Scheduling of single operating room with a shared place for the patient recovery and induction

Roberto Bargetto and Federico Della Croce

1 Problem description and formulation

A single operating room (OR) is available for the execution of a set of surgeries, each one of those corresponding to a patient, let I denote the set of surgeries/patients. Every patient $i \in I$ requires anesthesia before entering the OR to go under surgery and a post-surgery observation (awakening) time. One equipped place adjacent to the OR is available for the patient anesthesia and post-surgery observation, let this place be the anesthesia/observation place (AOP). We assume that the OR and the AOP compose the operating theater (OT), other rooms as warehouses, corridors, etc., are not relevant for the problem. We also assume that stretchers, operating tables and the personnel of the OT never constrain the flow of patients through the OT. Each surgery $i \in I$ is thus characterized by the three tasks to perform in the same sequence for every surgery, these three tasks are: (1) anesthesia, (2) surgery and (3) awakening/observation.

Let p_{1i}, p_{2i} and p_{3i} denote the times of the three tasks respectively, *i.e.*, the anesthesia time, the surgery time and the awakening/observation time. The anesthesia time p_{1i} starts with the patient accommodation into the AOP and completes with the patient departure from the AOP to enter the OR. The surgery time p_{2i} starts with the patient entering the OR and completes with the patient being ready to leave the OR. The awakening/observation time p_{3i} starts immediately after the surgery time completion and completes with the patient leaving the OT.

No delay time is admitted between the anesthesia completion and the patient OR admission, the patient must enter the OR immediately after the anesthesia is completed. The patient must spend the awakening/observation time in the AOP or inside the OR. If the patient awakening/observation starts in the AOP, it must complete in the AOP. If the patient awakening/observation starts in the OR, it must complete in the OR.

If we assume given the OR surgery sequence σ, two decisions are necessary to obtain a complete schedule for the OT. These two decisions are:

1. deciding for each patient if the awakening/observation takes place in the AOP or in the OR and

2. deciding for each patient if the awakening/observation is preceded in the AOP by the anesthesia of the next surgery of the OR sequence σ.

If we aim to complete the all burden of surgeries I as much as early as possible, we have to minimize the awakening/observation completion time of the last surgery in the sequence σ (*i.e.*, minimize the makespan). Considering the two decisions 1 and 2, even if it involves the OR, the problem mainly corresponds to the scheduling of the AOP. Let thus a MILP formulation for a such a scheduling problem be the a MILP for the AOP scheduling problem (AOPSP). The MILP is report in the next section, Section 1.1. The MILP formulation of the integrated problem for computing also the OR surgery sequence, *i.e.*, the integrated AOP and OR scheduling problem (IAOPORPS), is in Section 1.2.

1.1 A MILP for the AOPSP

Let x_j be a binary variable that takes value 1 if the awakening/observation of the j-th patient of σ takes place in the OR; 0 otherwise. Let $y_{j,j-1}$ be a binary variable that takes value 1 if the j-th patient anesthesia precedes in the AOP the awakening/observation of the $(j-1)$-th patient; 0 otherwise. Variables C_{1j}, C_{2j} and C_{3j} are, respectively, the anesthesia completion time, the surgery completion time and the awakening/observation completion time; namely, the completion times of the tasks 1, 2 and 3 of the j-th patient. More generally, let C_{tj} be the completion time of the task $t = 1, 2, 3$ for the j-th patient/surgery. Since the surgery sequence is given, the position j corresponds to a given surgery i and the times of the three OT tasks for a surgery (anesthesia, surgery and awakening/observation) can be expressed using the positional index j; *i.e.* p_{1j}, p_{2j} and p_{3j}. The MILP reads:

$$f_{obj} := \min\{C_{3n}\} \tag{1}$$

Awakening of last patient ~

s.t.

Beginning of surgery = ending of Anestesia

$$C_{2j} - p_{2j} = C_{1j} \qquad \forall j = 1, ..., n \tag{2}$$

Beginning of Awakening = ending of surgery

$$C_{3j} - p3j = C_{2j} \qquad \forall j = 1, ..., n \tag{3}$$

Beginning of surgery J after ending of surgery J-1

$$C_{2j} - p_{2j} \geq C_{2,j-1} \qquad \forall j = 2, ..., n \tag{4}$$

Beginning of surgery J after completion of Aw J-1
IF Aw IN OR -> X=1 -> M~0
IF Aw in AOP -> X=0 -> ALWAYS TRUE

$$C_{2j} - p_{2j} \geq C_{3,j-1} - M(1 - x_{j-1}) \qquad \forall j = 2, ..., n \tag{5}$$

Anestesia of patient 1 starts at t=0

$$C_{11} - p_{11} = 0 \tag{6}$$

Anestesia patient 2 after anestesia patient 1
IF Aw IN OR -> X=1 -> correct
IF Aw IN AOP -> X=0 -> always true -> NO CONSTRAINT

I would put $+P_{31}(1-X_1)$

$$C_{12} - p_{12} \geq C_{11} - M(1 - x_1) \tag{7}$$

Anestesia patient 2 after anestesia patient 1
IF A(2) BEFORE Aw(1) -> $Y_{21}=1$ ->

$$C_{12} - p_{12} \geq C_{11} - M(1 - y_{21}) \tag{8}$$

I don't think there make sense.

$$C_{1j} - p_{1j} \geq C_{3,j-1} - Mx_{j-1} - My_{j,j-1} \qquad \forall j = 2, ..., n \tag{9}$$

$$C_{3,j-1} - p_{3,j-1} \geq C_{1j} - Mx_{j-1} - M(1 - y_{j,j-1}) \qquad \forall j = 2, ..., n \tag{10}$$

$$C_{1j} - p_{1j} \geq C_{1,j-1} - M(1 - x_{j-1}) - M(1 - x_{j-2}) - My_{j-1,j-2} \qquad \forall j = 3, ..., n \tag{11}$$

$$C_{1j} - p_{1j} \geq C_{3,j-2} - M(1 - x_{j-1}) - Mx_{j-2} - M(1 - y_{j-1,j-2}) \qquad \forall j = 3, ..., n \tag{12}$$

$$x_j \in \{0, 1\}\, \forall j = 1, ..., n;\; y_{j,j-1} \in \{0, 1\}\, \forall j = 2, ..., n;\; C_{tj} \in [0, +\infty)\, \forall t = 1, 2, 3,\; j = 1, ..., n \tag{13}$$

The objective function (1) minimize the completion of the observation time of n-th surgery (the last surgery of the sequence σ). Constraints (2) enforce that no-wait is between the anesthesia completion and the patient admission to the OR. Constraints (3) enforce that no-wait is between the surgery completion and the start of the patient awakening/observation. Constraints (4) enforce a surgery to start in the OR after the preceding surgery completion in the case the awakening/observation is performed in the AOP. Constraints (5) enforce a surgery to start in the OR after the awakening/observation completion of the preceding surgery in the case the awakening/observation is performed in the OR. The constraint (6) enforces the first surgery anesthesia to start at time 0. Constraints (9) enforce the anesthesia of j to start in the AOP after the completion of the $(j-1)$-th awakening/observation in the case the $(j-1)$-th awakening/observation is performed in the OR and the anesthesia of j-th patient does not precede the awakening/observation of $(j-1)$-th patient. Constraints (10) enforce the awakening/observation of the $(j-1)$-th patient to start in the AOP after the completion of the j-th patient anesthesia in the case the $(j-1)$-th awakening/observation is performed in the OR and the j-th anesthesia precedes the $(j-1)$-th awakening/observation. Constraints (7) and (8) enforce the anesthesia of the second surgery to starts after the completion of the first surgery anesthesia in the case the second surgery anesthesia precedes the first surgery awakening/observation or in the case the first surgery awakening/observation takes place in the OR respectively.

1.2 A MILP for the IAOPORSP

The MILP formulation of the IAOPORSP is very close to that of the AOPSP, it includes variables x_j, $y_{j,j-1}$, and C_{tj}, plus variables z_{ij} that take value 1 if the surgery i is in the j-th position in the OR sequence of surgeries. The MILP reads:

$$f_{obj} := \min\{C_{3n}\} \tag{14}$$

s.t.

$$C_{2j} - \sum_{i \in I} p_{2i} z_{ij} = C_{1j} \qquad \forall j = 1, ..., n \tag{15}$$

$$C_{3j} - \sum_{i \in I} p_{3i} z_{ij} = C_{2j} \qquad \forall j = 1, ..., n \tag{16}$$

$$C_{2j} - \sum_{i \in I} p_{2i} z_{ij} \geq C_{2,j-1} \qquad \forall j = 2, ..., n \tag{17}$$

$$C_{2j} - \sum_{i \in I} p_{2i} z_{ij} \geq C_{3,j-1} - M(1 - x_{j-1}) \qquad \forall j = 2, ..., n \tag{18}$$

$$C_{11} - \sum_{i \in I} p_{1i} z_{i1} = 0 \tag{19}$$

$$C_{12} - \sum_{i \in I} p_{1i} z_{i2} \geq C_{11} - M(1 - x_1) \tag{20}$$

$$C_{12} - \sum_{i \in I} p_{12} z_{i2} \geq C_{11} - M(1 - y_{21}) \tag{21}$$

$$C_{1j} - \sum_{i \in I} p_{1j} z_{ij} \geq C_{3,j-1} - M x_{j-1} - M y_{j,j-1} \qquad \forall j = 2, ..., n \tag{22}$$

$$C_{3,j-1} - \sum_{i \in I} p_{3,j-1} z_{i,j-1} \geq C_{1j} - M x_{j-1} - M(1 - y_{j,j-1}) \qquad \forall j = 2, ..., n \tag{23}$$

$$C_{1j} - \sum_{i \in I} p_{1j} z_{ij} \geq C_{1,j-1} - M(1 - x_{j-1}) - M(1 - x_{j-2}) - M y_{j-1,j-2} \qquad \forall j = 3, ..., n \tag{24}$$

$$C_{1j} - \sum_{i \in I} p_{1j} z_{ij} \geq C_{3,j-2} - M(1 - x_{j-1}) - M x_{j-2} - M(1 - y_{j-1,j-2}) \qquad \forall j = 3, ..., n \tag{25}$$

$$x_j \in \{0,1\}\, \forall j = 1, ..., n;\; y_{j,j-1} \in \{0,1\}\, \forall j = 2, ..., n;$$
$$z_{ij} \in \{0,1\}\, \forall i \in I, j = 1, ..., n;\; C_{tj} \in [0, +\infty)\, \forall t = 1,2,3,\; j = 1, ..., n \tag{26}$$

Given the MILP for the AOPSP provided in Section 1.1, the MILP for the IAOPORSP model is self-explanatory. For each constraint in the model (1)-(13), there is a corresponding constraint in the model (14)-(26).

DATA INPUT

ORDER	ANESTESIA	SURGERY	AWAKENING
1 2 3 ⋮	5m < A(i) < 15m	20m < S(i) < 120m	10m < AW(i) < 20m

EXACT SOLUTION FOR PROBLEM ①

WORKING, BUT NOT OPTIMAL AS EXPECT.

INITIALIZATION: $B\_A(0) = 0$, $C\_OR = 0$
$C\_AOP = A(0)$

! $C\_OR \equiv$ current available time of OR
$C\_AOP \equiv$ current available time of AOP

TERMINATION: OR(LAST JOB) = 1

! B_A → BEGIN OF ANESTESIA
$P(1-1, J)$ → PYTHON STARTS FROM 0, SO PATIENT 1 IS 1-1 IN PYTHON, WROTE IT LIKE THIS TO NOT MAKE CONFUSION

FOR $i \in [0, \text{LEN(PATIENT\_LIST)} - 1]$

① IF $\left(A(i+1) \leq B(1-1,i) + P(1-1,i) + S(i) - C\_AOP\right)$

$$\Rightarrow \begin{cases} B\_S(i) = B\_A(i) + A(i) \\ B\_AW(i) = B\_S(i) + S(i), \quad AW\_OR = 0 \\ B\_A(i+1) = B\_AW(i) - A(i+1) \\ C\_OR = B\_S(i) + S(i) \\ C\_AOP = B\_AW(i) + AW(i) \end{cases}$$

② IF $\left(A(i+1) > B(1-1,i) + P(1-1,i) + S(i) - C\_AOP\right)$ AND $\left(A(i+1) \leq B(1-1,i) + P(1-1,i) - C\_AOP + S(i) + AW(i)\right)$

$$\Rightarrow \begin{cases} B\_S(i) = B\_A(i) + A(i) \\ B\_AW(i) = B\_S(i) + S(i), \quad AW\_OR = 1 \\ B\_A(i+1) = B\_AW(i) + AW(i) - A(i+1) \\ C\_OR = B\_AW(i) + AW(i) \\ C\_AOP = B\_A(i+1) + A(i+1) \end{cases}$$

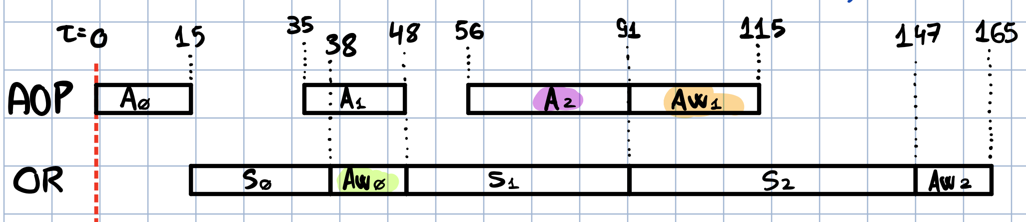
③ IF $A(i+1) > B(1-1,i) + P(1-1,i) + S(i) - C\_AOP$ AND $A(i+1) > B(1-1,i) + P(1-1,i) - C\_AOP + S(i) + AW(i)$

$$\Rightarrow \begin{cases} B\_S(i) = B\_A(i) + A(i) \\ B\_AW(i) = B\_S(i) + S(i), \quad AW\_OR = 1 \\ B\_A(i+1) = B\_A(i) + A(i) \\ C\_OR = B\_AW(i) + AW(i) \\ C\_AOP = B\_A(i+1) + A(i+1) \end{cases}$$

CASE STUDY: 3 JOBS: $P(i, J)$ = PROCESSING TIME OF OPERATION $i \in (0,1,2) \equiv (A, S, AW)$ ON PATIENT $J \in (0,1,2)$

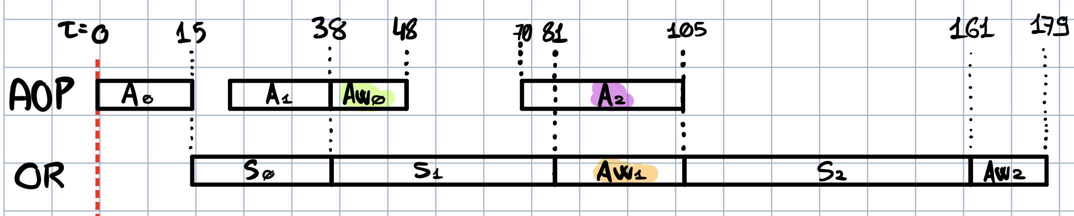
$P(:,0) = [15, 23, 10]$, $P(:,1) = [13, 43, 24]$, $P(:,2) = [35, 56, 18]$,

SCHEDULE WITH GUROBI (ORDER OF PATIENTS FIXED)



O.F. = 165

SAME ORDER OF PATIENTS WITH GREEDY ALGORITHM ABOVE

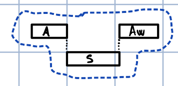


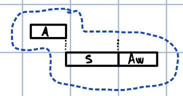
O.F. = 179

COMMENTS: GREEDY ALGORITHM PLACES AW OF JOB 0 (JOB ≡ PATIENT) IN AOP, AND INTUITIVELY THIS WOULD MAKE EVERYTHING SHIFT BACK (THUS IMPROVING THE OBJ. FUN) HOWEVER THIS MAKES SO THAT A OF JOB 2 DOESN'T FIT BETWEEN END OF $AW(0)$ AND END OF $S(1)$, THUS FORCING $AW(1)$ IN OR AND PUSHING EVERYTHING TO BE LATE (MORE LATE THAN $AW(0)$ IN AOP HAS PUSHED EVERYTHING BACK).
⇒ DECISION OF AW IN OR/AOP DEPENDS ALSO ON FUTURE JOBS, NOT ONLY ON NEARBY ONES.

EXACT SOLUTION OF ①, BACKWARD APPROACH

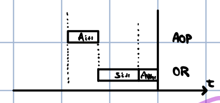
Same variables of above: C, OR, P (Nx3)

OR = ∅: 

OR = 1: 

SUPP ⟶ SUPPORT VARIABLE TO FIND AVAILABLE TIME BETWEEN C_A(N+1) AND B_A(N+2))

INITIALIZATION (after reordering...): LAST JOB



C, NOT P!

① $AW(i) \leq SUPP - C\_A(i+1)$ ⟶ Can i fit Aw(i) in the timeslot available after A(i+1)?

② $s(i) \geq A(i+1)$ ⟶ Can s(i) fit under A(i+1) by itself?

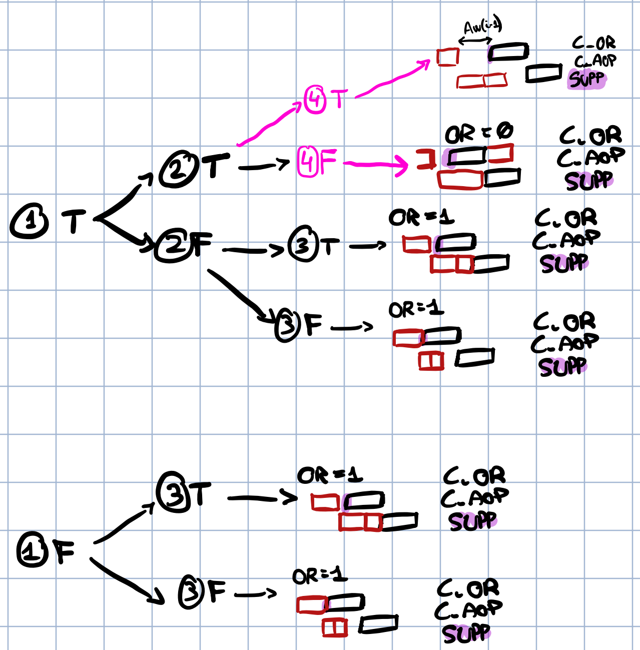
③ $AW(i) + S(i) \geq A(i+1)$ ⟶ Can s(i) and Aw(i) fit under A(i+1) together?

④ $Aw(i-1) + A(i+1) \geq S(i) + Aw(i)$ AND $Aw(i-1) \geq Aw(i)$ AND $i > 1$

N) $C\_Aw(N) = \emptyset$, $OR(N) = 1$, $C\_S(N) = -P\_AW(N)$, $C\_A(N) = -(P\_AW(N) + P\_S(N))$

$C\_OR = C\_A(N)$, $C\_AOP = C\_A(N) - P\_A(N)$, $SUPP = 1000$.

FOR i IN RANGE(N-2, -1, -1):



GUROBI SOLVING OF ①

- Problem 1 with GUROBI and provided algorithm.
- Operation 5 Anestesia overlaps with patient 1 Awakening in AOP
  - ⇒ Check for errors in copying model **NO ERRORS**
  - ⇒ Check for model correctness. ✱

- My attempt: $i \in [1,2,3] \equiv [A, S, Aw]$, $J \in [1, N]$, $OR \in \{0,1\}$ | $P(i,J)$ = PROC. TIMES
  $M = 5000$

$C(i,J) \longrightarrow$ Completion time of operation i of job J

$OR(J) \longrightarrow$ 1 if patient J awakens in the OR.

$$\underset{min.}{O.F.} = C(3,N)$$

MODEL:

⓪	$C(1,1) - P(1,1) = \emptyset$		Initialization of first job
①	$C(2,J) - P(2,J) = C(1,J)$	$\forall J$	No Time between $A(J), S(J), Aw(J)$
②	$C(3,J) - P(3,J) = C(2,J)$	$\forall J$	
③	$C(1,J+1) - P(1,J+1) \geqslant C(1,J)$	$\forall J \in [1 \dots N-1]$	Ordering of patients.
⑤	$C(3,J+1) - P(3,J+1) \geqslant C(3,J) - P(3,J)$	$\forall J \in [1 \dots N-1]$	
⑥	$C(2,J+1) - P(2,J+1) \geqslant C(2,J) + P(3,J) \cdot OR(J)$	$\forall J \in [1 \dots N-1]$	Non overlapping in the OR.
⑦	$C(1,J+2) - P(1,J+2) \geqslant C(3,J) - M \cdot OR(J)$	$\forall J \in [1, N-2]$	Non overlapping of $Aw(J)$ and $A(J+2)$ if $Aw(J)$ in AOP
⑧	$C(3,J) - P(3,J) \geqslant C(1,J+1) - M \cdot OR(J)$	$\forall J \in [1, N-1]$	Non overlapping of $Aw(J)$ and $A(J+1)$

! Last two job's Aw may overlap using this code, problem is solved by imposing $OR[N] = 1$ (O.F. doesn't change)

GUROBI SOLVING OF 2

- Attempt 1:

~~$C(i,J) \longrightarrow$ Completion time of operation $i \in [1,2,3] \xrightarrow{PY} [0,1,2]$ of Job $J \in [1,2 \dots N] \xrightarrow{PY} [0,1,\dots N\text{-}1]$~~

~~$OR(J) \longrightarrow$ Binary variable, 1 if patient J awakens in the OR.~~

~~$PREC(J1,J2) \longrightarrow$ Binary matrix, each entry = 1 if Job J1 precedes J2 in the schedule~~

~~DEFINITION OF PRECEDENCE OF J1 ON J2: A(J1) BEGINS BEFORE A(J2)~~

O.F. $\min(\max(C(3,J)))$

CONSTRAINTS:

	① $C(1,J) - P(1,J) \geq 0$	$\forall J$	] Initialization of first Job (whichever it is)
	② $C(2,J) - P(2,J) - C(1,J) = \emptyset$	$\forall J$	] No time between operations
	③ $C(3,J) - P(3,J) - C(2,J) = \emptyset$	$\forall J$	of each Job.
	④ $C(1,J1) - P(1,J1) - C(1,J2) + M \cdot PREC(J1,J2) \geq \emptyset$	$\forall J1,J2 \mid J_1 \neq J_2$	] Ordering of
	⑤ $C(2,J1) - P(2,J1) - C(2,J2) + P(2,J2) + M \cdot PREC(J1,J2) \geq 0$	$\forall J1,J2 \mid J_1 \neq J_2$	patients.
WRONG	⑥ $C(3,J1) - P(3,J1) - C(3,J2) + M \cdot PREC(J1,J2) \geq \emptyset$	$\forall J1,J2 \mid J_1 \neq J_2$	
	⑦ $C(2,J1) - P(2,J1) - C(2,J2) - P(3,J2) \cdot OR(J2) + M \cdot PREC(J1,J2) \geq \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] Non overlapping in the OR.
	⑧		
WRONG	⑨ $C(3,J1) - P(3,J1) - C(1,J2) + M \cdot OR(J1) + M \cdot PREC(J1,J2) \geq 0$	$\forall J1,J2 \mid J1 \neq J2$	] Non overlapping of Aw(J) and A(J+1)

- Attempt 2:

$C(i,J) \longrightarrow$ Completion of operation $i \in [1,2,3]$ of Job $J \in [1 \dots N]$

$OR(J) \longrightarrow$ Binary variable = 1 if Job J awakens in the OR.

$PREC(i1,J1,i2,J2) \longrightarrow$ Binary matrix (4 dimensions, $9 \times N^2$ elements), 1 if operation $(i1,J1)$ precedes $(i2,J2)$

O.F. $\min(\max(C(3,J))) \rightsquigarrow \min(F)$ ($F \longrightarrow$ INT, support variable to express $\max(3,J)$)

CONSTRAINTS:

① $C(1,J) - P(1,J) \geq 0$	$\forall J$	] Initialization of first Job (whichever it is)	
② $C(2,J) - P(2,J) - C(1,J) = \emptyset$	$\forall J$	] No time between operations	
③ $C(3,J) - P(3,J) - C(2,J) = \emptyset$	$\forall J$	of each Job.	
④ $C(2,J1) - P(2,J1) - C(2,J2) + M \cdot PREC(2,J1,2,J2) \geq \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] SURG AND SURG	No OVERLAPPING
⑤ $C(2,J1) - P(2,J1) - C(3,J2) + M \cdot PREC(2,J1,3,J2) + M \cdot (1 - OR(J2)) \geq \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] SURG AND Aw	
⑥ $C(3,J1) - P(3,J1) - C(2,J2) + M \cdot PREC(3,J1,2,J2) + M \cdot (1 - OR(J1)) \geq \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] Aw AND SURG	
⑦ $C(1,J1) - P(1,J1) - C(1,J2) + M \cdot PREC(1,J1,1,J2) \geq \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] A AND A	
⑧ $C(1,J1) - P(1,J1) - C(3,J2) + M \cdot PREC(1,J1,3,J2) + M \cdot OR(J2) \geq \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] A AND Aw	
⑨ $C(3,J1) - P(3,J1) - C(1,J2) + M \cdot PREC(3,J1,1,J2) + M \cdot OR(J1) \geq \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] Aw AND A	
⑨.1 $C(3,J1) - P(3,J1) - C(3,J2) + M \cdot PREC(3,J1,3,J2) + M \cdot (OR(J1) + OR(J2)) \geq \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] Aw AND Aw	Only in Aor they can overlap.
⑩ $F - C(3,J) \geq 0$	$\forall J$	] $F = \max(C(3,J))$	
⑪ $Prec(1,J1,1,J2) + Prec(1,J2,1,J1) - 1 = \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] A AND A	
⑫ $Prec(2,J1,2,J2) + Prec(2,J2,2,J1) - 1 = \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] S AND S	
⑬ $Prec(3,J1,3,J2) + Prec(3,J2,3,J1) - 1 = \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] Aw AND Aw	
⑭ $Prec(2,J1,3,J2) + Prec(3,J2,2,J1) - 1 = \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] S AND Aw	
⑮ $Prec(1,J1,3,J2) + Prec(3,J2,1,J1) - 1 = \emptyset$	$\forall J1,J2 \mid J1 \neq J2$	] A AND Aw	

GUROBI SOLVING OF ② - ALTERNATIVE

$C(i,J) \longrightarrow$ Completion time of operation i of JTH job in the sequence

$OR(J) \longrightarrow$ 1 if JTH patient of schedule awakens in the OR.

$Z(J1,J2) \longrightarrow$ Binary, 1 if job J1 is in J2TH position of the schedule

$$\text{O.F.} \overset{\min}{=} C(3,N)$$

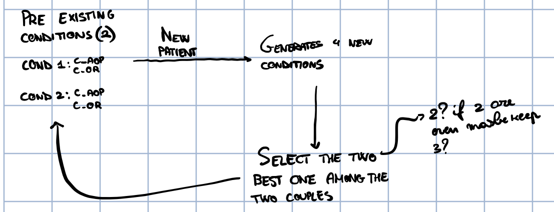
$$!\ S(2,K,J) = \sum_{K \in J1} P(2-1,K) \cdot Z(K,J)\ !$$

MODEL:

000	$C(1-1,1-1) - S(1,K,1) = \varnothing$		Initialization and finalization
001	$OR(N\_PAT) = 1$		
10	$\sum_{J1} Z(J,J1) = 1$	$\forall J$	Only one job per position.
11	$\sum_{J1} Z(J1,J) = 1$	$\forall J$	
1	$C(2,J) - S(2,K,J) = C(1,J)$	$\forall J$	No Time between A(J), S(J), Aw(J)
2	$C(3,J) - S(3,K,J) = C(2,J)$	$\forall J$	
3	$C(1,J+1) - S(1,K,J+1) \geq C(1,J)$	$\forall J \in [1 \dots N-1]$	Ordering of patients.
5	$C(3,J+1) - S(3,K,J+1) \geq C(3,J) - S(3,K,J)$	$\forall J \in [1 \dots N-1]$	
6	$C(2,J+1) - S(2,K,J+1) \geq C(2,J)$	$\forall J \in [1 \dots N-1]$	Non overlapping in the OR.
6bis	$C(2,J+1) - S(2,K,J+1) \geq C(2,J) + S(3,K,J) - M \cdot (1 - OR(J))$	$\forall J \in [1 \dots N-1]$	Non overlapping in the OR when OR=1
8	$C(3,J) - S(3,K,J) \geq C(1,J+1) - M \cdot OR(J)$	$\forall J \in [1, N-1]$	Non overlapping of Aw(J) and A(J+1)
7	$C(1,J+2) - S(1,K,J+2) \geq C(3,J) - M \cdot OR(J)$	$\forall J \in [1, N-2]$	Non overlapping of Aw(J) and A(J+2) if Aw(J) in AOP

CONSTRAINT INUTILE !

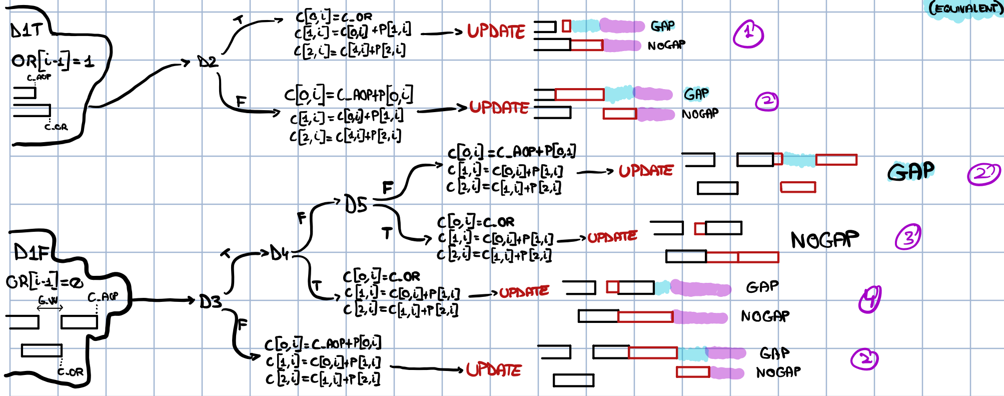
B&B WIDTH2



2ND ATTEMPT: how to fit patient i after patients i-1.
I only have P∈[3 n], OR∈[n]

D1: OR(i-1)=1
D2: A(i)≤C_OR-C_AOP
D3: A(i)≤GAP_W
D4: S(i)≥AW(i-1)
D5: OR(i)=1

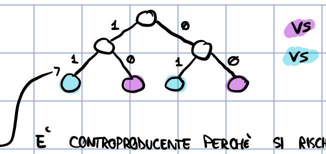
CHANGED TO C_AOP-C_OR (EQUIVALENT)



RESULTS:

The algorithm turns out to be just a more complicated version of the greedy forward algorithm.
Good results but not optimal.

B&B 1 (NON FUNZIONANTE)



CONDIZIONE DI PRUNING: $OF^{I} < OF^{II}$ AND $GAP^{I} > GAP^{II}$

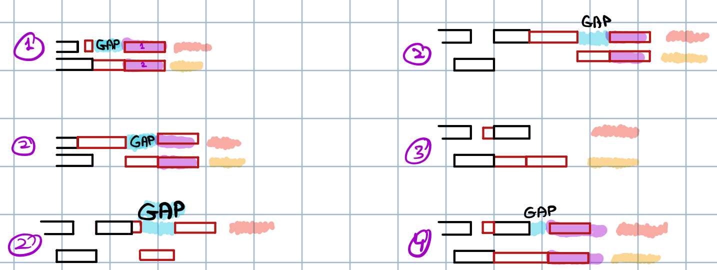
PROBLEMI: • CONFRONTARE "SOLO" JOB PROVENIENTI DALLO STESSO NODO È CONTROPRODUCENTE PERCHÉ SI RISCHIA DI LASCIARE BRANCH INUTILI. (PERCHÉ NON DOVREI CONFRONTARE CON BRANCH LONTANI?)

• LA CONDIZIONE $GAP^{I} > GAP^{II}$ NON SI PUÒ APPLICARE A SCHEDULE CHE FINISCONO CON RISVEGLIO IN O.R. (GAP = ∅)

B&B 2 (SOLUZIONE FINALE)

A DESTRA METTO TUTTI I POSSIBILI CASI DI POSIZIONAMENTO.

(I NUMERI VIOLA SERVONO A RAGGRUPPARE LA DECISIONE PRESA)
(EVIDENZIATORE VIOLA PER AW. NON POSIZIONATO)



LOGICA (ITERATO PER LIVELLI DELL'ALBERO)

1) PRENDO TUTTO IL SUBSET DI SOLUZIONI ANCORA VIVE DA LIVELLO PRECEDENTE E GENERO IL NUOVO LIVELLO (OGNI NODO SI DIVIDE IN DUE, OR=1 e OR=∅.

2) DIVIDO IN DUE GRUPPI, TUTTE LE SOL. CON OR=1 e TUTTE QUELLE CON OR=∅. APPLICO UN CRITERIO DIVERSO PER I DUE GRUPPI, CONFRONTANDO TUTTI GLI ELEMENTI DI UN GRUPPO.

3) PIÙ CHE CERCARE LA SOLUZIONE OTTIMALE, MI SONO CONCENTRATO SULLO SCARTARE LE NON OTTIMALI PER CERTO.

!C_AOP = PRIMO MOMENTO DISPONIBILE IN AOP!

• PER OR=1: $OF^{I} \leq OF^{II}$ AND $C\_AOP^{I} \leq C\_AOP^{II}$

COME SI PUÒ VEDERE NEI CASI ARANCIONI, SE LA CONDIZIONE È VERIFICATA DI SICURO LA SOL^{II} È UGUALE O PEGGIORE ALLA I (OSSIA, LA SOL^{I} PUÒ ESEGUIRE GLI STESSI POSIZIONAMENTI DELLA SOL^{II} PIÙ EVENTUALMENTE ALTRI), QUINDI LA SOL^{II} PUÒ ESSERE SCARTATA.
OVVIAMENTE SE LA CONDIZIONE INVERSA È VERIFICATA SCARTO LA SOL^{I} ($OF^{I} \geq OF^{II}$ AND $C\_AOP^{I} \geq C\_AOP^{II}$), SE NESSUNA DELLE DUE TENGO ENTRAMBE.

! ≤ INVECE DI < PERCHÈ SE SONO UGUALI TANTO VALE SCARTARNE UNA!

• PER OR=∅: $OF^{I} \leq OF^{II}$ AND $GAP^{I} \geq GAP^{II}$

STESSO PRINCIPIO DI PRIMA, SE LA CONDIZIONE È VERIFICATA ALLORA SOL^{II} SICURAMENTE PUÒ ESSERE TAGLIATA.

4) PROSSIMO LIVELLO

5) QUANDO FINISCO I LIVELLI HO SOLO 2÷6 FOGLIE TRA CUI SCEGLIERE.

CONSIDERAZIONI:

• NON CAMBIA MOLTO TRA B&B 1 e B&B 2, CI SONO ALCUNE COSE CHE NON AVEVAMO VALUTATO

• GAP MAGGIORE È UNA BUONA CONDIZIONE PERCHÉ LA DURATA DELL'AW È COSTANTE IN UNO STESSO LIVELLO, QUINDI C_AOP NON È RILEVANTE (O MEGLIO È AUTOMATICAMENTE GESTITO)

• CONFRONTARE TRA TUTTO IL LIVELLO HA SENSO VISTO CHE LE SOLUZIONI RIMANGONO POCHE, ANCHE QUANDO NIENTE VIENE TRONCATO (NON SO SE PUÒ SUCCEDERE) SI ARRIVA A 8 SOL. ATTIVE)

• CON QUESTI CRITERI DI SCELTA SI TROVA UNA SOLA SOLUZIONE (MAX 2 PER POSIZIONAMENTO ININFLUENTE ULTIMO PAZIENTE), TOGLIENDO GLI "UGUALE" DALLE CONDIZIONI SI TROVANO SOLUZIONI EQUIVALENTI ALLA OTTIMALE CON POSIZIONAMENTI DIVERSI)

ALLEGO FOTO DIMOSTRATIVE ESEGUITE CON 42 PAZIENTI.

CASE 2 PREPARATION:

I need to extract information about an existing schedule:

INPUTS

C
P
OR

→ EXTRACTOR →

METRIX

O.F.
AMOUNT OF GAPS
EQUILIBRIUM OF THE TWO ROOMS

Research:

- Order of patients can be seen as seating arrangement problem, with preferences, being dependent on OR placement, that are gaps and correct fitting.
- It's also possible to see it as a variation of the TSP, but cost function depends on previous and future choices. Like having a map of nodes with a different height.
- Simulated annealing often pops up.
- Graph partition problem: may be useful for making problem easier.
- Look more into Job-flow shop theory. Try to apply variations of Johnson's algorithm maybe?
- Graph theory can be useful.
- Sequence dependent setup, very relatable.

Non si può trovare una cost function tra un paziente e il successivo, dipende da più delle coppie, non si può impostare come un TSP.

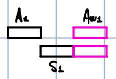
① GENETIC ALGORITHM

② SIMULATED ANNEALING.

③ METTO IL PRIMO, CHI È PIÙ VICINO A RIEMPIRE UNO DEI DUE SPAZI FORMATI?

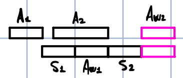
CASE 2, ATTEMPT GREEDY

1) Find job with shortest A and put it first, remove it from available jobs.

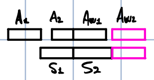


2) SECOND JOB: ⓐ Either the job with $A_2 \to S_1 + A_{w1} \implies OR(1) = 1$

ⓑ Either the job with $A_2 \overset{*}{\to} S_1^-$ $(A_2 < S_1) \implies OR(1) = \emptyset$

ⓐ 

In this situation i can take again choices ⓐ and ⓑ

ⓑ 

In this situation again ⓐ and ⓑ but

3) ⓒ: $A_3 \to (A_{w2} + S_2 - A_{w1}) \implies OR(2) = 1$

ⓓ: $A_3 \to (S_2 - A_{w1})^- \implies OR(2) = \emptyset$

$\left\{\begin{array}{l} * \text{ Here i would like to add a penalty for other factors, for now i'll leave it as } \emptyset. \\ S_2 \to A_{w1} + A_3 \text{ OR } S_2 + A_{w2} \to A_{w1} + A_3 \end{array}\right.$

3) ⓒ → Back to case ⓐ

ⓓ → Back to case ⓑ

Done, as expected not optimal, the more patients there are the bitter.

I can start a local search from here, it finds good sequences.

COSE DA DISCUTERE