Cinema Seating Planning

Edo Mangelaars, Tijmen van den Pol, Matthijs Wolters, Daniele Di Grandi, Bernd van den Hoek

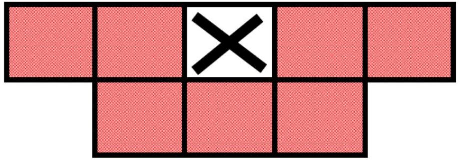
Cinema seating problem

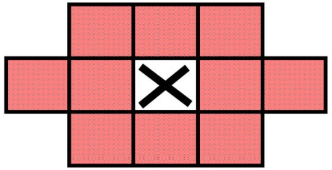
Goal is to optimize amount of people

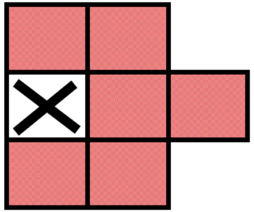
Group sizes can differ from 1 to 8

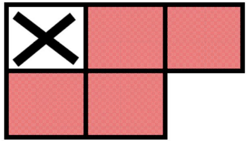
Groups cannot sit in the red areas

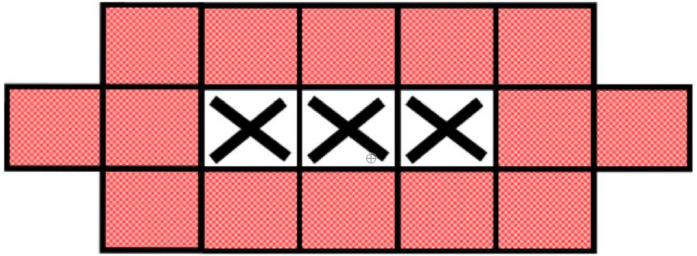
Offline vs Online











OFFLINE CINEMA PROBLEM

ILP APPROACH

ILP SOLUTION APPROACH FOR THE OFFLINE CINEMA PROBLEM

Advantages:

- Using well-known solving techniques like branch and bound and cutting planes, we know for sure that the solution of this model would be optimal because of the optimality proof behind these algorithms
- Implement the model in a well-known commercial solver, in order to be sure that the problem would be solved in the most efficient way, because of the best practice developed by the software owners

Disadvantages:

- Not easy to model and implement
- Modeling structure will directly determine the runtime necessary to solve this problem, especially for large inputs

GUROBI and Python

Commercial Solver choice: GUROBI (license provided by UU)

Programming language choice: Python (library: gurobipy)

Reason: More documentation available respect to other API for other languages

A WALKTHROUGH IN THE MODEL: VARIABLES

VARIABLES:

Cinema grid:

$$y_{ijk} = \begin{cases} 1 & if\ a\ (i,j)\ seat\ is\ assigned\ to\ a\ group\ of\ size\ k \\ 0 & otherwise \end{cases}$$

$$x_{ijk} = \begin{cases} 1 & if\ a\ k\ size\ group\ starts\ seating\ from\ the\ position\ (i,j) \\ 0 & otherwise \end{cases}$$

x_{111} y_{111} x_{112} y_{112} … x_{118} y_{118}	x_{121} y_{121} x_{122} y_{122} … x_{128} y_{128}	…	x_{1m1} y_{1m1} x_{1m2} y_{1m2} … x_{1m8} y_{1m8}
x_{211} y_{211} x_{212} y_{212} … x_{218} y_{218}	x_{221} y_{221} x_{222} y_{222} … x_{228} y_{228}	…	x_{2m1} y_{2m1} x_{2m2} y_{2m2} … x_{2m8} y_{2m8}
…	…	…	…
x_{n11} y_{n11} x_{n12} y_{n12} … x_{n18} y_{n18}	…	…	x_{nm1} y_{nm1} x_{nm2} y_{nm2} … x_{nm8} y_{nm8}

A WALKTHROUGH IN THE MODEL: OBJECTIVE FUNCTION

OBJECTIVE FUNCTION:

Maximize the total amount of people in the cinema that is equal to maximizing the place where each group starts sitting times the size of that group:

$$\max z = \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{k=1}^{8} k x_{ijk}$$

Note that we can’t simply maximize this because of the way the constraints of this model are defined.

A WALKTHROUGH IN THE MODEL: CONSTRAINTS (1)

1) Preprocessing constraints: for (i,j) positions without a chair (0 in the given input), we have to enforce the corresponding y_{ijk} variables to be equal to 0, because no one could sit there:

$$\sum_{k=1}^{8} y_{ijk} = 0 \quad \forall i = 1, \dots, n; \forall j = 1, \dots, m \text{ } where \text{ } (i,j) = 0 \text{ } in \text{ } the \text{ } given \text{ } input$$

2) The amount of people seated have to be less or equal than the total people given in input:

$$\sum_{i=1}^{n} \sum_{j=1}^{m} y_{ijk} \leq k s_k \qquad \forall k = 1, \dots, 8$$

Where s_k is: how many groups of size k that have to be placed

A WALKTHROUGH IN THE MODEL: CONSTRAINTS (2)

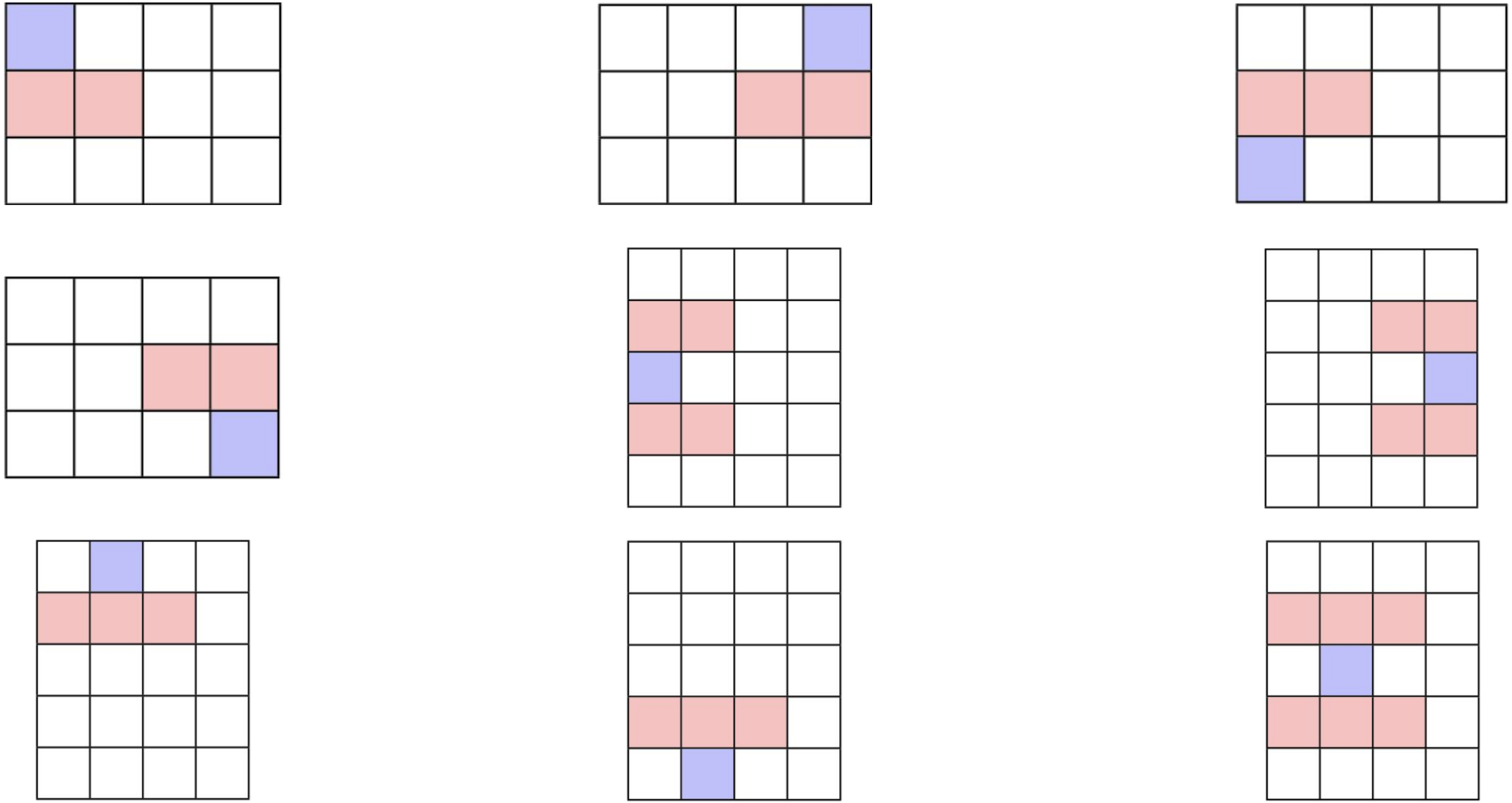
3) Constraints regarding the 1.5m distance rule. These constraints have been split in two parts: the first part yields that if some $y_{ijk} = 1$ it means that a person is seated in that chair, therefore, every chair in front, under and diagonally should be forced to be equal to 0.

FIRST PART

This first part also has been split in 9 sets of constraints. The problem of writing them in only 1 set of constraints was the dependence of these constraints by the position of where they have to be written.

We call that: patterns and we have 9 individual patterns.

A WALKTHROUGH IN THE MODEL: CONSTRAINTS (3)



A WALKTHROUGH IN THE MODEL: CONSTRAINTS (4)

For each pattern a constraint:

$$M\left(1-\sum_{k=1}^{8} y_{ijk}\right) \geq \sum_{k=1}^{8} y_{i+1,j,k} + \sum_{k=1}^{8} y_{i+1,j+1,k} \quad for\ i=1;\ j=1$$

$$M\left(1-\sum_{k=1}^{8} y_{ijk}\right) \geq \sum_{k=1}^{8} y_{i+1,j,k} + \sum_{k=1}^{8} y_{i+1,j-1,k} \quad for\ i=1;\ j=m$$

$$M\left(1-\sum_{k=1}^{8} y_{ijk}\right) \geq +\sum_{k=1}^{8} y_{i-1,j,k} + \sum_{k=1}^{8} y_{i-1,j+1,k} \quad for\ i=n;\ j=1$$

$$M\left(1-\sum_{k=1}^{8} y_{ijk}\right) \geq +\sum_{k=1}^{8} y_{i-1,j,k} + \sum_{k=1}^{8} y_{i-1,j-1,k} \quad for\ i=n;\ j=m$$

$$M\left(1-\sum_{k=1}^{8} y_{ijk}\right) \geq \sum_{k=1}^{8} y_{i+1,j,k} + \sum_{k=1}^{8} y_{i-1,j,k} + \sum_{k=1}^{8} y_{i+1,j+1,k} + \sum_{k=1}^{8} y_{i-1,j+1,k} \quad for\ j=1;\ \forall i=2,\dots,n-1$$

$$M\left(1-\sum_{k=1}^{8} y_{ijk}\right) \geq \sum_{k=1}^{8} y_{i+1,j,k} + \sum_{k=1}^{8} y_{i-1,j,k} + \sum_{k=1}^{8} y_{i-1,j-1,k} + \sum_{k=1}^{8} y_{i+1,j-1,k} \quad for\ j=m,;\forall i=2,\dots,n-1$$

$$M\left(1-\sum_{k=1}^{8} y_{ijk}\right) \geq \sum_{k=1}^{8} y_{i+1,j,k} + \sum_{k=1}^{8} y_{i+1,j+1,k} + \sum_{k=1}^{8} y_{i+1,j-1,k} \quad for\ i=1;\ \forall j=2,\dots,m-1$$

$$M\left(1-\sum_{k=1}^{8} y_{ijk}\right) \geq \sum_{k=1}^{8} y_{i-1,j,k} + \sum_{k=1}^{8} y_{i-1,j-1,k} + \sum_{k=1}^{8} y_{i-1,j+1,k} \quad for\ i=n;\ \forall j=2,\dots,m-1$$

$$M\left(1-\sum_{k=1}^{8} y_{ijk}\right) \geq \sum_{k=1}^{8} y_{i+1,j,k} + \sum_{k=1}^{8} y_{i-1,j,k} + \sum_{k=1}^{8} y_{i-1,j-1,k} + \sum_{k=1}^{8} y_{i+1,j+1,k} + \sum_{k=1}^{8} y_{i-1,j+1,k} + \sum_{k=1}^{8} y_{i+1,j-1,k}$$

$$\forall i=2,\dots,n-1;\ \forall j=2,\dots,m-1$$

A WALKTHROUGH IN THE MODEL: CONSTRAINTS (5)

SECOND PART

This second part yields that if a chair is selected to let seat-down a group of size k, the 2 chairs after that group (namely, the 2 chairs after k people) have to be forced equal to 0:

$$N(1 - x_{ijk}) \geq \sum_{s=1}^{8} \sum_{L=0}^{\begin{cases} 1 \; if \; m-j-k>0 \\ 0 \; otherwise \end{cases}} y_{i,j+k+L,s} \quad \forall i = 1, \dots, n; \; \forall j = 1, \dots, m-k; \; \forall k = 1, \dots, 8$$

Where N is a number big enough to not put a wrong constraint if $x_{ijk} = 0$. In this case: N = 16 is sufficient.

A WALKTHROUGH IN THE MODEL: CONSTRAINTS (6)

4) People in the same group have to be placed near each other:

$$(k-1)x_{ijk} \leq \sum_{L=1}^{k-1} y_{i,j+L,k} \quad \forall i = 1, \dots, n;\ \forall j = 1, \dots, m-1;\ \forall k = 2, \dots, \min(m+1-j, 8)$$

These constraints are written in order to not exceed the length of the cinema by writing them for j that goes till m-1 and for k with a min function

AFTER THE IMPLEMENTATION ON GUROBI

- The model can find an optimal solution in a reasonable time only for not too large inputs
- it can also handle larger input by being stopped while running: in this way, it will output the best solution found till that moment, but for very large cinemas (n,m > 250) it doesn’t find a first solution.

Though, consider the real sizes of real cinemas, this ILP model it is fully functional and will always find the optimal solution in reasonable time. However, in order to satisfy all the input sizes permitted, we need to find something that can handle inputs till n=1000 and m=1000.

ALGORITHMIC APPROACH

Algorithmic approach

Advantages:

- Much faster than the ILP solver
- Can handle bigger inputs
- Many different approaches possible

Disadvantages:

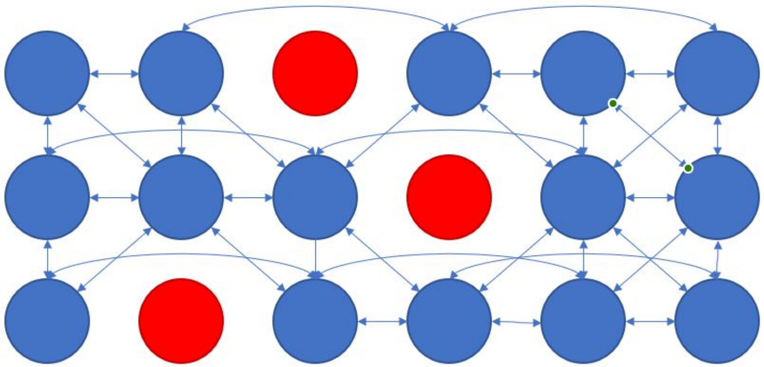
- Not all approaches guaranteed to be optimal
- Optimal approaches are slow

First Fit

Greedy algorithm

Cinema as graph-like structure

- Nodes are seats
- Edges to every seat blocked when occupying a seat



First Fit

Start with the biggest group available.

Search the cinema row by row.

Enough adjecent seats? Place group.

Not enough open seats? Discard any group of this size.

Solutions not optimal

Non optimality of First Fit

1. Does not consider the location of placement

Input	FF result	Optimal
11111	XXXXX	XXXXX
11111	11111	11111
11011	11011	X10XX
11111	XXXXX	11111
11111	11111	XXXXX
1 1 0 0 2 0 0 0		

2. Does not try different combinations of groups

Input	FF result	Optimal
100	100	X00
111	XX1	11X
111	111	X11
001	00X	00X
4 1 0 0 0 0 0 0		

First Fit Ordered Rows

Search is similar to First Fit

Order rows by least amount of edges to adjacent rows

More general bound

Overfit → no guarantee a row with few edges will contain optimal position

Slow → Sorting the list of rows

Least Blocked Seats

Place each group in the position that blocks the fewest seats

Search through the whole cinema per group

Keep track of location that blocks the least amount of seats

Combining ILP and Algorithmic Approach

Solution from an ALG approach could be used as the first solution of the ILP solver

Could help decrease runtime

Sadly no time to implement this

OFFLINE CINEMA PROBLEM

RESULTS COMPARISON

RESULTS ON INTERESTING CINEMAS (1)

LAYOUT:

```
7
12
000010111110
000110111111
000000000000
011110111111
111110111111
111000001111
111000001111
2 2 1 1 0 1 0 0
```

Number of seats: 49

ILP:

```
000010111110
000xx011x11x
000000000000
011110xxxxxx
xxxx10111111
111000001111
xxx00000xx11
```

NPS: 19

Runtime: 0.0334

GFO: 0.0 %

FIRSTFIT:

```
0000X0111110
000110XXXXXX
000000000000
0XXXX01XXX11
11111011111X
XX100000XX11
111000001111
```

NPS: 19

Runtime: 0.007

GFO: 0.0 %

LeastBlockFit:

```
0000X0111110
000110XXXXXX
000000000000
011XX0111111
XX1110X11111
111000001111
XXX00000XXXX
```

NPS: 19

Runtime: 0.000

GFO: 0.0 %

In this particular case, all the algorithm gave the same optimal answer, but note that the final layouts are all different

RESULTS ON INTERESTING CINEMAS (2)

LAYOUT:

```
15
15
110111011101110
101110111011101
011101110111011
111011101110111
110111011101110
101110111011101
011101110111011
111011101110111
110111011101110
101110111011101
011101110111011
111011101110111
110111011101110
101110111011101
011101110111011
20 20 6 0 0 0 0 0 0
```

Number of seats: 168

ILP:

```
xx01xx01xx01xx0
101110111011101
0xx10xx10xxx01x
111011101110111
xx01xx01xx01xx0
101110111011101
0xx10xxx01x10xx
111011101110111
xx01xx01xx01xx0
101110111011101
0xxx01x10xx10xx
111011101110111
x10xx10xx10xxx0
101110111011101
0xxx01x10xxx01x
```

NPS: 64

Runtime: 2.10301

GFO: 0.0 %

FIRSTFIT:

```
X10XXX01X10XXX0
101110111011101
0XXX01X10XXX01X
111011101110111
X10XXX01110XXX0
1011101XX011101
0XX1011101XX011
1110XX1011101XX
XX01110XX101110
101XX01110XX101
011101XX01110XX
XX1011101XX0111
110XX1011101XX0
101110XX1011101
0XX101110XX10XX
```

NPS: 61

Runtime: 0.007

GFO: 4.69 %

LeastBlockFit:

```
X10XXX01110XXX0
1011101XX011101
0XXX011101XX011
11101XX011101XX
XX011101XX01110
101XX011101XX01
011101XX0111011
XX1011101XX01XX
110XX1011101110
101110XX10XX101
0XX1011101110XX
1110X110XX10111
XX011101110XXX0
101110XX1011101
0XXX01110XXX01X
```

NPS: 61

Runtime: 0.000

GFO: 4.69 %

RESULTS ON INTERESTING CINEMAS (3)

LAYOUT:

```
13
31
0000000111111111111111110000000
0000111100000001000000011110000
0011110001111111111111000111100
0111000111000001000001110001110
1110011100011111111100011100111
1100011000110000000110001100011
1100111000110000000000000000000
1100011000110000000110001100011
1110011100011111111100011100111
0111000111000000000001110001110
0011110001111111111111000111100
0000111100000000000000011110000
0000000111111111111111110000000
60 6 6 6 0 0 0 0
```

Number of seats: 191

ILP:

```
000000011x11xx111x11xxx10000000
0000xxxx0000000x00000001xxx0000
00x111000xx11x111xxx11000111x00
0111000x1100000x00000x110001110
xxx00x110001xx111xx1000x1100x11
1100011000x100000001x0001x0001x
xx00xxx000110000000000000000000
1100011000x100000001x0001x0001x
xxx00x110001xxxx11x1000x1100x11
0111000x1100000000000x110001110
00xxxx000xxxx11xx11x11000111x00
0000111x000000000000000xxxx0000
000000011x11x11x11xxxx110000000
```

NPS: 86

Runtime: 5.45584

GFO: 0.0 %

FIRSTFIT:

```
0000000XXXX11XXXX11XXXX10000000
000011110000000100000001XXX0000
00XXXX000XXXX11XXXX111000111100
0111000111000001000000XXX000XXX0
XXX00XXX000XXX11XX11000111 00111
11000110001100000000XX000XX000XX
XX00XX1000X10000000000000000000
11000110001100000000X1000X1000X1
X1100X11000X11X11X1100011100111
01X1000X1100000000000X11000X110
0011X1000X11X11X11X111000X11100
000011X1000000000000000X1110000
00000001X11X11X11X11X1110000000
```

NPS: 80

Runtime: 0.006

GFO: 6.98 %

LeastBlockFit:

```
0000000XXXX11XXXX11XXXX10000000
0000XX1100000001000000 01XXX0000
00X111000XXXX11XXXX111000111X00
0111000111000001000000XXX0001110
X1100XXX000XXXX11XXX00011100X11
1100011000110000000110000XX0001X
X100XXX000XX0000000000000000000
1100011000110000000XX000X1000X1
X1100XXX000XX11XX111000111 00111
011100011X000000000001X10001110
00X11100011X11X11X11X1000111X00
0000X111000000000000000X11X0000
0000000X11X11X11X11X11110000000
```

NPS: 79

Runtime: 0.000

GFO: 8.14 %

This is the only instance that LeastBlockFit performed worse than FirstFit

RESULTS ON INTERESTING CINEMAS (4)

LAYOUT:

```
17
29
11111111101111111110111111111
11111111101111111110111111111
11111111101111111110111111111
11111111101111111110111111111
11111111101111111110111111111
00000000000000000000000000000
11111111101111111110111111111
11111111101111111110111111111
11111111101111111110111111111
11111111101111111110111111111
11111111101111111110111111111
00000000000000000000000000000
11111111101111111110111111111
11111111101111111110111111111
11111111101111111110111111111
11111111101111111110111111111
11111111101111111110111111111
7 15 9 10 5 4 0 6
```

Number of seats: 405

ILP:

```
xxxxxx1110xxxxxxxx10xxxx11xxx
1111111xx01111111110111111111
xxxxx11110xxx11xxxx01xxxxxxxx
111111xxx01111111110111111111
xxxxx11110xxxxx11xx01xxxxxxxx
00000000000000000000000000000
xx111xxxx011xxxx1110xx11xxxxx
111x111110x111111xx0111111111
xx111xxxx0111xxx1110xxxx11xxx
111x111110xx11111xx0111111111
xx111xxxx0111xxx1110xxxx11xxx
00000000000000000000000000000
xxxxxxxx10xxxxx11xx0111xxxxxx
11111111101111111110xx1111111
xxxxxxxx10xxx11xxxx0111xxxxxx
11111111101111111110xx1111111
xxxx11xxx01xxxxxxxx0111xxxxxx
```

NPS: 191

Runtime: 1673.21 (27 min)

GFO: 0.0 %

FIRSTFIT:

```
xxxxxxxx10xxxxxxxx10xxxxxxxx1
11111111101111111110111111111
xxxxxxxx10xxxxxxxx10xxxxxxxx1
11111111101111111110111111111
xxxxxx1110xxxxxx1110xxxxxx11x
00000000000000000000000000000
xxxxxx1110xxxxx11110xxxxx1111
1111111xx0111111xxx0111111xxx
xxxxx11110xxxxx11110xxxxx1111
111111xxx0111111xxx0111111xxx
xxxx111110xxxx111110xxxx11111
00000000000000000000000000000
xxxx111110xxxx111110xxxx11111
11111xxxx011111xxxx011111xxxx
xxxx111110xxx1111110xxx111111
11111xxx101111xxx1101111xx111
xx11111110xx11111110xx11111xx
```

NPS: 177

Runtime: 0.008

GFO: 7.33 %

LeastBlockFit:

```
xxxxxxxx10xxxxxxxx10xxxxxxxx1
11111111101111111110111111111
xxxxxxxx10xxxxxxxx10xxxxxx111
111111111011111111101111111xx
xxxxxxxx10xxxxxx1110xxxxxx111
00000000000000000000000000000
xxxxxx1110xxxx111110xxxx11111
1111111xx011111xxxx011111xxxx
xxxxx11110xxxx111110xxxx11111
111111xxx011111xxxx011111xxxx
xxxxx11110xxxx111110xxxx11111
00000000000000000000000000000
xxxxx11110xxx11xxx10xxx11xxx1
111111xxx01111111110111111111
xxxxx11110xxx11xx110xx11xx111
111111xxx011111111x01111111xx
xxxxx11110xxx11xx110xx11xx111
```

NPS: 183

Runtime: 0.001

GFO: 4.19 %

RESULTS ON INTERESTING CINEMAS (5)

LAYOUT:

```
39
36
111111110111111111111111101111111111
111111110111111111111001101111111111
111111110001111111111111101111111100
111111110111111111111111101111111100
011111110001111111111111101111111100
011111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111111
111111110000000000000000000000000000
111111110000000000000000000000000000
111111110000000000000000001111111100
111111110111111111111111101111111111
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111110
111111110000000000000000000000000000
111111110000000000000000000000000000
111111110111111111111111101111110000
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
111111110111111111111111101111111100
000111100011111111111111000111100000
000111100011111111111111000111100000
000011100011111111111111000110000000
000001110111111111111111101111100000
000000000000000000000000000000000000
000111000111101110111101111000000000
000111001111111111111111111000000000
001111001111100000111111111111100000
001111101111100000111111111111110000
21 21 21 21 21 21 21 21
```

Number of seats: 1020

ILP:

```
xxxxxxx10xxxxxxxx11xxxxxx0111xxxxxxx
11111111011111111111100110xx11111111
xxxxxxxx000xxxxxx11xxxxxx0111xxxxx00
111111110x1111111111111110xx11111100
0xxxxxxx000xxxxxx11xxxxxx0111xxxxx00
01111111011111111111111110xx11111100
111111110xxxxxxx11xxxxxxx0111xxxxx00
xxxxxxxx011111111111111110xx11111100
111111110xxxxxxx11xxxxxxx0111xxxxx00
xxxxxxxx011111111111111110xx11111100
111111110xxxxxxx11xxxxxxx0111xxxxxxx
xxxxxxxx0000000000000000000000000000
111111110000000000000000000000000000
xxxxxxxx000000000000000000xxxxxxx100
111111110xxxxxxx11xxxxxxx011111111xx
xxxxxxxx011111111111111110xxxxxxx100
111111110xxxxxx111111xxxx01111111100
xxxxxxxx01111111xxxx111110xxxxxxxx00
111111110xxxxxx111111xxxx01111111100
xxxxxxxx01111111xxxx111110xxxxxxxx00
111111110xxxxxx1111111xxx01111111100
xxxxxxxx01111111xxxxx111101xxxxxxxx0
111111110000000000000000000000000000
xxxxxxxx0000000000000000000000000000
111111110xxxxxxx11xxxxxxx01111110000
xxxxxxxx011111111111111110xxxxxxxx00
111111110xxxxxxx1111xxxxx01111111100
xxxxxxxx011111111xx1111110xxxxxxxx00
111111110xxxxxxx111111xxx01111111100
xxxxxxxx011111111xxxx11110xxxxxxxx00
0001111000xxxxxx11111111000111100000
000xxxx0001111111xxxxxxx000xxxx00000
0000111000xxxxxx11111111000110000000
00000xxx011111111xxxxxxx10xxxxx00000
000000000000000000000000000000000000
000xxx000xxxx0111011110xxxx000000000
00011100111111xxxxxxxx11111000000000
00111100xxxxx00000111111xx1111100000
00xxxxx01111100000xxxxx1111xxxxx0000
```

NPS: 488

Runtime: 16541.85 (4.5 hours)

GFO: 0.0 %

FIRSTFIT:

```
XXXXXXXX01XXXXXXXX11XXXX10XXXXXXXX11
11111111011111111111100110111111111X
XXXXXXXX000XXXXXXXX1111110XXXXXXXX00
111111110X1111111111XXXXX01111111100
0XXXXXXX000XXXXXXXX1111110XXXXXXXX00
01111111011111111111XXXXX01111111100
XXXXXXXX01XXXXXXXX11111110XXXXXXXX00
1111111101111111111XXXXXX01111111100
XXXXXXXX01XXXXXXXX11111110XXXXXXXX00
1111111101111111111XXXXXX01111111100
XXXXXXXX01XXXXXXXX11111110XXXXXXXX11
111111110000000000000000000000000000
XXXXXXXX0000000000000000000000000000
11111111000000000000000000XXXXXXXX00
XXXXXXXX01XXXXXXXX11XXXXX0111111111X
11111111011111111111111110XXXXXXX100
XXXXXXX10XXXXXXX11XXXXXXX01111111100
11111111011111111111111110XXXXXXX100
XXXXXXX10XXXXXXX11XXXXXXX01111111100
11111111011111111111111110XXXXXXX100
XXXXXXX10XXXXXXX11XXXXXXX01111111100
11111111011111111111111110XXXXXXX110
XXXXXXX1000000000000000000000000000
111111110000000000000000000000000000
XXXXXXX10XXXXXXX11XXXXXXX01111110000
11111111011111111111111110XXXXXXX100
XXXXXXX10XXXXXXX11XXXXXX101111111100
11111111011111111111111110XXXXXX1100
XXXXXX110XXXXXX11XXXXXX1101111111X00
11111111011111111111111110XXXXXX1100
000XXXX000XXXXXX11XXXXXX000111100000
000111100011111111111111000XXXX00000
0000XXX000XXXXXX11XXXXXX000110000000
00000111011111111111111110XXXXX00000
000000000000000000000000000000000000
000XXX00011110111011110XXXX000000000
00011100XXXXXX11XXXXXX11111000000000
00111100111110000011111XXXXXX1100000
00XXXXX01XXXX00000XXXX11111111XX0000
```

NPS: 466

Runtime: 0.007

GFO: 4.5 %

LeastBlockFit:

```
XXXXXXXX01XXXXXXXX11XXXX10XXXXXXXX11
11111111011111111111100110111111111X
XXXXXXXX000XXXXXX111111110XXXXXXXX00
111111110X11111111XXXXXX101111111100
0XXXXXXX000XXXXXX111111110XXXXXXXX00
011111110111111111XXXXXXX01111111100
XXXXXXX10XXXXXXXX111111110XXXXXXXX00
111111110111111111XXXXXXX01111111100
XXXXXXX10XXXXXXXX111111110XXXXXXXX00
111111110111111111XXXXXXX01111111100
XXXXXXX10XXXXXXXX111111110XXXXXXXX11
111111110000000000000000000000000000
XXXXXXX10000000000000000000000000000
11111111000000000000000000XXXXXXXX00
XXXXXXX10XXXXXXXX11XXXXXX0111111111X
11111111011111111111111110XXXXXXXX00
XXXXXXX10XXXXXXXX11XXXXXX01111111100
11111111011111111111111110XXXXXXXX00
XXXXXXX10XXXXXXX11111111101111111100
11111111011111111XXXXXXX10XXXXXXXX00
XXXXXXX10XXXXXXX11111111101111111100
11111111011111111XXXXXXX10XXXXXXXX10
XXXXXXX10000000000000000000000000000
111111110000000000000000000000000000
XXXXXXX10XXXXXXX11XXXXXX10XXXXXX0000
111111110111111111111111101111111X00
XXXXXXX10XXXXXX11111111110XXXXXX1100
1111111101111111XXXXXX11X01111111X00
XXXXXXX10XXXXXX11111111110XXXXXX1100
1111111101111111XXXXXX11X01111111X00
000XXXX00011111111111111000XXXX00000
0001111000XXXXXX11XXXXXX000111100000
0000XXX00011111111111111000110000000
000001110XXXXXXXX11XXXXXX01XXXX00000
000000000000000000000000000000000000
000XXX00011110111011110XXXX000000000
00011100XXXXXX11XXXXXX11111000000000
001111001111100000111111111XXXX00000
00XXXX10XXXXX00000XXXXXXXX1111110000
```

NPS: 470

Runtime: 0.012

GFO: 3.69 %

RESULTS ON INTERESTING CINEMAS (6)

LAYOUT:

```
31
41
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011111111111111111111111111111111111000
00011110011111111111111111111111001111000
00011110011111111111111111111111001111000
00011110011111111111111111111111001111000
00011110011111111111111111111111001111000
00011110011111111111111111111111001111000
11111110011111111111111111111111001111000
11111110011111111111111111111111001111000
11111110011111111111111111111111001111000
11111110011111111111111111111111001111000
11111110011111111111111111111111001111111
11111110011111111111111111111111001111111
152  100  25  30  20  20  15  16
```

Number of seats: 1065

ILP:

```
000xxxxx11xxxxxxxx11xxxxxxxx11xxxxxxxx000
00011111111111111111111111111111111111000
000xxxxxxx11xxxxxx11xxxxxxxx11xxxxxxxx000
00011111111111111111111111111111111111000
000xxxxxx11xxxxx11xxxxx11xxxxx11xxxxxx000
00011111111111111111111111111111111111000
000xxxxx11xxxxxx11xxxxx11xxxxx11xxxxxx000
00011111111111111111111111111111111111000
000xxxxx11xxxxx11xxxxxx11xxxxx11xxxxxx000
00011111111111111111111111111111111111000
000xxxxxx11xxxxxxxx11xxxxxxxx11xxxxxxx000
00011111111111111111111111111111111111000
000xxxxxxx11xxxxxxxx11xxxxxxxx11xxxxxx000
00011111111111111111111111111111111111000
000xxxxxxxx11xxxxxxx11xxxxxxx11xxxxxxx000
00011111111111111111111111111111111111000
000xxxxxxx11xxxxxxxx11xxxxxx11xxxxxxxx000
00011111111111111111111111111111111111000
000xxxxxxx11xxxxxx11xxxxxxxx11xxxxxxxx000
00011111111111111111111111111111111111000
000xxxx00xxxxx11xxxxxxxx11xxxxxx00xxxx000
00011110011111111111111111111111001111000
000xxxx00xxxxxx11xxxxxxx11xxxxxx00xxxx000
00011110011111111111111111111111001111000
000xxxx00xxxxx11xxxxxx11xxxxx11x00xxxx000
11111110011111111111111111111111001111000
xxxxxxx00xxxxxx11xxxxxx11xxxxxxx00xxxx000
11111110011111111111111111111111001111000
xxxxxxx00xxxxx11xxxxxxxx11xxxxxx00xxxx000
11111110011111111111111111111111001111111
xxxxxxx00xxxxxxx11xxxxxx11xxxxxx00xxxxxxx
```

NPS: 456

Runtime: 515.235 (9 min)

GFO: 18.8 %

FIRSTFIT:

```
000XXXXXXXX11XXXXXXXX11XXXXXXXX1111111000
00011111111111111111111111111111XXXXXX000
000XXXXXXXX11XXXXXXXX11XXXXXXXX1111111000
00011111111111111111111111111111XXXXXX000
000XXXXXXXX11XXXXXXXX11XXXXXXXX1111111000
00011111111111111111111111111111XXXXXX000
000XXXXXXXX11XXXXXXXX11XXXXXXXX1111111000
00011111111111111111111111111111XXXXXX000
000XXXXXXXX11XXXXXXXX11XXXXXXXX1111111000
00011111111111111111111111111111111111000
000XXXXXXXX11XXXXXXX11XXXXXXX11XXXXXXX000
00011111111111111111111111111111111111000
000XXXXXXX11XXXXXXX11XXXXXXX11XXXXXXX1000
00011111111111111111111111111111111111000
000XXXXXXX11XXXXXXX11XXXXXXX11XXXXXXX1000
00011111111111111111111111111111111111000
000XXXXXXX11XXXXXXX11XXXXXXX11XXXXXXX1000
00011111111111111111111111111111111111000
000XXXXXX11XXXXXX11XXXXXX11XXXXXX11111000
0001111111111111111111111111111111XXXX000
000XXXX00XXXXXX11XXXXXX11XXXXXX1001111000
0001111001111111111111111111111100XXXX000
000XXXX00XXXXXX11XXXXXX11XXXXXX1001111000
0001111001111111111111111111111100XXXX000
000111100XXXXXX11XXXXXX11XXXXXX1001111000
XXXXXX1001111111111111111111111100XXXX000
111111100XXXXXX11XXXXXX11XXXXX11001111000
XXXXX11001111111111111111111111X00XXXX000
111111X00XXXXX11XXXXX11XXXXX1111001111000
XXXXX110011111111111111111111XXX00XXXXX11
111111X00XXXXX11XXXXX11XXXXX111100111111X
```

NPS: 438

Runtime: 0.008

GFO: 22 %

LeastBlockFit:

```
000XXXXXXXX11XXXXXXX11XXXXXXX11XXXXXXX000
00011111111111111111111111111111111111000
000XXXXXXXX11XXXXXXX11XXXXXXX11XXXXXXX000
00011111111111111111111111111111111111000
000XXXXXXXX11XXXXXXX11XXXXXXX11XXXXXXX000
00011111111111111111111111111111111111000
000XXXXXXXX11XXXXXXX11XXXXXXX11XXXXXX1000
00011111111111111111111111111111111111000
000XXXXXXXX11XXXXXX11XXXXXX11XXXXXX111000
000111111111111111111111111111111111XX000
000XXXXXXXX11XXXXXX11XXXXXX11XXXXXX111000
000111111111111111111111111111111111XX000
000XXXXXXXX11XXXXXX11XXXXXX11XXXXXX111000
000111111111111111111111111111111111XX000
000XXXXXXXX11XXXXXX11XXXXXX11XXXXXX111000
000111111111111111111111111111111111XX000
000XXXXXXXX11XXXXXX11XXXXXX11XXXXXX111000
000111111111111111111111111111111111XX000
000XXXXXXXX11XXXXXX11XXXXXX11XXXXXX111000
00011111111111111111111111111111111111000
000XXXX00XXXXXXXX11XXXXXX11XXXXX00XXXX000
00011110011111111111111111111111001111000
000XXXX00XXXXXXXX11XXXXX11XXXXX100XXXX000
00011110011111111111111111111111001111000
000XXXX00XXXXXXXX11XXXXX11XXXXX100XXXX000
11111110011111111111111111111111001111000
XXXXXXX00XXXXXXXX11XXXXX11XXXXX100XXXX000
11111110011111111111111111111111001111000
XXXXXXX00XXXXXXXX11XXXXX11XXXXX100XXXX000
11111110011111111111111111111111001111111
XXXXXXX00XXXXXXXX11XXXXX11XXXXX100XXXXXXX
```

NPS: 450

Runtime: 0.013

GFO: 19.87 %

This is an example of the ILP being stopped before find the optimal value. Note that, after a runtime of 39112 seconds (about 11 h), the GFO is equal to 12.5 % but always 456 people seated. Thus, the GFO of the FirstFit algorithm is 15.95 % and the GFO of the LeastBlockFit is 13.65 % in this case.

RESULTS FOR LARGER INPUTs

For large inputs, the ILP solver is becoming useless since it takes too much time even for compute a first solution, thus, the results are calculated only with both the FirstFit and clever FirstFit based algorithm.

Let’s look at the nearly worst size input: n = 998, m = 993

For that layout, we used the test instance on MS Teams: Exact21.txt (number of seats: 868537)

FirstFit results:

NPS: 330348 Runtime: 807.487 sec (about 14 min) GFO: not available (NO upper bound)

LeastBlockFit results:

NPS: 343494 Runtime: 11443 sec (about 3.2 hours) GFO: not available (NO upper bound)

This last result shows that both the algorithms are able to compute a good solution in a reasonable time even for the worst size input allowed.

OFFLINE CONCLUSIONS

Interesting consideration about the ILP model:

More “ordered” (more 1s)

$\Rightarrow$ combinatorially more possibilities

$\Rightarrow$ ILP slower (but FF/FBF don’t care)

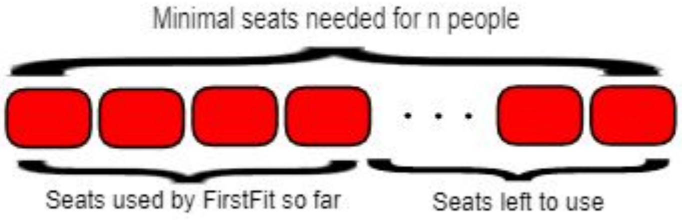
ONLINE CINEMA PROBLEM

Proof competitiveness of FF

Proof - One Row

- **Lemma 1.** *The minimum number of seats needed on a single row for n people $s(n)$, is*

$$s(n) = n + \lfloor \frac{n-1}{8} \rfloor \cdot 2$$



1

10

1111111111

1 8

The optimal solutions would be:

xxxxxxxx00

Proof - One Row

▶ **Lemma 2.** *The FirstFit algorithm uses at most $3 \cdot N$ chairs when placing N people on a single row.*

```
1
10
1111111111
1 8
```

The solution by FirstFit would be:

```
x001111111
```

Proof - One Row

- **Theorem 3.** *The competitive ratio* $\frac{OPT}{ALG}$ *on a single row cannot exceed 8*

$$
\begin{aligned}
\#\text{free chairs} &\geq OPT + \lfloor \frac{OPT - 1}{8} \rfloor \cdot 2 - 3 \cdot ALG \\
&\geq 8 \cdot ALG + 1 + \lfloor \frac{8 \cdot ALG + 1 - 1}{8} \rfloor \cdot 2 - 3 \cdot ALG \\
&= 5 \cdot ALG + 1 + \lfloor \frac{8 \cdot ALG}{8} \rfloor \cdot 2 \\
&= 7 \cdot ALG + 1
\end{aligned}
$$

Thus, CONTRADICTION, so competitive ratio of ALG/OPT=1/8

Proof - Multiple Rows

▶ **Lemma 4.** *The minimum number of seats needed on a single row, with gaps of size α inbetween groups, for n people $s_\alpha(n)$, is*

$$s_\alpha(n) = n + \lfloor \frac{n-1}{8} \rfloor \cdot \alpha$$

Proof - Multiple Rows

▶ **Lemma 5.** *We need $s_9(n)$ or more seats in a cinema with multiple rows to seat n people.*

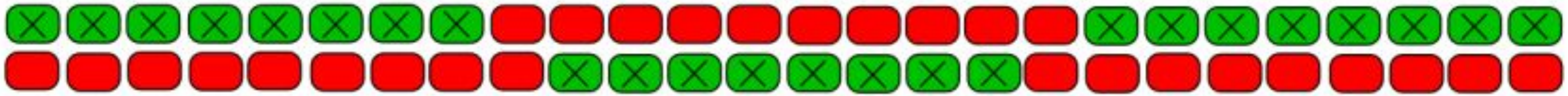
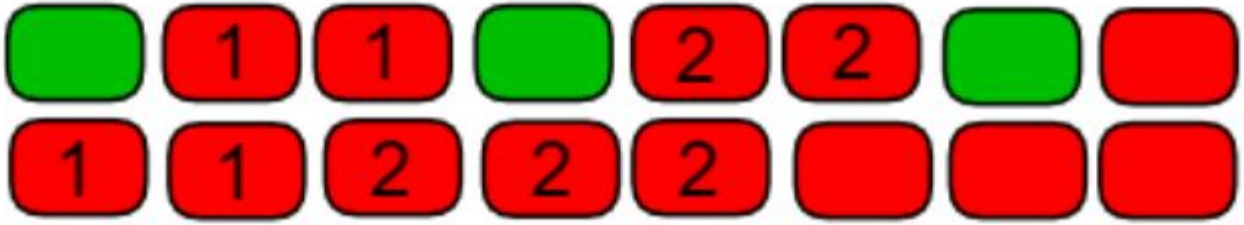


Figure 1 An example setup for $n = 24$ with row width $m = 26$. If n was bigger this pattern is repeated until n is reached

$$s_9(n) + \beta.$$

Proof - Multiple Rows

- **Lemma 6.** *In the worst case the FirstFit algorithm uses less than $6 \cdot N$ seats for N people.*



Proof - Multiple Rows

- **Theorem 7.** *The ratio $\frac{OPT}{ALG}$ on multiple rows cannot exceed 8.*

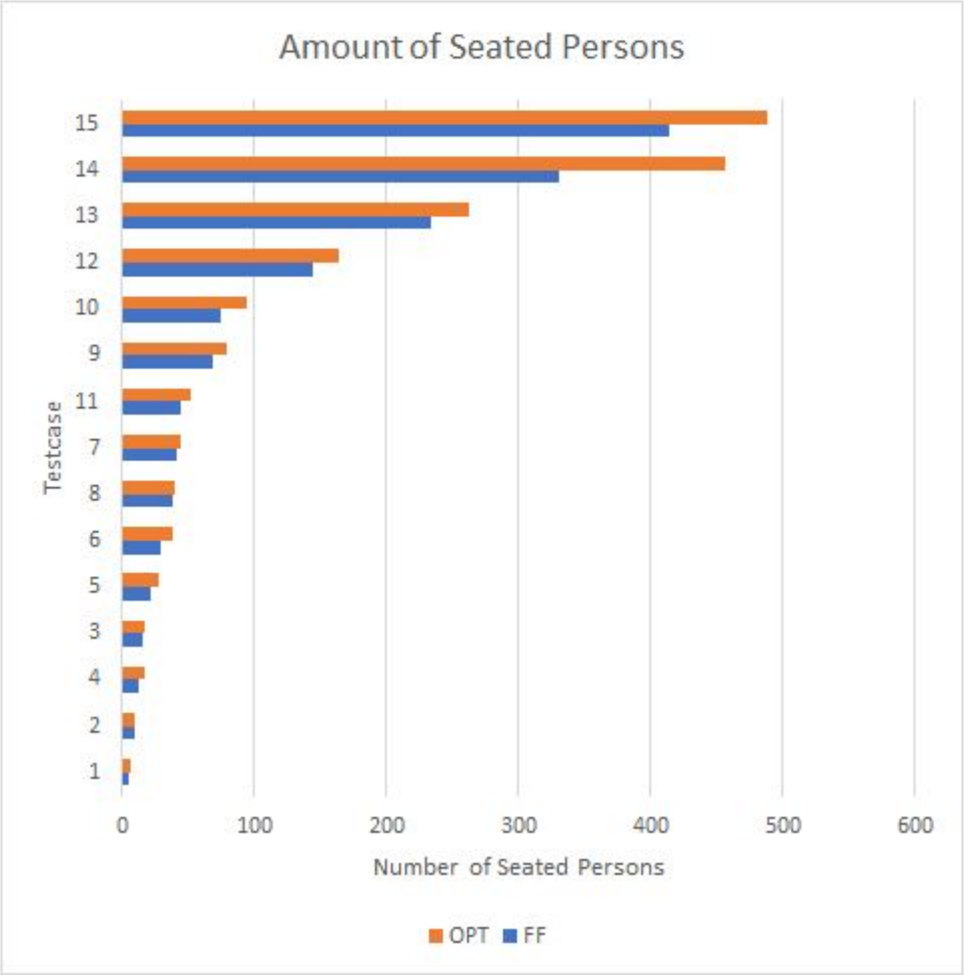
$$
\begin{aligned}
\#\text{free chairs} &\geq s_9(OPT) - 6 \cdot ALG \\
&= OPT + \lfloor \frac{OPT - 1}{8} \rfloor \cdot 9 - 6 \cdot ALG \\
&\geq 8 \cdot ALG + 1 + \lfloor \frac{8 \cdot ALG + 1 - 1}{8} \rfloor \cdot 9 - 6 \cdot ALG \\
&= 2 \cdot ALG + 1 + \lfloor \frac{8 \cdot ALG}{8} \rfloor \cdot 9 \\
&= 11 \cdot ALG + 1
\end{aligned}
$$

Thus, CONTRADICTION, so competitive ratio of ALG/OPT = 8

RESULTS COMPARISON:
OFFLINE vs ONLINE PROBLEM

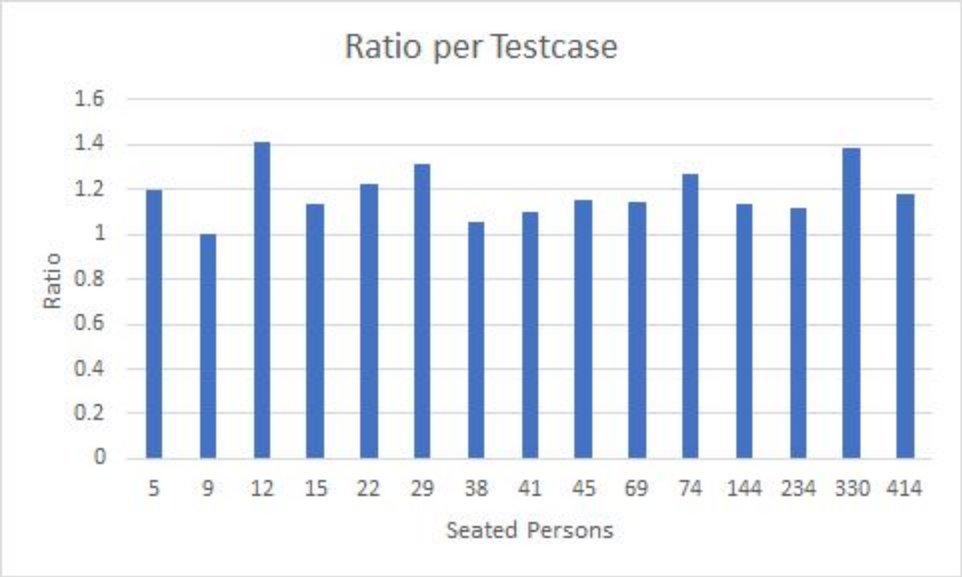
Online VS Offline

- Ratio seems not so bad



Ratio OPT/ALG

- Quite consistent
- Once 1.42



- Only test cases where OPT could be calculated

Runtime FF

- Quite consistent
- Very high OnlineA18