6)$$

$$\sum_{k=1, j \neq k}^J z_{jkn} \leq x_{jn} \quad \forall j \in J, \forall n \in N \quad (7)$$

$$\sum_{k=0}^J z_{0kn} \geq 1 \quad \forall n \in N \quad (8)$$

$$z_{jkn} = zt_{jkn} \quad \forall j \in J^* = \{j | ct_j < t\}, \forall k \in J, \forall n \in N \quad (9)$$

$$s_j \geq 0 \quad \forall j \in J \quad (10)$$

$$c_j \geq 0 \quad \forall j \in J \quad (11)$$

$$x_{jn} \in \{0, 1\} \quad \forall j \in J, \forall n \in N \quad (12)$$

$$z_{jkn} \in \{0, 1\} \quad \forall j \in J, \forall k \in K, \forall n \in N \quad (13)$$

Eq. 1 is the objective function of the model that minimises the total weighted waiting time over all patients. The waiting time is acquired from the difference between the patient's arrival time a_j and treatment start time s_j . The constraint in Eq. 2 defines the completion time of each patient as the sum of the patients' treatment start time and treatment duration. Eq. 3 ensures that the treatment starts after the arrival time. Eq. 4 ensures that each patient is treated by only one nurse. Eq. 5 is the treatment sequence constraint. Eq. 6 sets that each patient has a direct predecessor and Eq. 7 ensures each

patient has no more than one direct successor in the patient sequence. Eq. 8 lets each nurse have only one dummy patient ($j=0$) so that each other patient has a direct predecessor. Eq. 9 sets the current z_{jkn} value to be equal to the value from the previous run (zt_{jkn}) if the completion time of the last run (ct_j) is larger than the current scheduling time t . In other words, Eq. 9 makes sure that all ongoing treatments are not stopped, that is, not rescheduled.

In every run, the dynamic allocation algorithm is triggered by the arrival of one patient at the triage nurse. The algorithm receives the arrival time and triage (urgency) level of the patient after the triage process was performed by the triage nurse. The algorithm sets the scheduling time t equal to the arrival time of this last patient. Next, it solves the allocation problem in Eq. 1-13 in order to generate the nurse-to-patient allocation. Then, the algorithm stores the completion time to variable ct_j and the patient order variable to zt_{jkn} for the next run. This schedule is used until the next patient arrives; such an arrival triggers a new run of this algorithm.

3.2 Decentralised model

The decentralised model is a multi-agent system in which each nurse is modelled as an agent who possesses local information and is able to make local decisions. Each nurse agent has the capability to interact with the other nurse agents according to the Contract Net Protocol (CNP). In a few words, our version of CNP is a decentralised auction in which the role of the auctioneer is played by a nurse, who is referred to as host in this article. This protocol is decentralised because the host may be different in every round of CNP. Scheduling has two subproblems, namely, allocating patients to nurses and sequencing the patients of every nurse, and the CNP solves the allocation subproblem. Every nurse saves the patients currently allocated to them in their variable *patientList* and the CNP manages the exchange of patients between nurses.

Our CNP has several interaction rounds in which each nurse agent becomes either a host or a guest. Each interaction round is executed after a nurse has finished treating a patient and must decide which patient to be treated next. This nurse becomes a host who takes the role of an auctioneer and broadcasts a patient to be proposed for exchange to all other nurses. The other nurses (guests) reply to the broadcast by proposing and sending one of their patients to be given to the host. The host calculates the allocation of the patients and sends the allocation results to the guest nurses. The next interaction round occurs when another nurse completes a treatment process and becomes the host of this round. This model also involves a triage nurse agent whose role is to generate patient information and send the patient to the waiting room.

We now detail our implementation of CNP by explaining the three state charts in Figure 2. The number in the names of the states and transitions in the host and guest state charts indicates their order of activation,

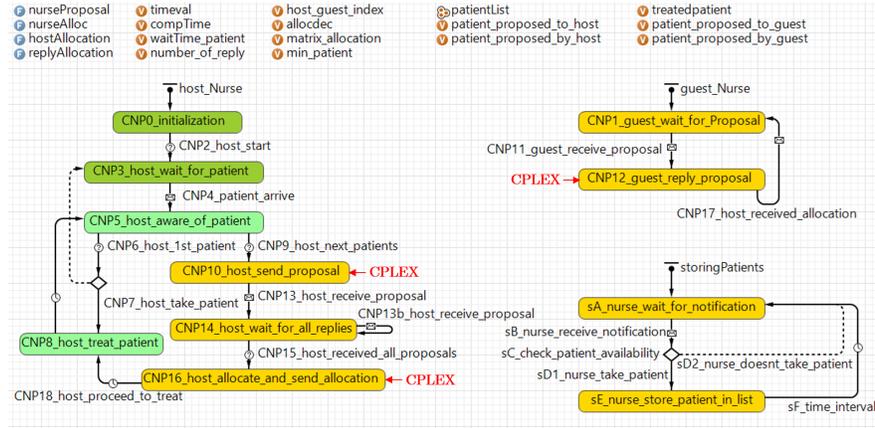


Fig. 2: Decentralised model

e.g., the message sent in State `CNP10_host_send_proposal` fires Transition `CNP11_guest_receive_proposal`. Initially, all nurses start in State `CNP0_initialization` for the host where they put the dummy patient ($j = 0$) into their *patientList*. All nurses are also in State `CNP1_guest_wait_for_Proposal` of their guest state chart and State `sA_wait_for_notification` for the *storingPatient* state chart because we use Anylogic which always executes the first state in all state charts. Then, Transition `CNP2_host_start` in the host state chart fires when decentralisation is chosen as the mode of decision. The host nurses are waiting for a patient from the guest in their State `CNP3_host_wait_for_patient`. After the triage nurse has generated the patient's information and sent this patient to the waiting room, the triage nurse sends a message to all nurses to trigger their Transition `CNP4_patient_arrive`. The nurses are now in State `CNP5_host_aware_of_patient`. All nurses then fire Transition `CNP6_host_1st_patient` if they are going to treat their first patient. The first nurse to fire Transition `CNP7_host_take_patient` takes and removes the patient from the waiting room, while the other nurses go back to State `CNP3_host_wait_for_patient` as there are no patients in the waiting room. The nurse who takes the patient continues to state `CNP8_host_treat_patient` to treat the patient during p_j .

As the first nurse host treated the first patient, other patients arrive and other nurses are aware, fire transitions, take, and treat their first patients. After the nurses acquire their first patients, if new patients arrive, the triage nurse sends different messages which trigger Transition `sB_nurse_receive_notifications` in the nurse system. The nurses check the waiting room in branch `sC_check_patient_availability`. The first nurse who fires Transition `sD1_nurse_take_patient`, takes and keeps the patient on the *patientList* using the system. The nurse who has just taken the patient waits in `sF_time_interval` to prevent the same nurse from taking all newly arrived patients.

After one of the nurses finishes treating her first patient, this nurse gets back to state `CNP5_host_aware_of_patient`. This nurse fires Transition `CNP9_host_next_patients` because now the nurse is going to treat the next patients. Instead of treating directly the patient in the *patientList*, the nurse performs scheduling and interaction with other nurses to find and treat the patient that optimizes the local weighted waiting time. In order to do this, first, the nurse who is in State `CNP10_host_send_proposal` uses the solver CPLEX to locally solve a modified version of the above MILP. More precisely, this MILP is the same class of model with Eq. 1-13, except that there are some adjustments explained now. The nurse n index, decision variable x_{jn} , and some constraints (Eq. 8 and 9) are removed. A new binary decision variable $kept_j$ is added to find the patient with the largest weighted waiting time to be proposed. The variable $kept_j$ also replaces the x_{jn} in Eq. 6 and 7 to ensure that each patient, except the one proposed, has one direct predecessor and has no more than one successor. After solving the patient proposal problem, the host nurse broadcasts the obtained patient to other nurses who become guests.

This broadcast fires all guests' Transition `CNP11_guest_receive_proposal`. The guest also solves the modified MILP explained in the previous paragraph to find the patient to be proposed back to the host in State `CNP12_guest_reply_proposal`. The guest also calculates the weighted waiting time for several cases:

- Case 1: All the patients in the *patientList* without the host's proposal
- Case 2: All the patients in the *patientList* with the host's proposed patient
- Case 3: All the patients in the *patientList*, including the host's proposal, and excluding the patients to be proposed

The calculation is done by making CPLEX solve another version of the MILP in Eq. 1-13 with the following adjustments. The nurse n index, decision variable x_{jn} , and some constraints (Eq. 8 and 9) are removed. After the calculation is performed, these weighted waiting times and the proposed patient are sent back to the host.

The host nurse fires Transition `CNP13_host_receive_proposal` after receiving the first reply from a guest and waits for the replies of the other guests in State `CNP14_host_wait_for_all_replies`. Meanwhile, the host also saves the waiting time received in these replies in a matrix for allocation. This matrix for allocation contains the weighted waiting time of all scheduling cases from the nurses involved in the exchange. It is used later for allocating the patient bundle to the respective nurse. After that, the host nurse fires Transition `CNP15_host_received_all_proposals` and executes the allocation in state `CNP16_host_allocate_and_send_allocation`. First, the host nurse calculates the weighted waiting time for all the following cases:

- Case 1: All the patients in the *patientList* without the guests' proposals
- Case 2: Patients in the *patientList* of host with each of guest's proposed patient

Case 3: All the patients in the *patientList*, including each of the guest's proposal, and excluding the patients proposed to host

The host then inserts these weighted waiting times into the matrix for allocation. The host then solves the allocation problem with CPLEX and sends the result to the guests. The message from the host triggers Transition `CNP16_host_receive_allocation`, which allows the guest to update its *patientList* based on the allocation result. The host also modifies the *patientList* based on the result and determines the next patients to be treated. After that, the host fires Transition `CNP17_host_proceed_to_treat` and begins to treat the next patient in State `CNP7_host_treat_patient`. The next rounds of scheduling and interaction take place after another nurse finishes the treatment of a patient. In the meantime, the nurses still run the system state chart to add patients to their *patientList* whenever the triage nurse sends another.

4 Experimental results and discussion

This section compares the schedule obtained by our (de-)centralised models on the same test set. The test aims to assess whether the proposed models generate appropriate and expected results. The models are assessed from their objective function values of total weighted waiting time and by seeing the patient schedule generated. The test set contains the data of 15 patients with various triage levels and interarrival times, as well as three nurses (plus the triage nurse). Both proposed models are built in AnyLogic 8.5.2 and IBM ILOG CPLEX 22.1.1.

The latest re-scheduling generated by each model at time $t = 55$ is shown in the Gantt charts in Figure 3. The x-axis is the time and the bold vertical line shows the current time point $t = 55$. This is the time when the rescheduling happens in the decentralised model. The value on each block represents the patient's index $j=1..15$. The colour on each patient block represents the triage level of the corresponding patient, as described in Table 1. The solid colour patient blocks mean that these patients' treatment is either finished or ongoing at scheduling time t , hence these patients are not involved in the rescheduling process. On the contrary, the shaded blocks mean that these patients' treatment has not started yet and may thus be rescheduled.

From the scheduling result in Figure 3, it can be seen that both models generate proper schedules for the patients. In general, the three patients (Patient $j = 1, 2, 3$) who arrived first are scheduled first to the three nurses. Then, the schedules prioritize all the patients with triage level one (patients $j = 1, 4, 7, 8, 11$). After that, they are followed by patients of triage levels two to five, respectively. The patients are distributed almost evenly among the three nurses.

The schedules generated for centralised and decentralised models are slightly different. The differences are in the patient whose treatment is not

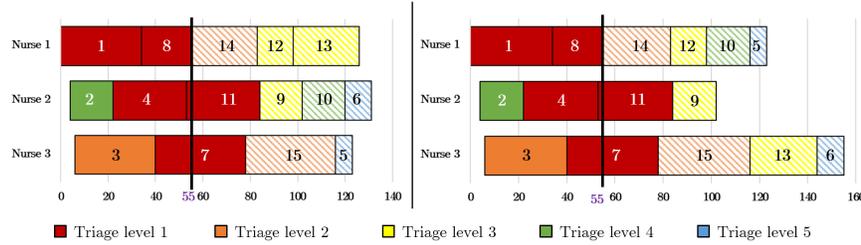


Fig. 3: Generated Schedule of Centralised and Decentralised Model

started yet. On the other hand, both schedules generated for all the patients whose treatment is finished or ongoing at time t are the same. The differences in the schedules occur as the result of the interaction and patient exchange that happened in the decentralised model. In spite of the differences in the patient sequence generated in the schedules, both schedules produced similar objective functions. The objective value difference between the two models is relatively small, less than 0.1%.

5 Conclusion

The exploration of different (de-)centralisation levels of decision structure in resource scheduling for Emergency Departments (EDs) is crucial to manage and improve ED performance and patients' pathways. This article has proposed and compared one decentralised and one centralised structure. The centralised model has a dynamic allocation algorithm relying on a MILP that minimises the total weighted waiting time of patients. The decentralised model is a multi-agent system in which each nurse is an agent solving a variant of the centralised MILP to generate her own schedule, and interacting with other nurses to exchange patients. Both models are assessed and compared using the same test sets, and the result shows that both models generate appropriate patient schedules with slight differences. The difference in objective function generated from both models is also relatively small (less than 0.1%). Overall, the centralised model provides a slightly better solution compared to the decentralised one.

This paper explores two different decision structures on resource scheduling and compares them to a small-sized test set. In future works, both models will be modified in order to generate each reschedule in five steps: patients with triage level 1 will be scheduled in Step 1, next patients with triage level 2 will be added without modifying the patients already scheduled in Step 2, and so on. This change may solve the problem described above on the centralised model by removing the use of weight to model the triage level. In addition, this will allow for scheduling more patients since fewer patients will be scheduled in each of the 5 steps. Besides, we will test both our models on real datasets from the ED in Saint-Étienne, France. The exploration and

comparison of the developed models with the real patient-taking process done in the ED will also be performed.

Acknowledgement

This work contributes to JUNEAU project that is funded by ANR (Agence Nationale de la Recherche – France) “*ANR-22-CE46-0010-01*”.

References

1. Bouleux, G., El Haouzi, H.B., Cheutet, V., Demesure, G., Derigent, W., Moyaux, T., Trilling, L.: Requirements for a digital twin for an emergency department. In: Proceedings of SOHOMA 2022, pp. 130–141. Springer (2023)
2. Cao, D., Chen, M., Wan, G.: Parallel machine selection and job scheduling to minimize machine cost and job tardiness. *Computers & operations research* **32**(8), 1995–2012 (2005)
3. Cardin, O., Trentesaux, D., Thomas, A., Castagna, P., Berger, T., Bril El-Haouzi, H.: Coupling predictive scheduling and reactive control in manufacturing hybrid control architectures: state of the art and future challenges. *Journal of Intelligent Manufacturing* **28**(7), 1503–1517 (2017)
4. Chaabane, S., Kadri, F.: Toward a proactive and reactive simulation-based emergency department control system to cope with strain situations. *Operations Research and Simulation in Healthcare* pp. 123–152 (2021)
5. Das, M., Jana, D.K., Alam, S.: Comparative study of centralized and decentralized scenarios of a three-tiered green supply chain in two-period using the game theoretical approach. *Cleaner Logistics and Supply Chain* **4**, 100,054 (2022)
6. Davidsson, P., Persson, J.A., Holmgren, J.: On the integration of agent-based and mathematical optimization techniques. In: Proceedings of KES-AMSTA 2007, pp. 1–10. Springer (2007)
7. Duma, D., Aringhieri, R.: Real-time resource allocation in the emergency department: A case study. *Omega* **117**, 102,844 (2023)
8. Harzi, M., Condotta, J.F., Nouaouri, I., Krichen, S.: Scheduling patients in emergency department by considering material resources. *Procedia computer science* **112**, 713–722 (2017)
9. He, S., Sim, M., Zhang, M.: Data-driven patient scheduling in emergency departments: A hybrid robust-stochastic approach. *Management Science* **65**(9), 4123–4140 (2019)
10. Luscombe, R., Kozan, E.: Dynamic resource allocation to improve emergency department efficiency in real time. *European Journal of Operational Research* **255**(2), 593–603 (2016)
11. Moyaux, T., Marcon, E.: Cost of selfishness in the allocation of cities in the multiple travelling salesmen problem. *Engineering Applications of Artificial Intelligence* **89**, 103,429 (2020)
12. Pach, C., Berger, T., Bonte, T., Trentesaux, D.: Orca-fms: a dynamic architecture for the optimized and reactive control of flexible manufacturing scheduling. *Computers in Industry* **65**(4), 706–720 (2014)
13. Traub, S.J., Stewart, C.F., Didehban, R., Bartley, A.C., Saghafian, S., Smith, V.D., Silvers, S.M., LeCheminant, R., Lipinski, C.A.: Emergency department rotational patient assignment. *Annals of emergency medicine* **67**(2), 206–215 (2016)