

SCHEDULING OF EXAMINATION TERMS BASED ON PAST EXPERIENCE

Damir Kalpic ¹⁾, Zvonimir Vanjak ²⁾, Mirta Baranovic ³⁾

Faculty of Electrical Engineering and Computing
University of Zagreb
Unska 3, 10000 Zagreb, Croatia

1) damir.kalpic@fer.hr <http://www.zpm.fer.hr/kalpic>
2) zvonimir.vanjak@fer.hr <http://www.zpm.fer.hr/zvone>
3) mirta.baranovic@fer.hr <http://www.zpm.fer.hr/mirta>

Abstract: - *A faculty with few thousand students, two studies and few hundred subjects faced the problem of how to devise the examination schedule. The aim of the examination schedule is to maximally disperse the examination terms as perceived by the students. Mixed-integer model turned out to be too complex and therefore genetic algorithm was applied. The achieved results were satisfactory enough to be accepted in the current academic year*

Keywords: - *scheduling, quadratic assignment problem, genetic algorithm, university, examination, data mining*

1. INTRODUCTION

The problem is how to propose a schedule of examination terms at a university institution with enrolled few thousand students, two fields of study, multiple specialisations, few hundred compulsory and elective subjects – depending on study and specialisations, a five years long regular curriculum and three major examination periods – winter, summer and autumn. A year of study is divided into two semesters, winter and summer. The mentioned examination periods fall after the winter semester, after the summer semester and before the start of the new academic year, i.e., before the winter semester. The students are allowed to make three trials at examination in a single subject. If unsuccessful, they face examination performed by a committee of three professors. Those students who fail, have to enrol the same subject again. The whole procedure can be repeated, but if the terminal result is again unsatisfactory, the student has lost further right to study here. As a rule, each examination consists of a written and an oral session. Successful written session is a prerequisite for the oral part of examination. The education is performed by 11 departments.

During the major examination periods, which last about one month each, the institution has to offer to the students two or three non-exclusive examination terms.

Depending on the student's study and/or specialisation, the same subject can be taught in different years of study. On the other hand, the

same department having a limited staff has to engage rather work-intensively to organise and evaluate written examinations.

The whole examination schedule must be known and announced to the students at the beginning of an academic year, i.e., at the beginning of the winter semester. Scheduling of all these examinations turned out to be a difficult problem, not properly solvable by some elementary combinatorial techniques. The solution was attempted by using OR techniques and two approaches were implemented: mixed-integer programming and the genetic algorithm. Acceptable results have been achieved by the genetic algorithm.

2. MATHEMATICAL MODEL

As the objective of the scheduling, the maximum weighted distance among the examination terms was set, and as a weight the number of students who can potentially apply for both of these two terms was used. As the initial number of students for these weightings, the count of regularly enrolled students was taken.

The following input data are required to construct the model:

- CALENDAR
- EXAMINATION PERIOD (examination period code, starting date, terminal date)
- RULE (examination period code, code of the semester in which lecturing of the subject has terminated, number of examination terms)
- PLAN (study, specialisation, subject code, code

of the semester in which lecturing of the subject has terminated)

- EQUIVALENCE (subject code, code of an equivalent subject)
- CONJUNCTION (subject code_1, subject code_2)
- DISJUNCTION (subject code_1, subject code_2)
- ENROLMENT (study, specialisation, subject code, count of the regularly enrolled students last year)

All input data required to construct mathematical model were gathered from the database developed for the administrative needs of the Faculty. Already for ten years all the relevant data regarding education and student administration are recorded in the database [1]. Student behaviour patterns can be extracted as a sort of data mining. Data for scheduling of examinations are partly based on such past experience.

Model was defined so that optimisation can proceed independently for each of the examination periods and for each of them the following sets can be established.

2.1 SETS

Let $J = \{j\}$ be the set of all the terms available for examination within an examination period and let the cardinality of the set J be equal N .

Let us divide the ordered set of all the M subjects $P = \{p\}$ to the subset $P_{ys}^{(w)}$ representing all the subjects that terminate in the winter semester of academic year y of the study/specialisation s , and to $P_{ys}^{(s)}$ respectively for the summer semester. Their cardinalities are M_{ys}^w and M_{ys}^s . The intersections among these subsets need not be empty, because identical subject p can exist on different studies/specialisations and in different semesters.

Set $U = \{u_p\}$ contains the count of students enrolled in subject p ; $p = 1 \dots M$. The set that contains the count of regularly enrolled students simultaneously in subjects p and p' is $V = \{v_{pp'}\}$. For $p = p'$ it is obviously $v_{pp'} = u_p$. If the zero elements are omitted, the cardinality of V is $W J M^2$.

Let us expand the input rules, to form a set containing within every examination period for each subject p the number of available terms $R = \{r_p\}$.

Also, we needed one more parameter in our model, and that is L , which will represent the prescribed count of examinations for each subject. Usually in the winter examination period $L = 2$, and in summer and autumn $L = 3$.

Set of pairs $K = \{k, k'\}$ derives from the relation CONJUNCTION, representing the subjects

that preferably should have common terms for their respective written examinations.

The result are ordered sets of examination terms within an examination period for every subject p , denoted by $Q_p = \{q_1, q_2, \dots, q_{r(p)}\}$.

In further text, a reference to indices implies belonging to the respective sets.

3. MIXED-INTEGER MODEL

The first attempt to solve the problem was by formulating it as a mixed-integer programming problem, which should be solvable by standard mathematical programming software.

3.1 DECISION VARIABLES

The result of optimisation are the values of the binary decision variables D_{jp} showing whether on the day j ($j=1, N$) of an examination period, an examination term for subject p exist or not, whereby $D_{jp} = 0$ denotes "no" and 1 means "yes".

3.2 EXPLICIT CONSTRAINTS

For each subject p the count of examination terms must be respected:

$$\sum_j D_{jp} = r_p, \quad \forall p \quad (1)$$

Requirement for common terms is:

$$D_{jp} = D_{jp'}, \quad \forall p, p' \in K \quad (2)$$

3.3 OBJECTIVE FUNCTION

The weighted distance among all the examination terms for subjects p' and p'' is:

$$z_{pp'} = v_{pp'} * \sum_{j'=1}^N \sum_{j=j'}^N (j' - j)^k D_{jp} D_{j'p'} \quad (3)$$

It is obvious that $k = 1$ would make all the feasible solutions alternatively optimal. One can be misled by trying to motivate for greater mutual distance by putting $k > 1$. However, $k > 1$ and $r_p > 2$ would lead to unacceptable grouping of terms at the ends of the examination period. To motivate for proper spread of terms, $k < 1$ must be applied. The value $k = 0.5$ was chosen in implementation.

The formulation in (1) to (3) belongs to the class of quadratic assignment problems [2]. Trying to adapt the model to be solvable by generally available mathematical programming tools [3], the model was expanded in size, but reformulated as a linear binary programming problem. The binary variables $D_{j|p|p'}$ were introduced to notify whether a pair of terms simultaneously exist or do not exist; whether on a day j an examination term exist for the sub-

ject p and simultaneously on the day j' falls the examination term for subject p' . The objective function is finally formulated as:

$$\max z = \sum_p \sum_{p' \geq p} v_{pp'} * \sum_{j'=1}^N \sum_{j=j'}^N (j' - j)^{0.5} D_{jpp'} \quad (4)$$

3.4 PROBLEM COMPLEXITY

For an examination period of N working days, with W nonempty shared subjects, it yields $N * W * W/2$ binary variables. As a binary programming model has the a priori complexity of $O(2^n)$, its size appears prohibitive.

To reduce the problem size, the group-variables property should be applied:

$$\sum_j \sum_{j' \geq j} \sum_{p' \geq p} D_{jpp'} = r_p, \quad \forall p \quad (5)$$

$$\sum_j \sum_{j' \geq j'} \sum_{p' \geq p} D_{jpp'} = r_{p'}, \quad \forall p \quad (6)$$

Instead of 2^N , this reduces the number of combinations to $(N \text{ over } r_p)^2$. Some examinations must be simultaneous:

$$D_{jpp'} = 0, \text{ for } j \neq j', = p, p' \in K \quad (7)$$

Such variables are omitted from the model.

For a modest estimation, let us take the summer examination period consisting of about 40 days and let us suppose that there exist only 100 values v_{pp} , $\forall 0$. There are 3 possible terms for each subject. This would produce:

$$100 * 40 * 20 = 80000 \text{ binary variables.}$$

Due to $80000 * 79999 * 79998 / (1 * 2 * 3)$, instead of $2^{80000} \gg 10^{24082}$, the upper bound for iterations becomes approximately $8.5 * 10^{13}$. The reduction is dramatic, but it would not help much to make the model applicable.

There was a hope that the linear programming solution in the domain of real numbers could give a good starting point for integer search. However, the disappointment was complete. The solution in the real numbers domain could not be further from the binary one. The solution for all the decision variables regarding a subject p was uniform and equal to N / r_p .

Therefore, the attempts of using mixed integer programming have been abandoned and genetic algorithm was used instead.

4. GENETIC ALGORITHM APPROACH

In any attempt to apply the genetic algorithm, one is confronted with two tasks [4]. The first one

is to formulate the algorithm, what includes the choice of a strategy to build the initial population and to define crossover and mutation processes. The second task, formulation of the objective function is more sensitive and it will strongly influence the quality of solution.

The first task is to define the form of the chromosomes. Every chromosome defines the properties of an individual and it must contain an entire solution. In our case, it must contain a list of all the examination terms for all the subjects in the problem space. Accordingly, a chromosome contains the set

$$T = \{ t_{pl} \},$$

where p denotes the subject, and l the examination sub period; $p = 1 \dots M$; $l = 1 \dots L$.

To reduce the degrees of freedom and to improve the convergence, L sub periods were defined in each examination period.

The next step was the decision how to represent this set. The first choice was a bit map approach. Every chromosome contains a set of M integers for each subject, where each of these integers consists of N bits for each examination term. The bits would denote whether the examination takes place ($=1$), or not ($=0$). The expectation was that the genetic algorithm operations such as crossover and mutation would be easily performed by logical operations over the bits.

However, this approach showed significant setbacks in the phase of presentation and manipulation of the optimisation results. Instead, L integers were introduced bearing the value of the date for each examination term. In the final form every chromosome contains a matrix:

$$T_{pl}, \quad p = 1, \dots, M, \quad l = 1, \dots, L$$

where $T_{pl} \in \mathbf{O} \mathbf{J}$ represents the examination term for subject p within sub period l .

After determining chromosome representation, we turn our attention to genetic algorithm implementation. For solving this problem, 3-tournament elimination selection [5] genetic algorithm was chosen, and it can be formulated as follows:

```

generate an initial population P1;
repeat
  generate 3 distinct random numbers; a_k, k=1,2,3
  evaluate the objective function for individuals P_ak
  detect the worst individual
  crossover the remaining two individuals
  replace the worst individual
  mutate the population
until iteration count < ITER_NUM
    
```

To start genetic algorithm, an initial population of size POP_SIZE has to be created. A random generation was applied. For every individual all the subjects were activated and for each of them L distinct numbers representing the examination terms were generated.

An attempt was made to prepare “advanced” solutions, by respecting the division into sub periods and by avoiding too short time distance between examination terms for the same subject. However, it turned out that random solutions provide for better solutions and faster convergence.

For every crossover, the objective function for three randomly selected individuals is evaluated. An individual resulting from crossover of the remaining two replaces the worst individual of three. The crossover of two chromosomes results in a new one having their common examination terms. The unmatched terms are filled by copying them from randomly selected one of the two chromosomes.

For the mutation, a random number is generated from the interval $[1, MOD_MUT]$. MOD_MUT is a parameter for genetic algorithm ranging from 5000 to 25000. When it occurs that the generated number falls within the interval $[1, L]$, then the mutation of a chromosome corresponding to that number is performed.

```

for each individual k
  for each subject p ∈ k
    if rand() % MOD_MODUL < L
      repeat
        term=rand()%card(R)
      until(term contains L terms for subject p)
    
```

5. OBJECTIVE FUNCTION

To formulate the objective function it was necessary to define the requirements imposed on the solution and to define the priorities, because some requirements were mutually in contradiction. The requirements were:

0) The overall time span summed among all the examination terms, weighted by the number of students who are candidates for each pair of examinations should be at maximum.

1) The time span between two examination terms for the same subject must be at least MIN_DIST days.

2) For every subject, single examination terms should fall within sub periods.

3) Satisfaction of examiners’ preferences for grouping of subjects.

4) Feasibility regarding the limited examiners’ capacity within a day.

It is obvious that some of the requirements are contradictory and therefore the importance of each of these requirements was weighted by a corresponding parameter. Of course, the results of optimisation significantly depended upon these parameter relative values. Human arbitration based on experience was the final judge regarding the quality and applicability of the achieved solutions.

The objective function consists of z_0 , the overall sum of time spans between all the pairs of examination terms weighted by the number of students who could be subjects to respective examinations. The function is diminished by 4 penalty members, $z_1 \dots z_4$ representing the satisfaction of respective requirements. The value of the objective function is a weighted sum of $z_0 \dots z_4$.

5.1 CONTRIBUTION OF DISTANCES AMONG EXAMINATION TERMS FOR DIFFERENT SUBJECTS

```

for each subject p
  for each subject p' > p
    for each sub period l
      dist = |tp1 - tp'1|
      z0 := z0 + sqrt (vpp' · dist);
    
```

In practical implementation the set V is represented by a sparse matrix. For every subject a list of only those subjects having non-empty intersection is formed.

5.2 CONTRIBUTION AND PENALTY DUE TO DISTANCES BETWEEN SAME SUBJECTS

To z_1 the penalty for violation of minimum distance MIN_DIST between the adjacent examination terms for the same subject is added. If no violation occurs, the z_0 term is increased instead.

```

for each subject i
  for each pair of examination terms
    dist = time span between the terms
    in pair
    if dist < MinDist
      z1 := z1 + MinDist - dist
    else
      z0 := z0 + sqrt (ui · dist);
    
```

5.3 PENALTY FOR NON OBEDIENCE OF DISTRIBUTION INTO SUB PERIODS

If an examination term does not fall within the prescribed sub period, the penalty z_2 is incremented:

```

for each subject p
  for each sub period l
    if tp1 ∉ interval for l
      z2 := z2 + 1
    
```

5.4 PENALTY FOR NON OBEDIENCE OF CONJUNCTION REQUIREMENTS

The requirement satisfaction is tested against the set of subjects with preferably common examination terms $K = \{k, k'\}$:

```

for every pair k, k' ∈ K
  for each sub period l
    if tk1 ≠ tk'1
      z3 := z3 + 1
    
```

5.5 PENALTY FOR WORK OVERLOAD

As the faculty chairs can simultaneously examine a limited number of students, the penalty for a

possible work overload MAX_LOAD is introduced:

```

for each subject p
  detect its belonging chair g
  for each sub period l
    |  $C_{tp1, g} := C_{tp1, g} + up$ 
for each examination term r
  for each chair g
    | if  $C_{r, g} > MAX\_LOAD$ 
    |  $z_4 := z_4 + MAX\_LODAD - C_{tp1, g}$ 
    
```

The resulting objective function is formulated as:

$$z = \sum_{i=0}^4 W_i \cdot z_i \quad (8)$$

For a better comparison among the solutions achieved with different weightings W_i ; $i = 0 \dots 4$, following values have been singled out:

- the count of pairs of examination terms for the same subject, which are violating the minimum distance requirement,
- the count of examination terms not properly split into sub periods,
 - the count of violated conjunction requirements,
 - the work overload

Ideally, all the above values should be zero. However, it is a difficult quadratic assignment problem where for example for the autumn examination period 215 subjects, with 3 examination terms each, had to be placed within 18 workdays. The requirement violations were allowed. Final result can be achieved by manual editing about the declared optimum solution, because a mathematical model can rarely contain all the real-life details.

5.6 Optimisation results

The achieved results have shown advantage over the manual scheduling or over applying some simple combinatorial techniques. However, some unexpected practical problems arose due to the ever-changing syllabus. Examinations for some obsolete subjects, which are still relevant for some students had to be added manually.

6. CONCLUSION

We can conclude that the application of genetic algorithm has shown to be an appropriate tool for scheduling of examination terms. Some subjective evaluation and slight editing of the results achieved by the algorithm was necessary. The next step should be a refinement of the objective function evaluation, to increase the subjective level of satisfaction and to diminish the need to edit the achieved results manually.

The quality of the solution depended upon the problem complexity. The solutions for the first two years of study were near to excellent. Also a sub-

jective evaluation agreed that the terms were properly split. On the other hand, the scheduling for final years of study with 150–160 subjects was relatively time consuming. To calculate the autumn examination period, it took about 5–6 hours on a Pentium with 160 MB RAM and a 400 MHz processor. Nevertheless, the distribution of examination terms appeared fair, although it was very difficult to evaluate it manually.

The expected further investigation should proceed in two directions. The first direction shall be to improve the evaluation and analysis of the achieved results. It is very difficult to tell how good a solution is, so comparison with solutions obtained by some other method could be helpful. Some of these method are simulated annealing [6] or tabu search [7], which, according to some investigations are somewhat superior to solving by genetic algorithm, although Ahuja et al. [8] recently reported about genetic algorithm, called by them greedy genetic algorithm, that performs equally well on the quadratic assignment problem. The second direction is an improvement of the applied genetic algorithm. Parallel processing could increase the speed. Further improvement in formulation of the objective function could contribute to higher quality of results.

7. REFERENCES

- [1] Kalpic D, Mornar V: *Student Administration System, European review conference proceedings "University-Enterprise Information Systems"*, Graz, September 15-16, 1994, pp 124-131
- [2] Koopmans TC and Beckmann MJ, *Assignment problems and the location of economic activities. Econometrica* 25 (1957). pp. 53-76.
- [3] *SAS/OR User's Guide, SAS Institute Inc., Cary, NC, USA, 1989.*
- [4] Michalewicz Z, *Genetic Algorithms + Data Structures = Evolution Programs, 2ed, Springer-Verlag, New York, 1994.*
- [5] Bòck T, *Generalized Convergence Models for Tournament- and (m,l)-Selection, Proceedings of the Sixth International Conference on Genetic Algorithms, San Francisco, CA, 1995, pp. 2-8.*
- [6] Wilhelm MR and Ward TL, *Solving quadratic assignment problems by "simulated annealing"*, *IEEE Transactions* 1 (1987). pp. 107-119.
- [7] Skorin-Kapov J, *Taboo search applied to the quadratic assignment problem, ORSA Journal on Computing* 2 (1990). pp. 33-45.

[8] Ahuja RK, Orlin JB, Tivari A, *A greedy genetic algorithm for the quadratic assignment problem, Computers & Operations Research* 27 (2000). pp. 917-934.



Damir Kalpic is professor in the field of computing at the Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia. He achieved his B.Sc., M.Sc. and Ph.D. at the same institution. His professional interest and activity is the application of computers in different fields.

Primarily, it is the development of information systems, supported by database systems and extended with mathematical models stemming from operational research. From this field of activity, a scientific, professional and educational group, in which he is the senior member, offers consulting, education and software development for business, industry, administration and other institutions. He lectures Algorithms and Data Structures, Operational Research and Information Systems. He mentored more than 150 graduation theses, 25 master's theses and 6 doctorates. Currently he acts as vice dean. He is author or co-author of about 30 internationally published articles. He has been member of numerous international conference programme committees and has led more than 15 projects. Besides his mother tongue Croatian, he can communicate, in descending order of fluency, in English, German, Italian, Spanish, French and Portuguese.