# University Course Classroom Assignment During a Pandemic

Mohammad Khamechian, PhD Candidate
Matthew Petering, Associate Professor
Department of Industrial and Manufacturing Engineering, College of Engineering and Applied Science

## INTRODUCTION

University course classroom assignment is very challenging and highly constrained real-word problem. Many published papers have investigated this problem from different angles. This year the pandemic made this task harder due to reduced classroom capacities in order to implement the social distancing.

This research investigates university course classroom assignment with limited capacities for classrooms and the potential for more than one classroom to be used for the same course. A mathematical model is developed and coded in C++ and solved with CPLEX.

## PROBLEM DESCRIPTION

In this problem, each classroom has a lower capacity to be able to follow CDC social distancing regulations. Assumptions considered in this model are as follows:

- The set of courses needing classrooms (i.e., rooms) is given.
- The (expected) number of students taking each course is given.
- The pandemic-adjusted room capacities are precomputed and given as an input parameter.
- Eligible time slots and rooms for each course are given. (Some courses cannot be taught in some classrooms.)
- The desired number of face-to-face meetings for each course in each week is given by professors and lecturers but it is not guaranteed to be fully satisfied.
- Distance between classrooms assigned to the same course is considered.

The practical constraints in this model are as follows:

- No more than one course is allowed to meet face to face in the same room in time slot *t* during any week *w*.
- Combined capacity of all rooms assigned to a course must be greater than the number of students in that course.
- The requested number of face-to-face meetings for course *c* is at most three per week.
- If course *c* is scheduled in both rooms *r* and *s*, the distance between rooms must be less than or equal to maximum distance between rooms assigned to course *c*.

There are two objectives:

1. Minimize the deviation between the requested and scheduled face-to-face meetings (both in terms of timing and number of meetings)
2. Minimize the maximum distance between rooms assigned to the same course

## CONTACT INFORMATION

Corresponding author: Mohammad Khamechian
Email: Khamech@uwm.edu

## METHODOLOGY:

The main methodology used in this research is mathematical optimization. An integer programming model is developed to solve the defined problem.

Mathematical model:

| Indices | |
|---|---|
| $r, s$ | Rooms |
| $c$ | Courses |
| $w, k$ | Weeks |
| $t$ | Time slots (8 am-8 pm, 2 hours, Fridays till 6 pm, 28 slots per week) |
| **Parameters** | |
| $R$ | Total number of available rooms |
| $C_r$ | Room capacity (during COVID-19) |
| $C$ | List of courses which at least has one face to face meeting |
| $W$ | Total number of available weeks (e.g., 16) |
| $T$ | Total number of available time slots (e.g., 28) |
| $S_c$ | Number of students enrolled in course c |
| $DN_{cw}$ | Desired number of face-to-face meetings for course *c* that is desired to take place during week *w* |
| $CuDN_{cw}$ | Cumulative number of desired face to face meetings for course c that is desired to take place during week w |
| $J_{cr}$ | = 1 if course c is eligible to be scheduled in room r<br>= 0 otherwise (binary) |
| $I_{ct}$ | = 1 if course *c* is eligible to be scheduled in time slot *t*<br>= 0 otherwise (binary) |
| $\alpha$ | Weight for part of the objective function |
| **Decision variables** | |
| $X_{ctw}$ | = 1 if course *c* has a face-to-face meeting during time slot t in week *w*<br>= 0 otherwise (binary) |
| $Y_{cr}$ | = 1 if room *r* is used for course *c*<br>= 0 otherwise (binary) |
| $AN_{cw}$ | The actual number of face-to-face meetings for course *c* that is scheduled to take place during week *w* |
| $CuAN_{cw}$ | Cumulative number of the actual face to face meetings for course *c* that is scheduled to take place during week *w* |
| $Diff_{cw}$ | Deviation between actual and desired number of face-to-face meetings during week *w* or earlier for course *c* |
| $Z_{crtw}$ | = 1 if course *c* meets face to face in room *r* during time slot *t* in week *w*<br>= 0 otherwise |
| $V_{ct}$ | =1 if course *c* is eligible to meet face to face during time slot *t*<br>=0 otherwise (binary) |
| $M_c$ | Max distance between rooms that assigned to course *c* |

**Objective function**

$$\text{Min } (\alpha * \sum_{c=1}^{C}\sum_{w=1}^{W} Diff_{cw} * S_c) + (1-\alpha) * (\sum_{c=1}^{C} M_c * S_c))$$

**Constraints**

| | | |
|---|---|---|
| $\sum_{c=1}^{C} Z_{crtw} \le 1$ | for all r, t, and w | (1) |
| $\sum_{r=1}^{R} C_r * Y_{cr} \ge S_c$ | for all *c* | (2) |
| $V_{ct} \le I_{ct}$ | for all *c*, *t* | (3) |
| $X_{ctw} \le V_{ct}$ | for all *c*, *t* and *w* | (4) |
| $Y_{cr} \le J_{cr}$ | for all *c*, *r* | (5) |
| $Z_{crtw} \le Y_{cr}$ | for all *c*, *r*, *t*, and *w* | (6) |
| $Z_{crtw} \le X_{ctw}$ | for all *c*, *r*, *t*, and *w* | (7) |
| $Z_{crtw} \ge Y_{cr} + X_{ctw} - 1$ | for all *c*, *r*, *t*, and *w* | (8) |
| $\sum_{w=1}^{W} AN_{cw} \le \sum_{w=1}^{W} DN_{cw}$ | for all *c* | (9) |
| $D_{rs} \le Mc + (BigM)*(2 - Y_{cr} - Y_{cs})$ | for all *c*, *r*, and *s* | (10) |
| $Diff_{cw} \ge CuDN_{cw} - CuAN_{cw}$ | for all *c* and *w* | (11) |
| $Diff_{cw} \ge CuAN_{cw} - CuDN_{cw}$ | for all *c* and *w* | (12) |
| $\sum_{t=1}^{T} X_{ctw} = AN_{cw}$ | for all *c* and *w* | (13) |
| $CuAN_{cw} = \sum_{k=1}^{w} AN_{ck}$ | for all *c* and *w* | (14) |
| $\sum_{t=1}^{T} V_{ct} \le 3$ | for all *c* | (15) |
| $Z_{crtw} \le V_{ct}$ | for all *c*, *r*, *t*, and *w* | (16) |
| $CuAN_{cw} \ge 1$ | for all *c* | (17) |

## RESULTS

We created a small fictional instance with 15 courses, 10 rooms, and 4 weeks. In this problem instance each day has 3 time slots which makes a total of 15 time slots per week. Altogether, the number of Z decision variables is (15*10*3*4) = 1800.

We consider two scenarios: (1) all rooms and time slots can be used for all courses and (2) 80% of rooms and time slots can be used for each course. In each scenario, the value of α is 0.9.

Each scenario was solved to optimality by IBM ILOG CPLEX 12.10 within one minute. The optimal value for scenario 1 was 380.19. In the optimal solution, the value of objective 1 is 10, and the value of objective 2 is 3711.87. The optimal value for scenario 1 was 484.92. In the optimal solution, the value of objective 1 is 10, and the value of objective 2 is 4759.2.

The tables below show the optimal schedules for the two scenarios. In each table, the four main columns indicate the weeks. Each week is divided to 5 smaller columns which represent the days, and each day is divided to 3 very small columns to show the time slots. There is one row for each room. A 0 in the table means no course is scheduled for that room and time slot. Each highlighted cell shows the number of the course scheduled for a given room and time slot. For example, the value 4 appears in 5 highlighted cells at the bottom of the first table. This means course number 4 is scheduled in room 10 and time slots 1-2 in week 1 and room 10 and time slots 1, 2, and 15 in week 3.

Schedule for scenario 1 (no room or time slot limitations)

| Weeks | Week 1 | | | | | Week 2 | | | | | Week 3 | | | | | Week 4 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Time slots | Monday | Tuesday | Wednesday | Thursday | Friday | Monday | Tuesday | Wednesday | Thursday | Friday | Monday | Tuesday | Wednesday | Thursday | Friday | Monday | Tuesday | Wednesday | Thursday | Friday |
| Rooms | 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 | | | | | | | | | | | | | | | | | | | |
| 1 | 14 7 0 0 0 14 0 0 0 0 7 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 | | | | |
| 2 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 7 0 0 0 14 0 0 0 0 0 0 0 0 0 0 | | | | |
| 3 | 8 0 0 12 13 10 10 10 0 0 0 0 0 0 0 | | | | | 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 3 0 0 6 11 0 10 10 10 0 0 0 12 0 0 | | | | | 0 8 0 0 0 10 10 10 0 0 0 8 0 0 8 | | | | |
| 4 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 5 | 9 5 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 5 0 15 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 15 15 0 0 0 0 0 9 0 0 0 0 0 15 0 | | | | | 5 5 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 6 | 8 5 12 13 10 10 10 0 0 0 0 0 0 0 0 | | | | | 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 6 11 0 10 10 10 0 0 0 12 0 0 0 0 0 | | | | | 6 0 0 10 10 10 0 0 0 8 0 5 5 8 0 | | | | |
| 7 | 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 8 | 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 2 2 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 9 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 10 | 4 4 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 4 4 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |

Schedule for scenario 2 (room and time slot limitations)

| Weeks | Week 1 | | | | | Week 2 | | | | | Week 3 | | | | | Week 4 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Time slots | Monday | Tuesday | Wednesday | Thursday | Friday | Monday | Tuesday | Wednesday | Thursday | Friday | Monday | Tuesday | Wednesday | Thursday | Friday | Monday | Tuesday | Wednesday | Thursday | Friday |
| Rooms | 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 | | | | | | | | | | | | | | | | | | | |
| 1 | 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 | | | | | 0 15 15 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 15 2 0 0 0 2 0 0 0 0 0 0 15 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 2 | 13 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 6 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 3 | 0 10 5 12 0 0 0 0 0 0 0 0 0 0 0 | | | | | 10 8 10 5 7 0 0 0 0 0 0 0 0 0 0 | | | | | 7 11 10 8 0 12 0 0 0 10 8 10 0 0 0 | | | | | 7 0 0 10 5 10 5 0 0 0 0 0 0 0 0 | | | | |
| 4 | 15 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 | | | | |
| 5 | 9 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 9 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 6 | 0 10 5 12 0 0 0 0 0 0 0 0 0 0 0 | | | | | 10 8 10 5 0 0 3 0 0 0 3 0 0 0 0 | | | | | 10 8 10 0 12 0 0 0 0 0 0 0 0 0 0 | | | | | 10 0 0 10 5 10 5 0 0 0 0 0 0 0 0 | | | | |
| 7 | 0 0 0 0 0 0 14 0 0 0 0 0 0 14 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 14 0 0 0 0 0 0 0 0 0 0 0 14 0 | | | | | 0 0 0 0 0 0 14 0 0 0 0 0 0 0 0 | | | | |
| 8 | 0 0 0 0 0 0 0 0 0 0 0 4 4 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 4 4 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 9 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |
| 10 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | |

## CONCLUSION & FUTURE WORK

The model identifies optimal solutions for small problem instances. In the future we plan to develop a metaheuristic algorithm that can get excellent solutions for large problem instances.