Scheduling medical tests: a solution to the problem of overcrowding in a hospital emergency department

Di Lin[1], Fabrice Labeau[1], Xidong Zhang[2], GuiXia Kang[2]
[1]Department of electrical and computer engineering, McGill University, Canada
[2]Key Laboratory of Universal Wireless Communications (Beijing University of Posts and Telecommunications), Ministry of Education, China

***Abstract*—To improve the performance of hospital emergency department (ED), we propose a scheduling strategy to reduce the total time of medical tests in ED for a given number of patients. We model the schedulig strategy as a constraint logic programming problem, and then solve this problem with an open source software: Minizinc. Also we compare the medical test time with the scheduling strategy and that without scheduling, which is viewed as a benchmark for our proposed scheduling strategy. The results show that the scheduling strategy can save $12\%-25\%$ of medical test time.**

***Index Terms*—Scheduling medical tests, Emergency department, Jobshop problem, Constraint logic programming, Operational research**

I. Introduction

Overcrowding in emergency department (ED) is a worldwide problem [1]–[3], and it refers to the situation in which the ability to offer emergency care cannot meet the demand for emergency services within a reasonable time [4]. Overcrowding in ED will affect patient satisfaction, and even worse, it will increase the risk of death and hospital readmission for patients who have been discharged from the emergency department [5]. By observing more than 20 million patient visits to emergency departments over five years, Guttmann et al. in [5] presented that the risk of death and hospital readmission increases with the degree of crowding in the emergency department, and estimated that about 150 fewer patients would die each year if the average waiting time to access emergency department was less than an hour. Thus, how to reduce the waiting time to access the emergency department has been an important research issue, and we will discuss this issue in this paper.

By observing the process of patients from the main entrance, to the greeter, triage and registration into the ED [6], we summarize the flow of patients in ED as model (shown in Fig.1.), which includes two parts: the emergency department (ED) and the inpatient units (IU). Specifically, arriving patients include both the Walk-in patients and the Ambulance-in patients. After entering into the main entrance of a hospital, the patient starts the visit with the greeter desk, then triage, registration, and waiting to access ED. In our model these steps are merged in the unit of Patients waiting to access ED. Then a patient will access the ED and be allocated a bed by a nurse. After a pre-examination by the ED nurse, the patient will see a physician for further examination, test, and treatments. If the physician determines to release the patient after examinition, an ED nurse facilitates the process of releasing patients, who will then depart the ED. Otherwise, if the physician decides to transfer the patient to the IU, a consulting physician from the IU will arrange the admission of this patient. Depending on the availablity of IU beds, the patient is either transferred into the IU immediately or has to wait in the ED for an IU bed to be available. These steps in our model are merged in the unit of 'Patients staying in ED'. A patient in the IU will receive tests and treatments until he/she reachs the medical requirement to be released. After the final examination by a physician, the patient will depart the IU by completing the checkout process.These steps in our model are merged in the unit of 'Patients staying in IU'. Finally, the patient will be released from the hospital, and this step is characterized as 'Patients release'.

As shown in Fig.1, the waiting time to access ED is mainly determined by two issues: (1) the emergency service time in ED, and (2) the length of waiting time of patients to be transferred into IU. Thus, reducing the length of waiting time in ED is equivalent to reducing the emergency service time in ED and the length of waiting time of patients to be transferred into IU. In this paper, we focus on the first issue, namely, how to save emergency time in ED. The emergency service time in ED mainly includes the time for tests (80% of total service time) and the time for treatment (20% of total service time). In the following, we focus on how to reduce the average test time for patients, not only because the time for test takes a proportion of 80%, but because little time for treatment can be reduced to guarantee the quality of treatment. More specifically, we focus on how to reduce the unnecessary time for tests in ED, such as the waiting time to take a test for a patient. From the methodology perspective, we will use scheduling to reduce the length of testing time for a given number of patients. The rational is that patients can take common tests in ED, including Urine sample, Blood test, Electrocardiogram (ECG), X-ray, Computed Tomography scan, Magnetic Resonance Image (MRI), in a flexible sequence, and we can save test time by scheduling different testing sequences for different patients to fully utilize the test equipments.

This paper consists of four sections. Section I introduces patient flow to access and depart ED of a hospital. Section II

E-mail address: di.lin2@mail.mcgill.ca, fabrice.labeau@mcgill.ca, chang.storm@gmail.com, gxkang@bupt.edu.cn

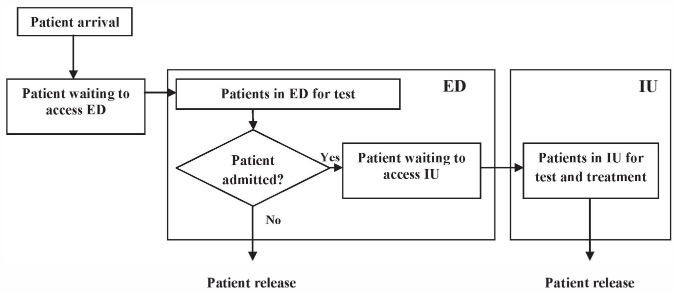

Fig. 1. Patient flow to access and depart ED [6].

ED test	Function
US	Find urinary infections and screen for metabolic conditions
BT	Identify infections, analyze blood clots, analyze blood clots, and determine electrolyte composition of bodily fluids
ECG	Record the electrical activity of the heart
XR	Diagnose lung and heart disease, pneumonia, heart failure, broken bones, degenerative joint disease, and internal trauma
CTS	Diagnose bone tumors, cancers, and fractures, as well as detect internal injuries and bleeding
MRI	Find tumors, bleeding, blood vessel diseases, infections, and bone and organ damage

TABLE II
SIX TYPES OF COMMON ED TESTS [7]

Emergency level	Triage	Condition
I	Resuscitation	Threats to life or limb
II	Emergent	Potential threat to life limb or function
III	Urgent	Progress to a serious problem
IV	Less Urgent	Potential progress to a serious problem
V	Non Urgent	Could be delayed or even released

TABLE I
EMERGENCY LEVELS IN TRIAGE [7]

focuses on the mathematical model and the logic programming model for scheduling ED tests. Section III presents the simulation results to show that the scheduling strategy outperforms the strategy without scheduling. Section IV concludes this paper.

II. SCHEDULING TESTS IN ED

In this section, we will first present the process of triage and the process of pre-treatment and tests in ED. Then, building on these processes, we will utilize the logic programming language to characterize the model of scheduling various medical tests in ED.

A. Process of triage and tests

As shown in Fig.1, the triage is usually completed by a triage nurse, who must be able to explore enough critical information to determine patient acuity and any immediate care needs by interviewing the patients with in a 2-5 minutes period. Building on the vital signs and pain scales, the patients are classified into a few categories, each of which corresponds to a level of emergency. Taking the Canadian Triage and Acuity Scale (CTAS) as an example, the patients' emergency levels include Resuscitation, Emergent, Urgent, Less Urgent, Non Urgent (the detailed classification is shown in TABLE I).

Patients in level I or level II, namely, the patients who are at risk of losing their life, should receive immediate initial treatment. If immediate treatment is successful, the patient will improve (although this may be temporary) and this improvement may allow categorizing this patient into a lower priority in the short term. Triage should be continuous by checking categories regularly to ensure the correctness of priority. For patients in lower priority (level III or level IV or level V), they will be arranged to receive medical tests in ED. Thus, in the following model for scheduling tests, we assume three priorities of patients, including high priority (HP), medium priority (MP), and low priority (LP).

B. Model for scheduling tests in ED

For three priorities of patients, including high priority (HP), medium priority (MP), and low priority (LP), we assume that they will receive any combination of six types of independent ED tests, including Urine sample (US), Blood test (BT), Electrocardiogram (ECG), X-ray (XR), Computed tomography scan (CTS), Magnetic resonance image (MRI) (The functions of these ED tests are summarized in TABLE II). We assume that the triage nurse will schedule the tests for patients regularly, so the number of patients is known within a regular time frame. Also, we assume the number of resources for each type of test is 1, and the test resourcse includes test equipment and staff.

1) Ranking patients to receive ED tests: For arrivals of patients to receive tests, the sequence is determined by two factors: (1) the priority of patients, and (2) the arrival time of patients. The general rule for ranking patients is: (1) the patients in higher priority can take ED tests before those in lower priority; (2) Among the patients in the same priority, those who arrive first will take ED tests first. For any two patients i and j, we use $Seq_i < Seq_j$ to represent that patient i has higher rank than patient j, so patient i should finish all the tests before patient j.

2) Mathematical model for scheduling ED tests: Given N_P patients, N_D types of tests, as well as the sequence of patients Seq_i $(i = 1, 2, ..., N_P)$, we characterize the problem to minimize the test time End as

$$\begin{aligned} &Min \quad End \\ &\qquad s.t. \; S_{ik} + T_{ik} \leq S_{ij} \\ &\qquad\quad (j \neq k, j, k = 1, 2, ..., N_D, i = 1, 2, ..., N_P) \\ &\qquad\quad S_{ik} + T_{ik} \leq end_i \\ &\qquad\quad (i = 1, 2, ..., N_P, k = 1, 2, ..., N_D) \\ &\qquad\quad end_i \leq end_j \; if \; Seq_i > Seq_j \\ &\qquad\quad (i, j = 1, 2, ..., N_P) \\ &\qquad\quad S_{ik} + T_{ik} \leq S_{jk} \; and \; S_{jk} + T_{jk} \leq S_{ik} \\ &\qquad\quad (i \neq j, i, j = 1, 2, ..., N_P, k = 1, 2, ..., N_D) \\ &\qquad\quad end_i \leq End \; (i = 1, 2, ..., N_P) \end{aligned} \tag{1}$$

where S_{ik} denotes the starting time of patient i to take test k, T_{ik} denotes the duration of test k for patient i, end_i denotes the ending time of all tests for patient i.

The first constraint ensures that one patient can take at most one test at the same time. The second constraint and the third constraint ensure that the patients in higher rank should finish all the tests earlier than those in lower rank. The fouth constraint is due to the limited number of test resources for one particular type of test. In this paper, we assume the number of each test resources is 1, so the constraint on the limitation of test resources equals the constraint that no more than one patient is taking a certain test at the same time. The fifth constraint defines the ending time of all tests for all patients.

3) Logic programming model for scheduling ED tests: The optimization problem characterized in equation (1) is a typical constraint logic programming (CLP) problem, and more specifically, is a extension of the regular job-shop scheduling problem. In our problem, we extend the regular job-shop scheduling problem by taking into account the priority of patients. In the following, to characterize this CLP problem, we will use an off-the-shelf CLP language: Minizinc. Minizinc is an open source CLP modelling language, and it is a medium-level language. It is high-level enough to express most CLP problems easily and in a largely solver-independent way, and it is also low-level enough to be easily mapped onto many solvers.

A Minizinc program is composed of both a model (.mzn file) to characterize the optimization problem and a dataset (.dzn file) to assign the values of variables in the model. For our problem, we can describe the Minizinc program as

```
int: size 1;
int: size 2;
% size of problem
array [1.. size 1, 1..size 2] of int: d;
% task durations
int: total = sum (i in 1..size1, j in 1.. size2)(d[i,j]);
% total duration
array[1..size1, 1..size2] of var 0..total: s;
% start times
var 0..total: end;
% total end time
Predicate nooverlap (varint: s1, int: d1, varint: s2, int: d2):-
s1+d1<=s2 ∧ s2+d2<=s1;
constraint
for all (i in 1..size1)(
for all (j,k in 1..size2 where j<k)
(noverlap (s[i,j],d[i,j], s[i,k], d[i,k]))
∧ for all (j in 1..size2) (s[i,j]+d[i,j] <= end));
constraint
for all (i in 1..size2)(
for all (j,k in 1..size1 where j<k)(
nooverlap(s[j,i], d[j,i], s[k,i], d[k,i])));
solve minimize end;
```

To show the efficiency of scheduling, we compare the total time of medical test with scheduling and that without scheduling, and the latter can be viewed as a benchmark. The Minizinc code for the latter is similar to that for the former, except using 'solve satisfy' to replace 'solve minimize'. The solver with 'solve satisfy' uses an incomplete searching strategy, namely, it will terminate once the first feasible solution is found. Thus, we use this strategy without scheduling as a benchmark to clarify the benefit obtained by scheduling, and the Minizinc code is:

```
int: size 1;
int: size 2;
% size of problem
array [1.. size 1, 1..size 2] of int: d;
% task durations
int: total = sum (i in 1..size1, j in 1.. size2)(d[i,j]);
% total duration
array[1..size1, 1..size2] of var 0..total: s;
% start times
var 0..total: end;
% total end time
Predicate nooverlap (varint: s1, int: d1, varint: s2, int: d2):-
s1+d1<=s2 ∧ s2+d2<=s1;
constraint
for all (i in 1..size1)(
for all (j,k in 1..size2 where j<k)
(noverlap (s[i,j],d[i,j], s[i,k], d[i,k]))
∧ for all (j in 1..size2) (s[i,j]+d[i,j] <= end));
constraint
for all (i in 1..size2)(
for all (j,k in 1..size1 where j<k)(
nooverlap(s[j,i], d[j,i], s[k,i], d[k,i])));
solve satisfy;
```

ED test	US	BT	ECG	XR	CTS	MRI
Length	5 min	5 min	20 min	30 min	30 min	45 min

TABLE III
LENGTH OF ED TESTS [7]

III. Simulation and Results

The simulation parameters are as follows: each patient needs to take 6 tests, and the length of each test is shown in TABLE III. Without loss of generality, we assume patient i has a higher rank than patient j if $i < j$, namely, patient i will finish tests before patient j.

A. Process of scheduling

Assuming that the number of patients is 5 and that the number of resources across tests is 1, we clarify the process of scheduling by 2 cases: (1) the patients will take the tests of XR, CTS, MRI, in the presence of scheduling strategy; (2) the patients will take the tests of XR, CTS, MRI without using the scheduling strategy. The time distribution of each test for each patient in case 1 is shown in TABLE IV, and that in case 2 is shown in TABLE V. By comparing the result of case 1 and that of case 2, we find that scheduling can reduce the total time for tests from 285 min to 225 min, and the reason is that in case 1 the resources are fully utilized by scheduling the different sequences of tests for different patients. Also, TABLE IV and TABLE V show that each test will be taken by a patient at the beginning in the presence of scheduling strategy, and a few test resources are idle at the beginning if we do not use the scheduling strategy, such as the period of 0-30 for CTS test and the period of 0-60 for MRI test.

Patient ID	XR	CTS	MRI
Patient 1	60-90	90-120	0-45
Patient 2	0-30	30-60	90-135
Patient 3	120-150	0-30	45-90
Patient 4	30-60	60-90	135-180
Patient 5	90-120	120-150	180-225

TABLE IV
TIME DISTRIBUTION FOR CASE 1

Patient ID	XR	CTS	MRI
Patient 1	0-30	30-60	60-105
Patient 2	30-60	60-90	105-150
Patient 3	60-90	90-120	150-195
Patient 4	90-120	120-150	195-240
Patient 5	120-150	150-180	240-285

TABLE V
TIME DISTRIBUTION FOR CASE 2

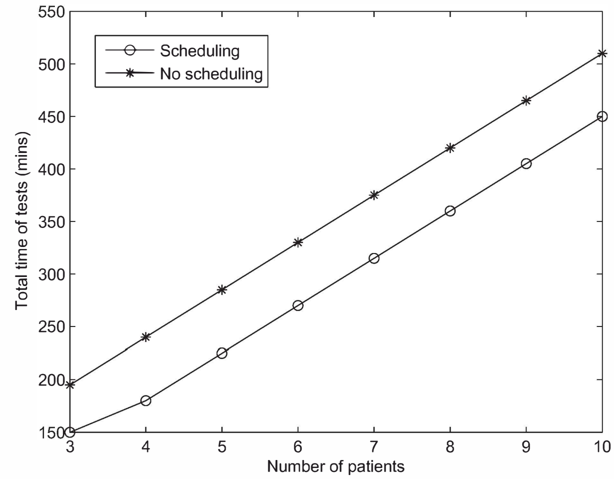

Fig. 2. Total time of tests with scheduling Vs. without scheduling.

B. Benifit of scheduling

In this section, we focus on the benefit of scheduling by comparing the total time for tests with scheduling and that without scheduling in the presence of various number of patients. The result of total time for tests is shown in Fig.2, and the gap of test time between with scheduling and without scheduling is shown in Fig.3. Fig.2 and Fig.3 show that the scheduling strategy can reduce the total time for taking tests across various numbers of patients, and the scheduling strategy can obtain a benefit of saving $12\% - 25\%$ test time than the strategy without using scheduling. Additionally, Fig.3 shows that the gap of test time between the strategy with scheduling and that wihout scheduling increases before the number of patients reach 4 and then decreases afterwards; a peak can be obtained when the number of patients equals 4. The reason is that test time with strategy without scheduling will increase at a high rate with the number of patients below 4, and this increase rate drops when the number of patients exceed 4 (shown in Fig.2).

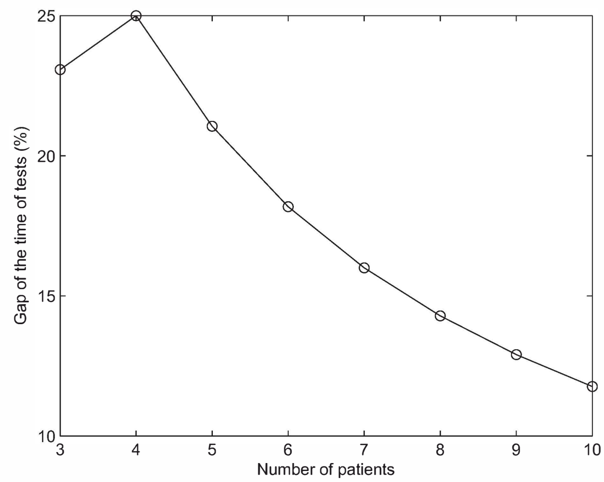

Fig. 3. Gap of the time of tests with scheduling Vs. without scheduling.

The benefit of using scheduling strategy can be explained from the perspective of idle time of test resources. The idle time of a test is defined as the length of time during which the resources for this test are not occupied, and the total idle time is defined as the summation of the idle time of each test. A strategy that can reduce the idle time of resources will save the test time. Fig.4 shows that scheduling strategy can save around 60 mins of idle time than the strategy without scheduling, and this result can explain why scheduling strategy outperforms the strategy without scheduling. Additionally, Fig.4 shows that the idle time for MRI test is always lower than the idle time for other tests, and the reason is that taking a MRI test will take 45 mins, which is longer than the time for other tests (30mins). This result shows that the scheduling strategy will attempt to reduce the idle time of a test which is the most time-consuming (45 mins for MRI in this case).

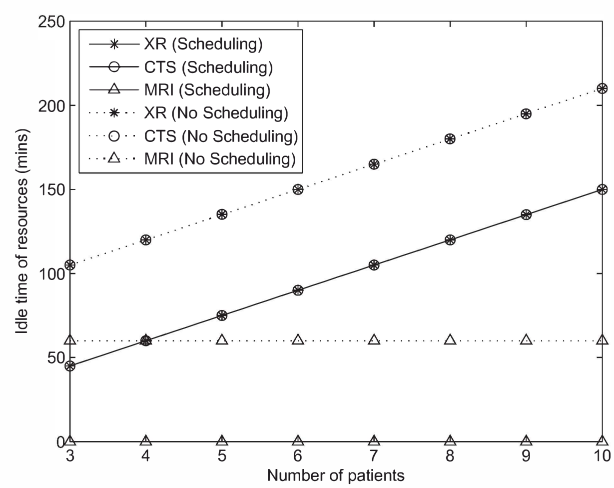

Fig. 4. Idle time of resources with scheduling Vs. without scheduling.

IV. CONCLUSIONS

In this paper, we propose a scheduling strategy to minimize the total time of taking tests in ED for a given number of patients. By modeling the optimization problem in a logic programming model, we solve this problem with an open source software Minizinc, which is designed for logic programming problems. Additionally, we also estimate the total time of tests without scheduling and view this time as a benchmark for our proposed scheduling strategy. The results show that the total time with scheduling strategy can save $12\% - 25\%$. The scheduling strategy as well as the logic programming can help develop a computer based decision support system for ED healthcare, and this system can assist triage nurses to reasonably arrange the medical tests for patients to improve performance in a hospital ED.

The problem of scheduling tests is an extension of a job-shop problem, which is a NP-hard problem, and Minizinc uses the greedy algorithm, which cannot solve the scheduling problem in polynomial time. Thus, in this paper we only show the number of patients up to 10. For the results with a large number of patients (100-1000), we need to develop an approximation algorithm which can solve the NP-hard problem in polynomial time. However, the discussion on the approximation algorithm has been out of the scope of this paper, and it will be out future work.

Acknowledgment

This work was partially supported by the Natural Sciences and Engineering Research Council (NSERC) and industrial and government partners, through the Healthcare Support through Information Technology Enhancements (hSITE) Strategic Research Network, was partially supported by Quebec MDEIE PSR-SiiRi program, was partially supported by National High Tech. Research and Development of China (No. 2009AA02Z412), and was partially supported by International technology cooperation project (No. 2010DFA11590).

References

[1] Kolb, E. M. W., J. Peck, et al. (2008). Reducing Emergency Department overcrowding - five patient buffer concepts in comparison. Simulation Conference, 2008. WSC 2008. Winter.

[2] Haifeng, X., T. J. Chaussalet, et al. (2006). A Model-Based Approach to the Analysis of Patterns of Length of Stay in Institutional Long-Term Care. Information Technology in Biomedicine, IEEE Transactions on 10(3): 512-518.

[3] Akcali, E., M. J Cote, et al. (2006). A network flow approach to optimizing hospital bed capacity decisions. Health Care Management Science 9(4): 391-404.

[4] Rowe BH, Bond K, Ospina MB, Blitz S, Friesen C, Schull M, Innes G, Afilalo M, Bullard M, Campbell SG, Curry G, Holroyd B, Yoon P, Sinclair D. (2006). Emergency department overcrowding in Canada: what are the issues and what can be done? [Technology overview no 21]. Ottawa: Canadian Agency for Drugs and Technologies in Health; 2006.

[5] Guttmann, A., Schull, M.J., Vermeulen, M.J., and Stukel, T.A. (2011). Association between waiting times and short term mortality and hospital admission after departure from emergency department: population based cohort study from Ontario, Canada. British Medical Journal. Retrieved from http://www.bmj.com/content/342/bmj.d2983.abstract.

[6] Kolb, E. M. W., L. Taesik, et al. (2007). Effect of coupling between emergency department and inpatient unit on the overcrowding in emergency department. Simulation Conference, 2007 Winter.

[7] Beveridge R, Ducharme J, Janes L, Beaulieu S, Walter S. Reliability of the Canadian emergency department triage and acuity scale: interrater agreement. Ann Emerg Med. 34:155159.