**OPTIMIZATION METHODS AND ALGORITHMS**

**PROBLEM FORMALIZATION**

|  |  |  |  |
| --- | --- | --- | --- |
|  | NOTATION: | Domain: |  |
| 1 |  |  | Student index |
| 2 |  |  | Exam index |
| 3 |  |  | Timeslot index |
| 4 |  |  | Auxiliary timeslot index |
|  |  |  |  |
|  | **DATA:** |  |  |
|  |  | Total number of student enrolled in at least 1 exam | |
|  |  | Total number of exams | |
|  |  | 1 if student *i* is enrolled in exam *j*, 0 o/w | |
|  |  | Number of available timeslots | |
|  |  |  |  |
|  | **VARIABLES:** |  |  |
| 1 |  |  | 1 if exam *j* is scheduled on timeslot *h*, 0 o/w |
| 2 |  |  | 1 if student *i* is enrolled in exam *j*, which takes place on timeslot *h*, 0 o/w |
| 3 |  |  | 1 if student *i* is occupied during timeslot *h*, 0 o/w |
| 4 |  |  | 1 if student *i* is occupied in both timeslots *h* and *k*, 0 o/w |
| 5 |  |  | Number of students occupied in both timeslots *h* and *k* |
|  |  |  |  |
|  |  |  |  |
|  | **CONSTRAINTS:** |  |  |
| 1 |  |  | Each exam will have one and only one time slot |
| 2 |  |  | yijk is 1 if tells if student *i*, is enrolled in exam *j* and exam *j* takes place in timeslot *k* |
| 3 |  |  | Student *i*  cannot be entrolled in more exams which are in the same timeslot, hence the sum of all exams in which student *i* is enrolled, which take place in timeslot *k*, is 1 or 0. |
| 4 |  |  | zih is 1 if student *i* is occupied in timeslot *h*, 0 otherwise. |
| 5 |  |  | uihk is 1 if student *i* is occupied in both timeslots h and k, 0 o/w. |
| 6 |  |  | nhk is the number of students involved in each potential collision. |
|  |  |  |  |
|  | **COST FUNCTION:** |  | |
|  |  |  | |
|  |  |  |  |
|  | **OBJECTIVE FUNCTION** |  |  |
|  |  | | |
|  |  | | For each timeslot h, we analyze the next 5 timeslots (index:k).  For each of couple of timeslots h, k we multiply the number of students occupied in exams in both timeslot and the cost function calculated in k-h. |
|  |  | | Last 5 timeslots:  For each of them (h=tmax-i), we analyze the next timeslots till the end of the timeslots array (k) . For each couple (*tmax*-*i),k* we multiply the number of students occupied in both timeslots and the cost function in (tmax-i)-k. |
|  | **Alternative solution to solve “the last 5 slot problem”**: We append 5 more timeslots, in which it’ll be forbidden to place new exams, thus the objective function can be reduced to: | | |
|  |  | | |
|  | Constraint #1 guarantees therefore that no exam will be placed in these new fictional timeslots. | | |

* The last 5 slot problem solution