SMU Singapore Management University | School of Computing and Information Systems

CS606 AI Planning and Decision Making

University Timetabling

Vu Viet Linh
Rao Ningzhen
Nguyen Thi Ngoc Anh
Huang Anni
Yang Guowei
Tang Jingxue

Table of contents

1. INTRODUCTION

1.Introduction

- ITC 2019: International Timetabling Competition

courses

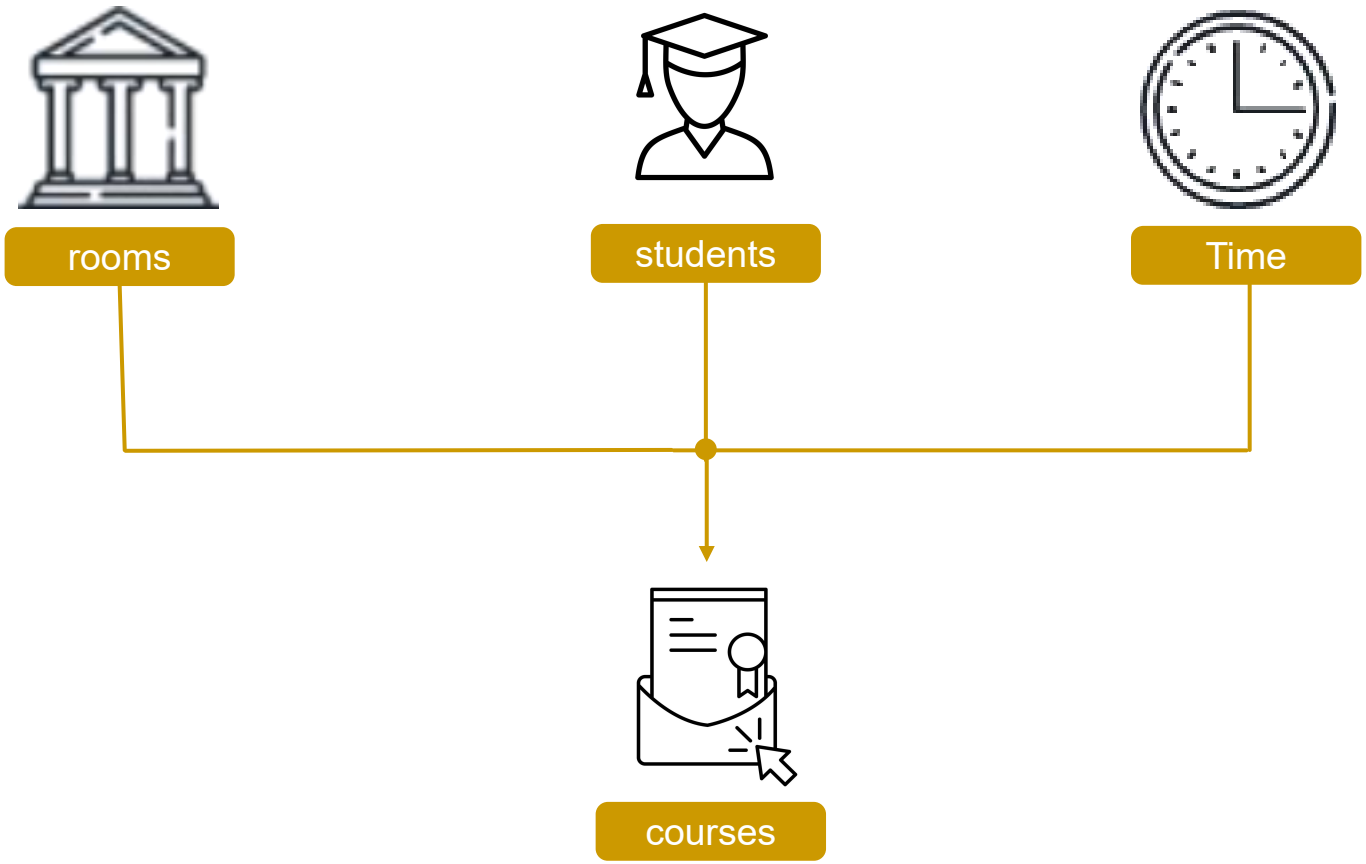
students

Goal: To find a proper time, a room and students for all the classes

https://www.itc2019.org/home

1.Introduction

Goal: find a proper time, a room and students for all the classes



https://www.itc2019.org/home

1.Introduction

- Model used by competition participants
  - Mixed Integer Programming

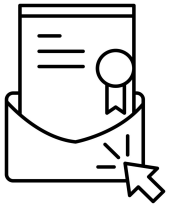
- Model used by us
  - Mixed Integer Programming
  - *Constraint Programming*

2. PROBLEM DEFINITION

2.Problem Definition

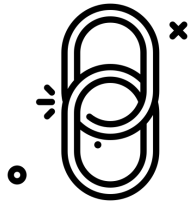
2.1 Data

rooms



courses

students



constraints

2.Problem Definition

2.1 Data

- Rooms

capacity

The max **students'** number in one room

room r

ROOM R	MON	TUE	WED	THU	FRI	SAT	SUN
08:00							
10:00							
12:00							
14:00							
16:00							
18:00							

Not available on Sat noon

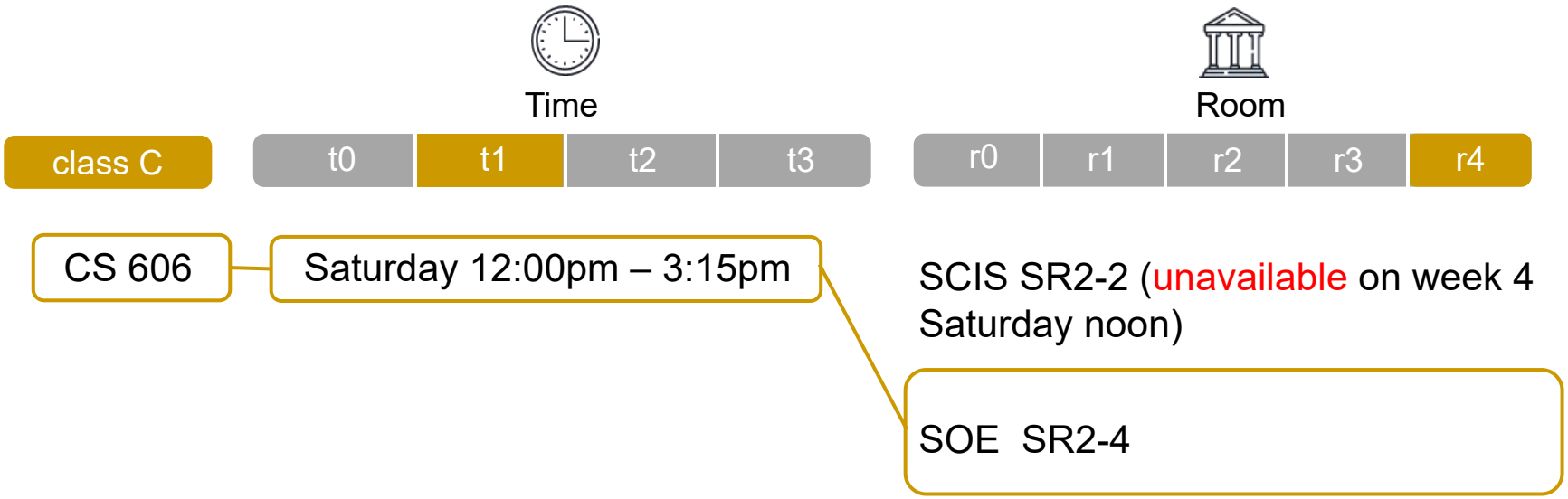
unavailable

Some rooms are **not available at some time slots**

2.Problem Definition

2.1 Data

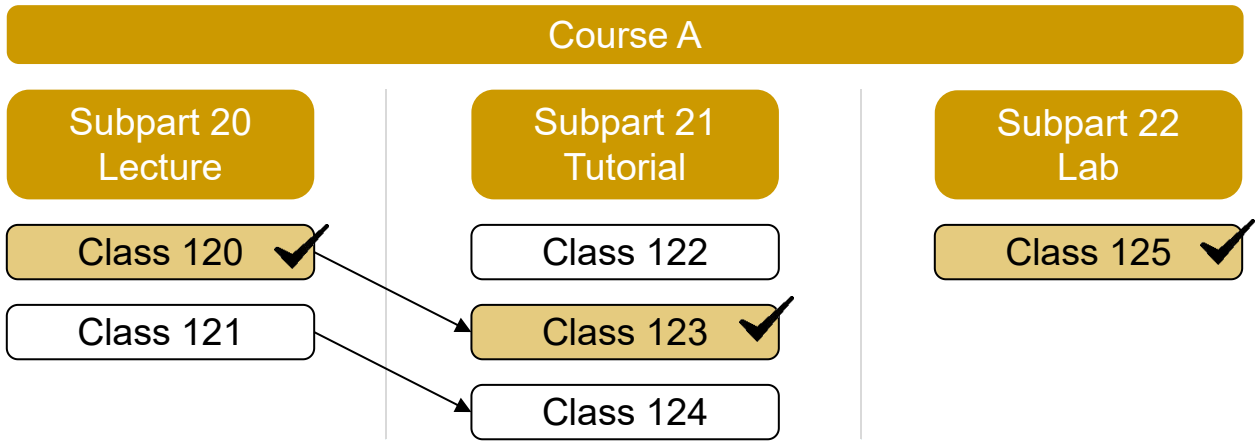
- Classes



2.Problem Definition

2.1 Data

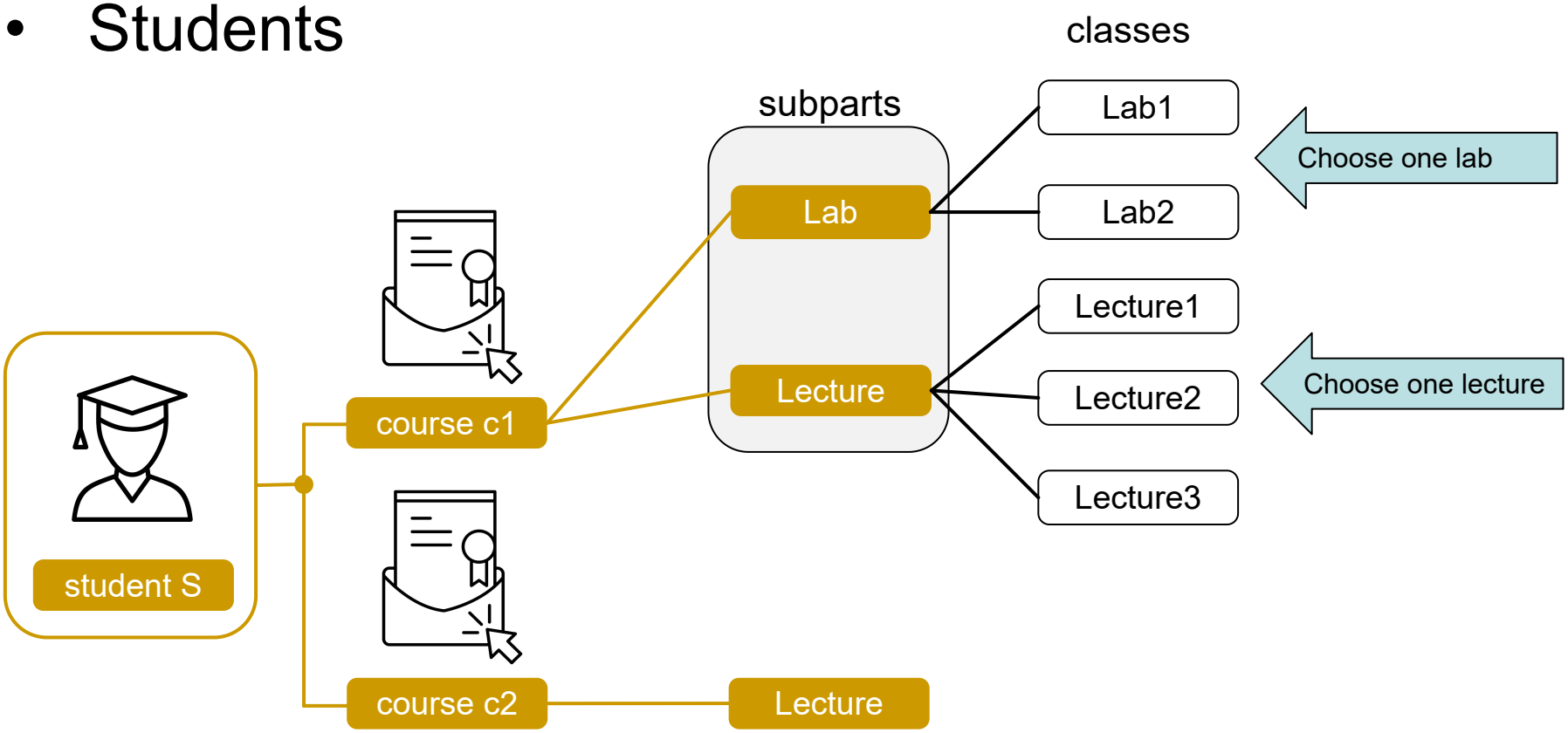
- Courses
  - Each course has multiple **subparts**
  - Each subpart has multiple **classes**
  - A student need to choose **one class in each subpart** for each course
  - Some class has **prerequisites**



2.Problem Definition

2.1 Data

- Students



2.Problem Definition

2.1 Data

- Distributed Constraints

  We only consider hard constraints: **SameAttendees** in this project

Constraint	Complementary	Time	Days	Weeks	Room	Pairs
SameStart		✓	✗	✗	✗	✓
SameTime	DifferentTime	✓	✗	✗	✗	✓
SameDays	DifferentDays	✗	✓	✗	✗	✓
SameWeeks	DifferentWeeks	✗	✗	✓	✗	✓
SameRoom	DifferentRoom	✗	✗	✗	✓	✓
Overlap	NotOverlap	✓	✓	✓	✗	✓
SameAttendees		✓	✓	✓	✓	✓
Precedence		✓	✓	✓	✗	✓
WorkDay(S)		✓	✓	✓	✗	✓
MinGap(G)		✓	✓	✓	✗	✓
MaxDays(D)		✗	✓	✗	✗	days over D
MaxDayLoad(S)		✓	✓	✓	✗	slots over S
MaxBreaks(R,S)		✓	✓	✓	✗	breaks over R
MaxBlock(M,S)		✓	✓	✓	✗	blocks over M

2.Problem Definition

2.2 Notations

Table 1: Notation

Symbol	Definition
Sets	
$c \in C$	set of classes
$t \in T$	set of time periods
$s \in S$	set of students
$r \in R$	set of rooms
$r \in CR_c$	set of rooms each class can use
$t \in CT_c$	set of time slots each class can use
$c \in SC_s$	set of Courses that each student need to learn
$(c_i, c_j) \in SA$	set of class pairs c_i, c_j that can't overlap
Parameters	
$pr_{c,r}$	penalty of choosing room r of class c
$pt_{c,t}$	penalty of choosing time t of class c
M_r	capacity of each room r
Decision Variables	
$x_{c,t}$	binary variable that indicates whether class c is scheduled to time slot t
$y_{c,r}$	binary variable that indicates whether class c takes place at room r
$z_{s,c}$	binary variable that indicates whether student s choose class c

2.Problem Definition

2.3 Assumptions

- There's **no travel time** between two classes.
- Do not consider soft constraints and only consider **SameAttendees** in **distributed constraints**.

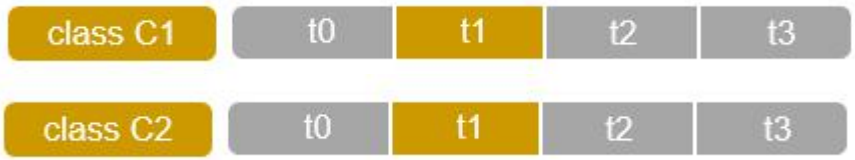
2. Problem Definition (MP)

- Decision variables

$$x_{c,t} = \begin{cases} 1 & \text{The class c takes place at time t} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{c,r} = \begin{cases} 1 & \text{The class c takes place in room r} \\ 0 & \text{otherwise} \end{cases}$$

$$z_{s,c} = \begin{cases} 1 & \text{The student s is assigned to class c} \\ 0 & \text{otherwise} \end{cases}$$



No global identifier for timeslot

2. Problem Definition (MP)

- Objective

$$\text{Min} \quad \sum_{c \in C} \sum_{r \in R} y_{c,r} pr_{c,r} \qquad \text{Room penalty}$$

$$+ \sum_{c \in C} \sum_{t \in T} x_{c,t} pt_{c,t} \qquad \text{Time penalty}$$

2.Problem Definition (MP)

- Constraints
  - H1: Every class must be assigned a time
  - H2: Every class must be assigned a room
  - H3: Every student must attend exactly one class for each subpart that he/she must attend

2.Problem Definition (MP)

- Constraints
  - H4: Prerequisite:
    - Ex: If you take AI Planning – you must take Algorithm as well
  - H5: The capacity limit of each class must not be exceeded.
  - H6: A room cannot be used when it is unavailable
  - H7: The pairs of classes in SameAttendees should not overlap in time
    - Ex: Prof Dai is teaching Algorithm & Applied ML – these 2 classes cannot happen at the same time
  - H8: Two classes can’t be at the same time and the same room

2. Problem Definition (CP)

- Decision variables

$$x_c = t, for\ c \in C \qquad \text{Domain: } t\ avail\ for\ c$$

$$y_c = r, for\ c \in C \qquad \text{Domain: } r\ avail\ for\ c$$

$$z_{s,sbp} = c, for\ s \in S, for\ sbp \in SP_s$$

Domain: c in SPs

Time for class c is t
Room for class c is r
The class choice of subpart sbp for student s is c

2. Problem Definition (CP)

- Objective

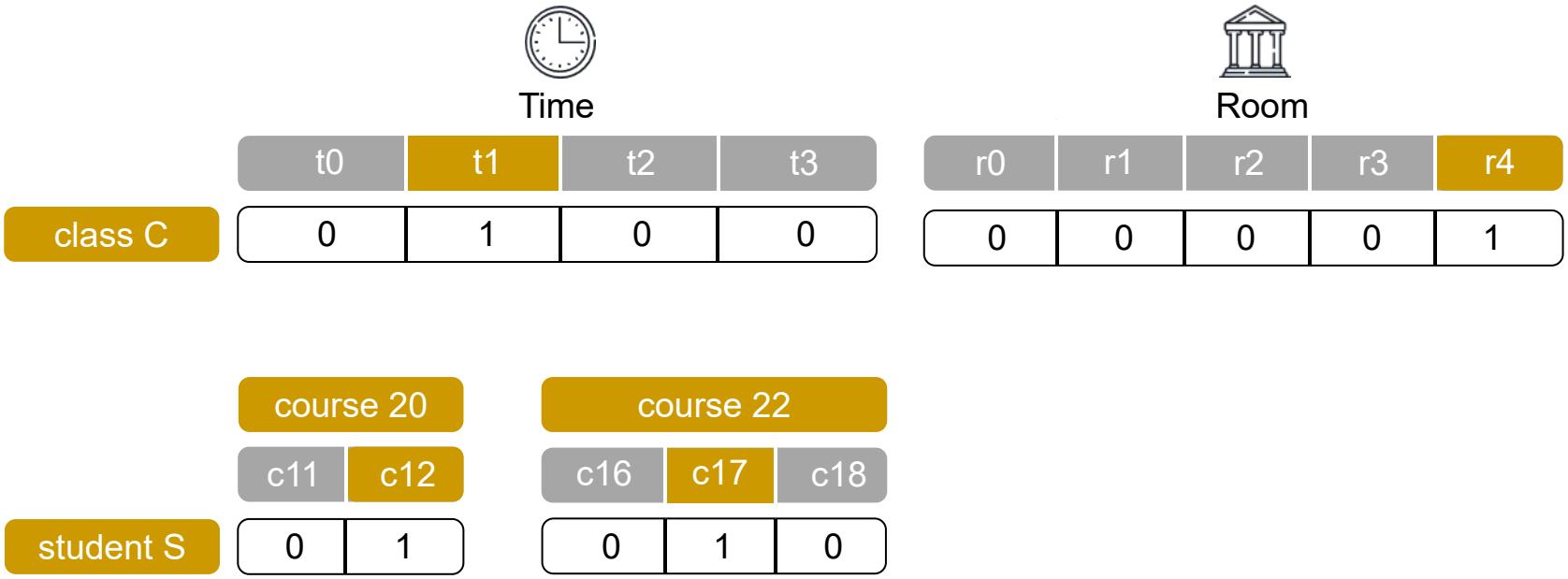
$$min \sum_c (p_{t_c} + p_{r_c})$$

- Constraints
  - H1,H2,H3 are satisfied within cp
  - H4: $if\ z_{s,sub} = c, then\ z_{s,sub`} = c.parent$
  - H5: $count(z, c) \leq limit(c)$
  - H6: $if\ x_c = t,\ then\ y_c \neq \mathrm{r}$ for t in r.avail()
  - H7: $if\ x_{c1} = t_1, then\ x_{c2} \neq t_2$ if ovl(t1,t2)
  - H8: $if\ x_{c1} = t_1, and\ x_{c2} = t_2, and\ y_{c1} = r1, then\ y_{c2} \neq r_2$ if ovl(t1,t2) and r1=r2

3. SOLUTION APPROACH

3. Solution Approach

- Mixed Integer Programming
  - Binary variables
  - Linear constraints

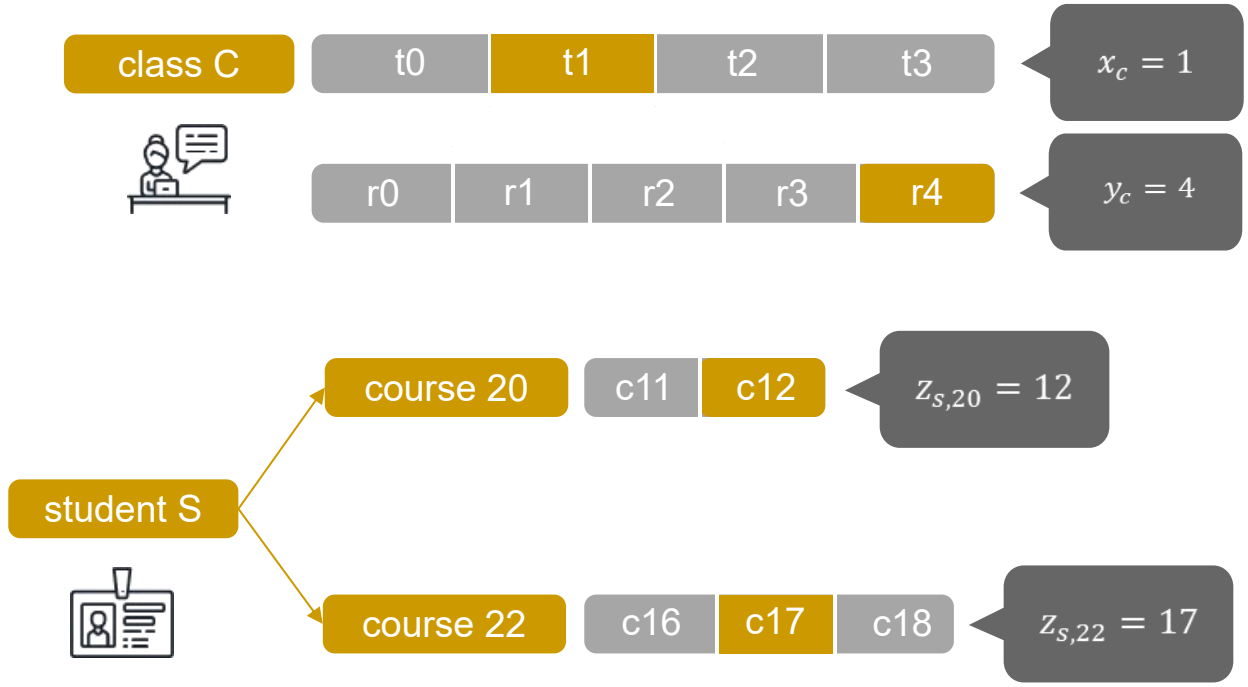


3.Solution Approach

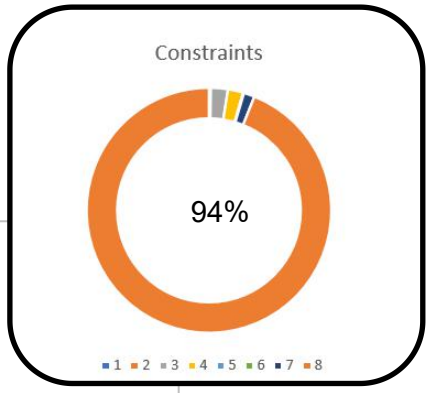
- Constraint Programming
  - Integer variables:

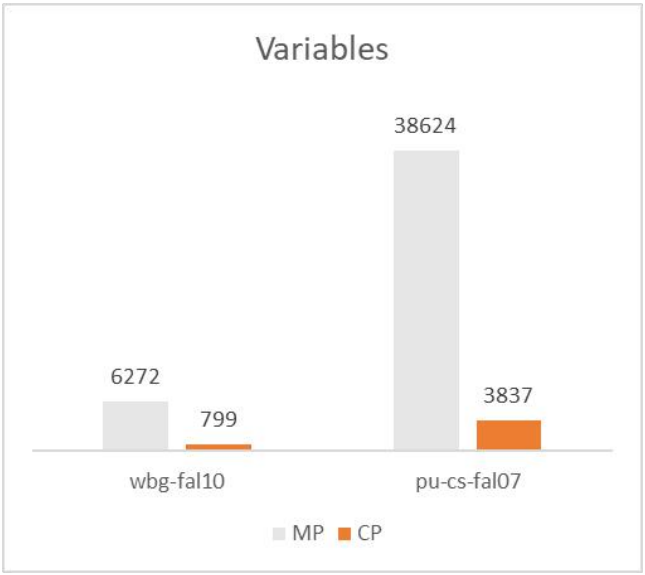
$$x_c = t$$
$$y_c = r$$
$$z_{s,sbp} = c$$

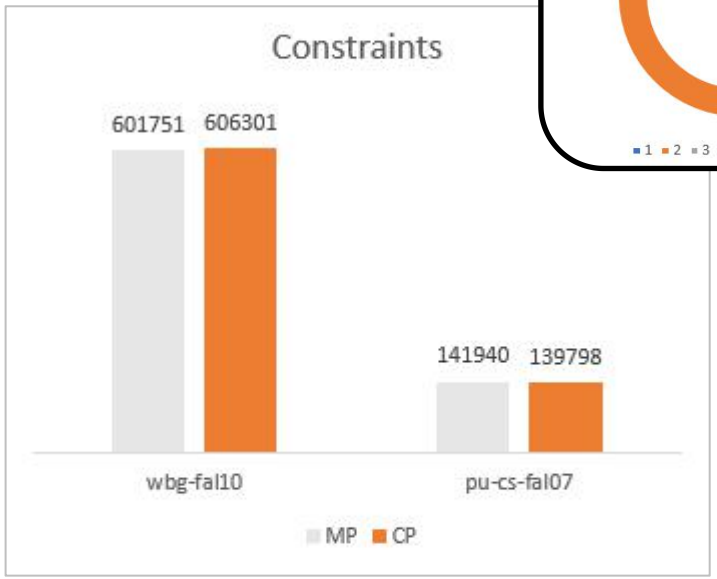
  - Logical constraints:
    - if_else
    - logical_and



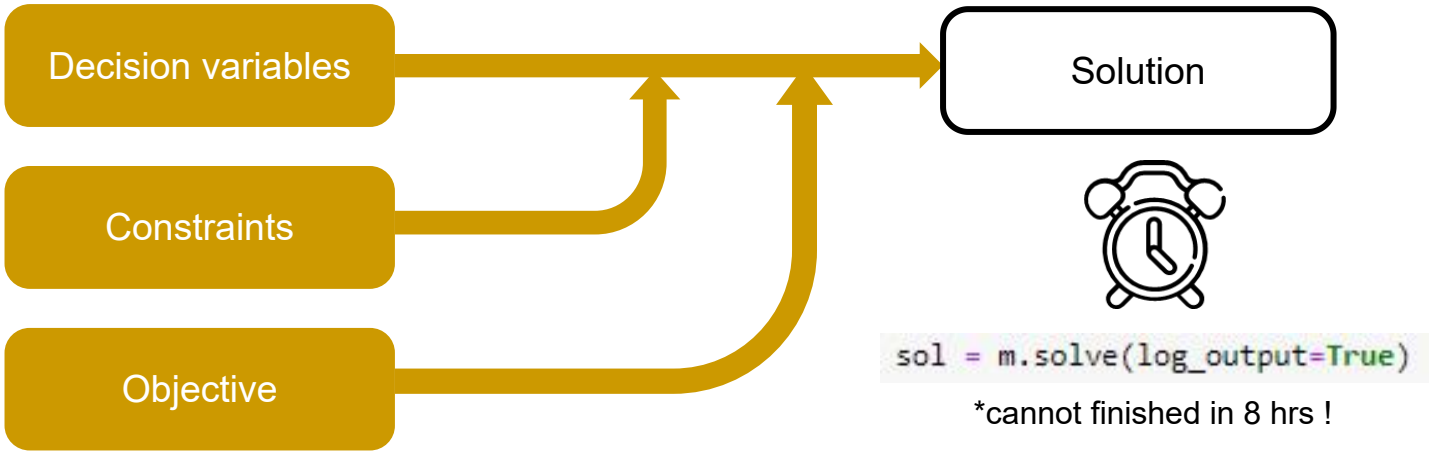
3.Solution Approach





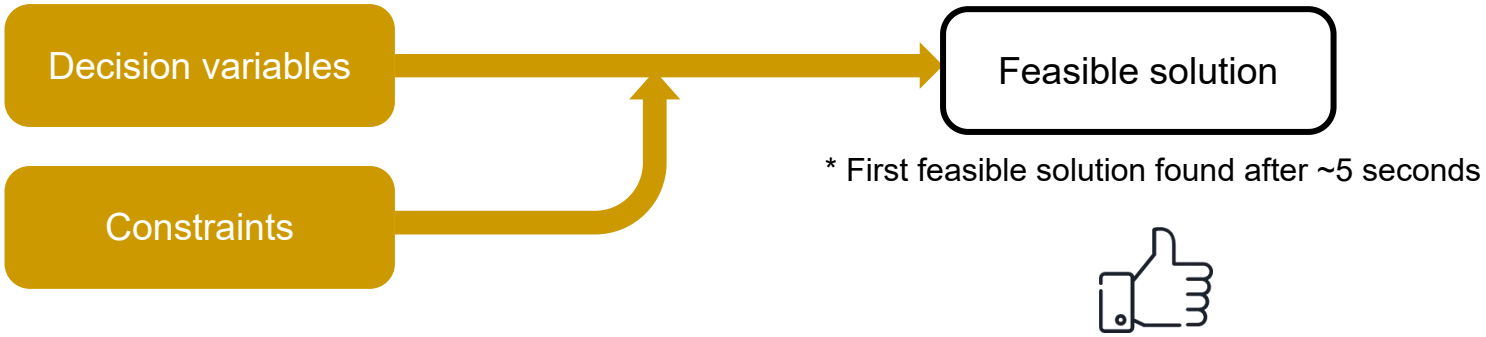


3. Solution Approach



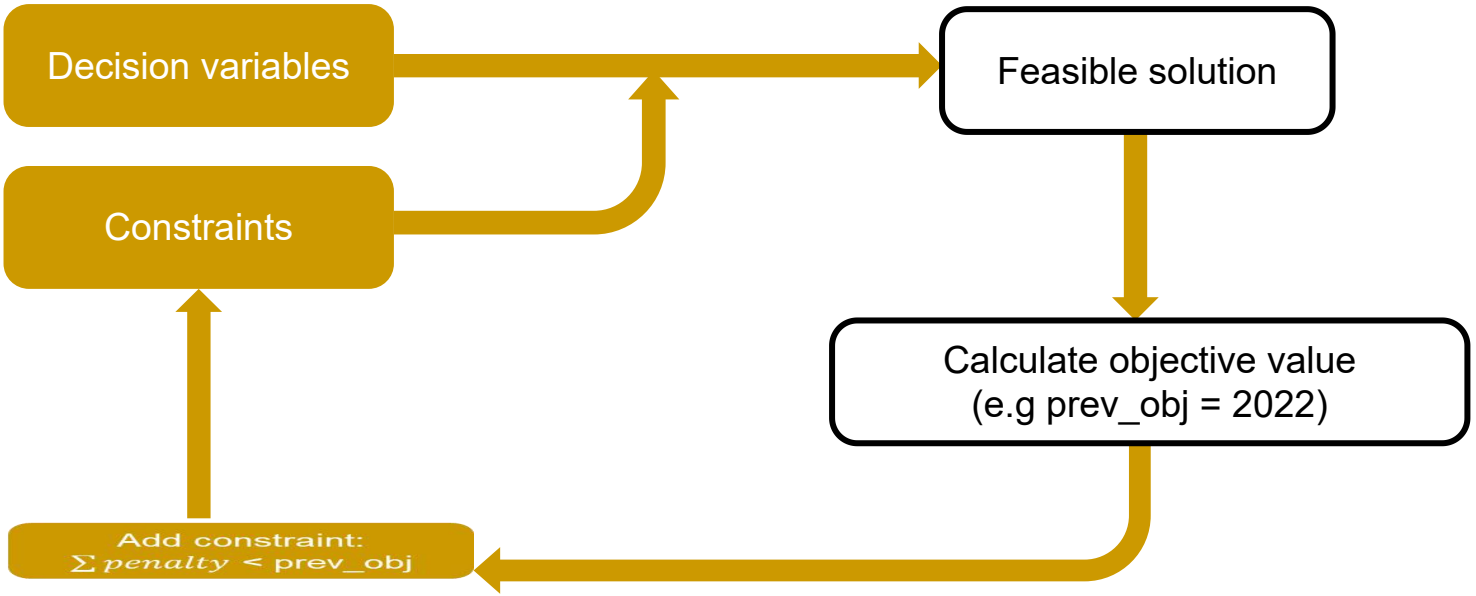
3. Solution Approach

- Mixed Integer Programming



3. Solution Approach

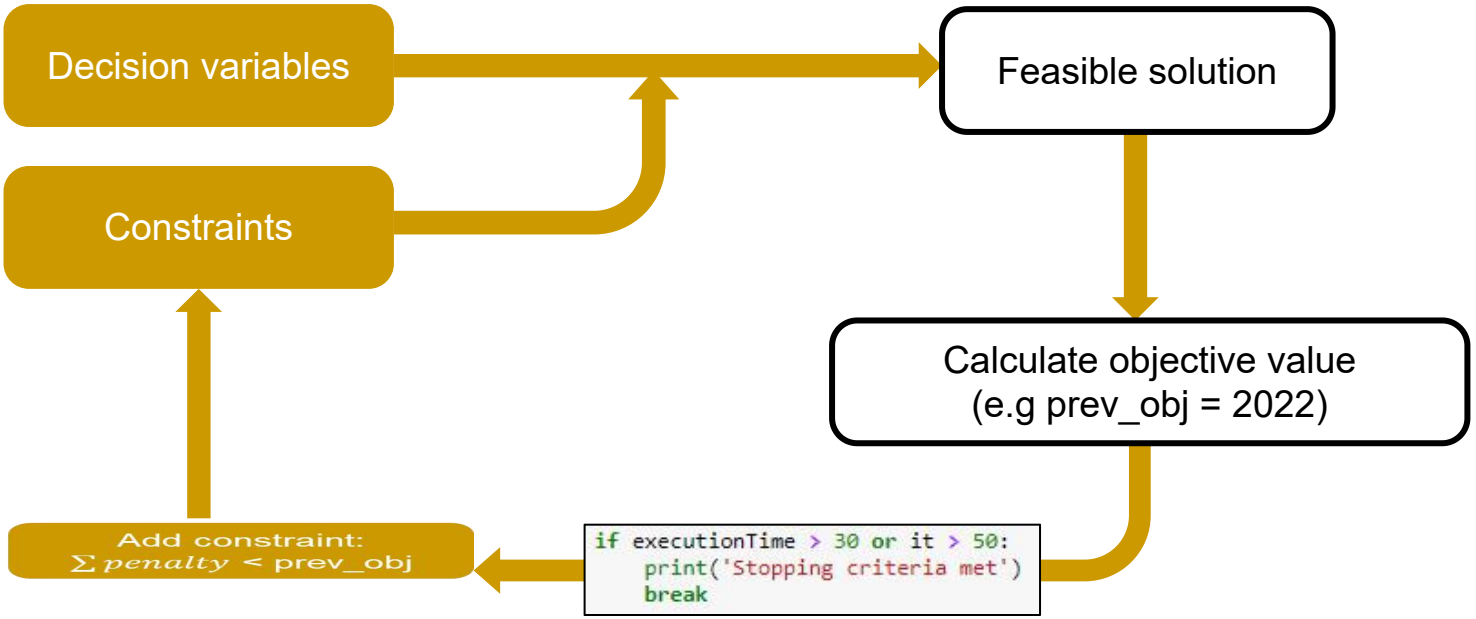
- Mixed Integer Programming




```
m.add_constraint(time_penalty + room_penalty <= sol_penalty-1)
```

3. Solution Approach

- Mixed Integer Programming



Stopping criteria: either
- number of iterations > 50
- time for an iteration exceed 30s

3. Solution Approach

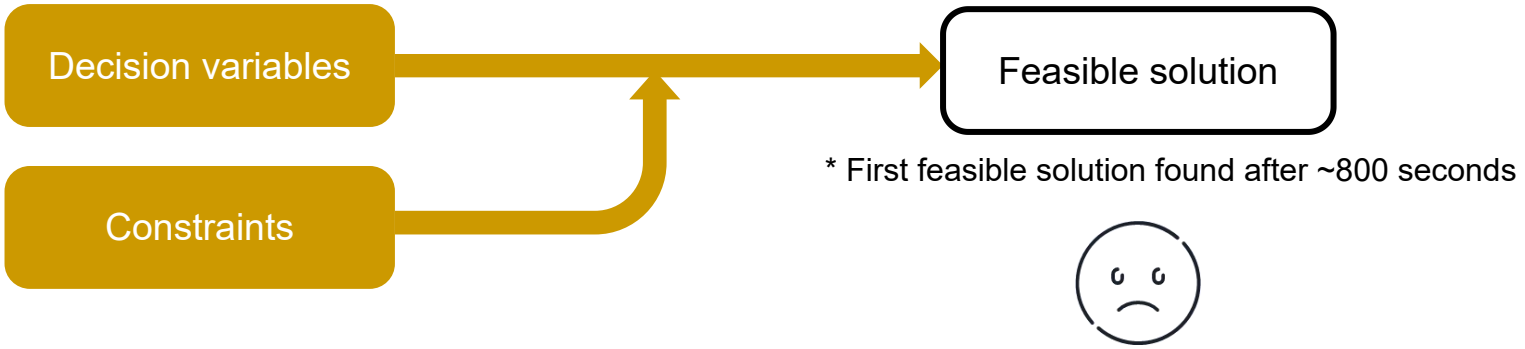
- Mixed Integer Programming

Algorithm 1 MIP approach general structure

$sol_{pre} \leftarrow None$
while not meet the stopping criteria **do**
  $sol \leftarrow model.solve()$
  if sol **then**
    $sol_{pre} \leftarrow sol$
  end if
  $penalty \leftarrow objective(sol_{pre})$
  $p_{time} \leftarrow m.sum(pt_{c,t} * x_{c,t} \ for \ c \in C, t \in CTime_c)$
  $p_{room} \leftarrow m.sum(pr_{c,r} * y_{c,r} \ for \ c \in C, r \in CRoom_c)$
  $model.add\_constraint(p_{time} + p_{room} <= penalty)$ ▷ New solution must be better than current one
end while

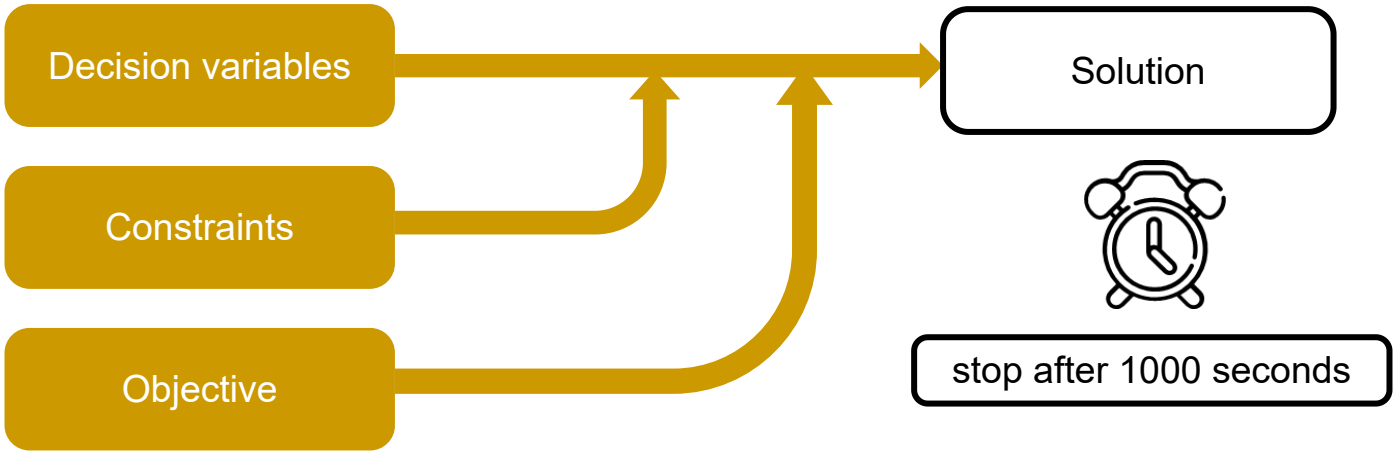
3. Solution Approach

- Constraint Programming



3. Solution Approach

- Constraint Programming

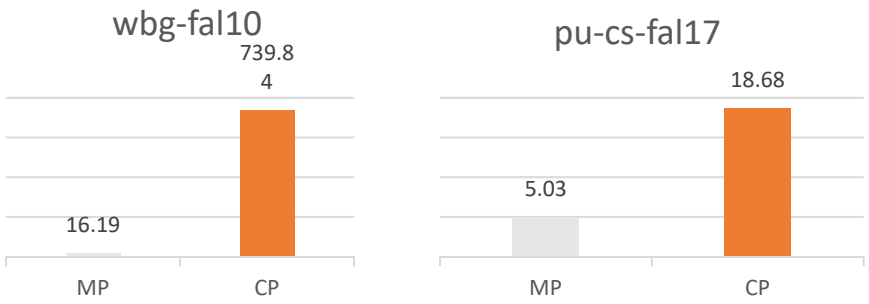


4. RESULTS

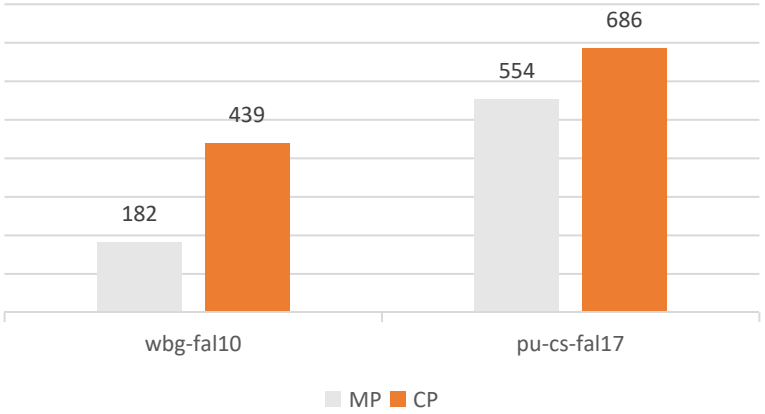
4.Results

First feasible solution

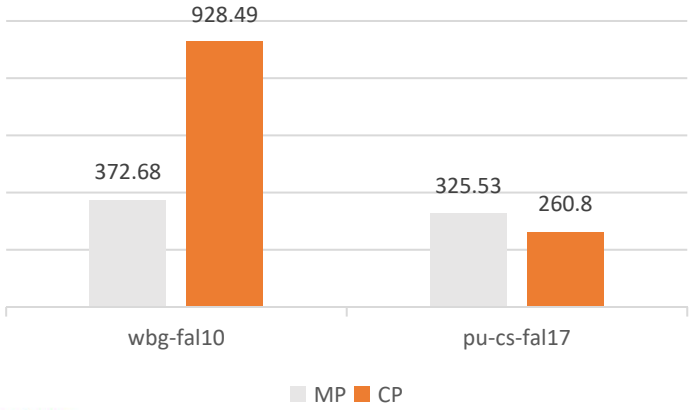
Runtime

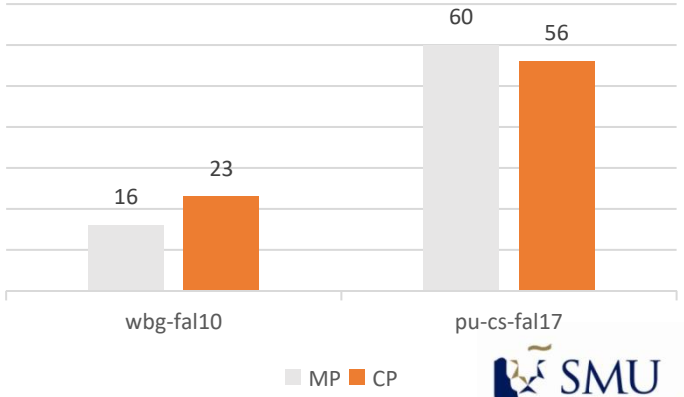


Objective



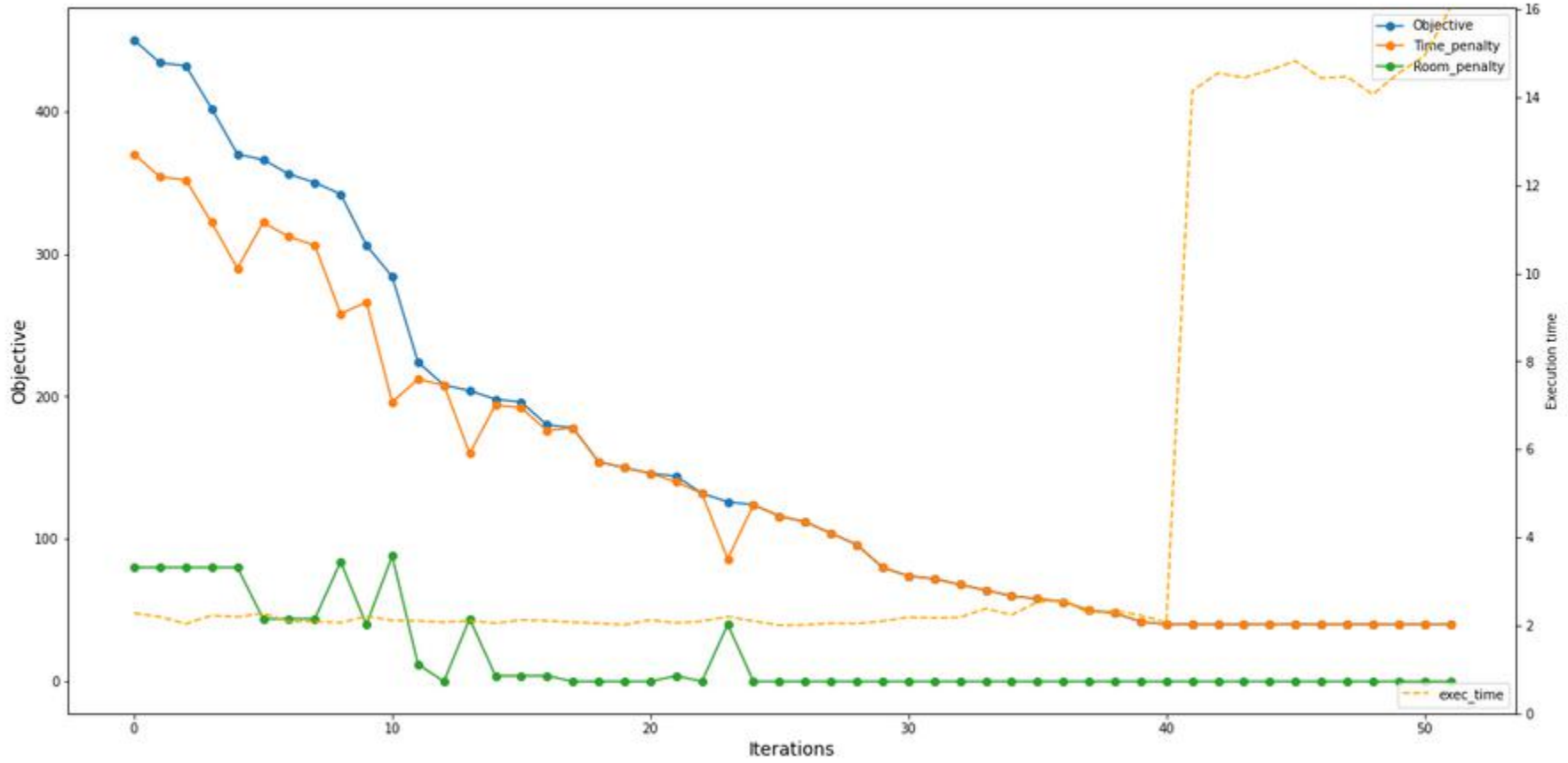
Optimal solution





MIP approach result

```
Total runtime: 249.4854 seconds
Objective value: 40.0
```



Validation

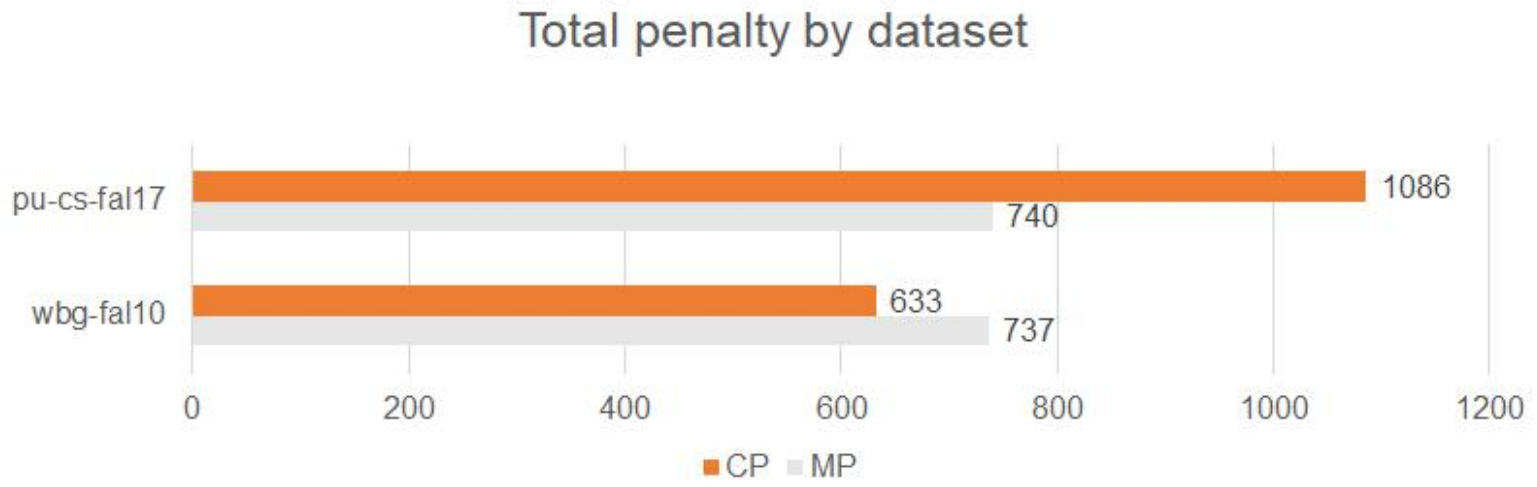
ITC 2019

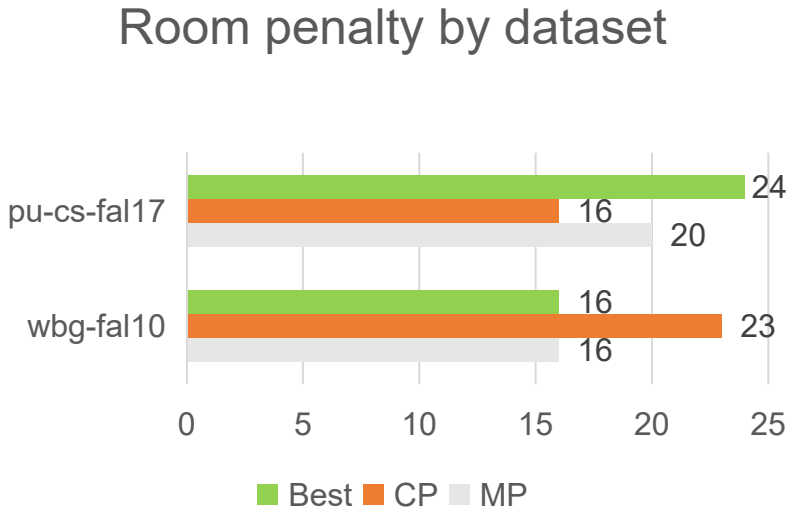
Summary

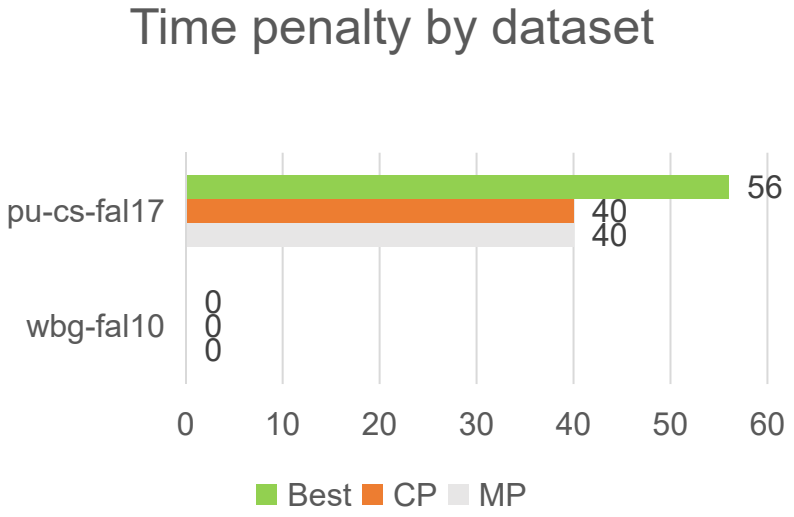
Variables assigned:	150/150
Enrollment problems detected:	0
Total time penalty:	0
Total room penalty:	16
Total distribution penalty:	13
Total student conflicts:	58
Valid solution:	true
Total penalty:	737

```
▼<solution name="wbg-fal10" runtime="12.3" cores="4" technique="MIP" author="G1" institution="SMU">
  ▼<class id="1" days="1010100" start="102" weeks="1111111111111111" room="3">
     <student id="1"/>
     <student id="2"/>
   </class>
  ▼<class id="2" days="1111100" start="138" weeks="1111111111111111" room="6">
     <student id="7"/>
     <student id="12"/>
   </class>
  ▼<class id="3" days="0101000" start="90" weeks="1111111111111111" room="6">
     <student id="4"/>
     <student id="5"/>
     <student id="6"/>
     <student id="8"/>
   </class>
  ▼<class id="4" days="0101000" start="102" weeks="1111111111111111" room="6">
     <student id="11"/>
     <student id="14"/>
     <student id="15"/>
     <student id="16"/>
   </class>
```

Validation


Total penalty by dataset
pu-cs-fal17
wbg-fal10
1086
740
633
737
0
200
400
600
800
1000
1200
CP
MP


Room penalty by dataset
pu-cs-fal17
wbg-fal10
24
16
20
16
23
16
0
5
10
15
20
25
Best
CP
MP


Time penalty by dataset
pu-cs-fal17
wbg-fal10
56
40
40
0
0
0
0
10
20
30
40
50
60
Best
CP
MP

5. Demo

- Colab: MIP
- Colab: CP

Q&A