

# A Quantitative Approach for the Design of Academic Curricula

Carlos Castro<sup>1</sup>, Broderick Crawford<sup>1,2</sup>, and Eric Monfroy<sup>1,3</sup>

<sup>1</sup> Universidad Técnica Federico Santa María, Valparaíso, Chile

<sup>2</sup> Pontificia Universidad Católica de Valparaíso, Chile

<sup>3</sup> LINA, Université de Nantes, France

Carlos.Castro@inf.utfsm.cl,

Broderick.Crawford@ucv.cl,

Eric.Monfroy@univ-nantes.fr

**Abstract.** We present experimental results of solving mathematical models, using very efficient complete and incomplete techniques, for designing balanced academic curricula. Solutions to these models give curricula with academic load for each period as similar as possible. Based on this work, a software tool has been developed to help administratives as well as students in the planning of academic curricula.

## 1 Introduction

When designing an academic curriculum a lot of factors are taken into account. Based on the objectives defined for the career under consideration, experts have to propose the courses covering the fundamentals in each domain. Roughly speaking, each domain is represented by a set of courses and some precedence relationships are also imposed among them. Indeed, the academic load of each course, representing the amount of effort required to successfully follow the course, is expressed in some way (generally called credit). Some explicit restrictions can also be imposed when developing a curriculum. For example, number of academic periods, maximum/minimum number of courses per period, and maximum/minimum academic load per period. Once this information is available, the assignment of courses to each academic period is carried out using a trail and error approach until an adequate curriculum is reached.

Assuming that a balanced academic load favors academic habits and facilitates the success of students, we are interested in designing balanced academic curricula. Considering that the academic load of each period is given by the sum of the credits of each course taken in a period, the problem of designing balanced academic curricula consists in assigning courses to periods in such a way that the academic load of each period will be balanced, i.e., as similar as possible. In order to obtain such a balanced curriculum we think that a quantitative approach, instead of the traditional trail and error approach, can be very useful.

Although the mathematical models representing these situations are hard to solve (in fact, they are Combinatorial Optimization problems [6]), the development of powerful complete as well as incomplete methods in the last years allows to tackle real-life instances. Complete techniques aim to find the global optimal solution, however, when problems become too hard, these techniques cannot find the optimal solution in a reasonable time. Hopefully, incomplete techniques have been used to deal with these hard problems obtaining, in general, good local optimal solutions very quickly. Currently, a lot of research is done on the development of hybrid methods integrating complete and incomplete techniques aiming to solve hard problems very efficiently.

This paper is organized as follows: section 2 describes the problem we are interested in. Section 3 presents an integer programming model for the balanced academic curriculum problem. In order to better explain this model, we present, in section 4, a very simple example. In section 5, we present the solution to this example, we present experimental results when solving three real cases, and we describe some applications of the model. Finally, in section 6, we conclude the paper and give some perspectives for further works.

## 2 The Balanced Academic Curriculum Problem

In [2], we introduce the Balanced Academic Curriculum Problem (BACP), we propose a mathematical model, and we solve some real-life instances. The problem was then studied by others authors to evaluate the performances of some solving techniques using different models [3]. BACP was included as a new benchmark (prob030) in the library CSPLIB, available at [www.csplib.org](http://www.csplib.org), a set of Constraint Satisfaction Problems [4]. As a general framework we consider administrative as well as academic regulations of the Technical University Federico Santa María:

- Academic Curriculum: An academic curriculum is defined by a set of courses and a set of precedence relationships among them.
- Number of periods: Courses must be assigned within a maximum number of academic periods.
- Academic load: Each course has associated a number of credits or units that represent the academic effort required to successfully follow it.
- Prerequisites: Some courses can have other courses as prerequisites.
- Minimum academic load: A minimum amount of academic credits per period is required to consider a student as full time.
- Maximum academic load: A maximum amount of academic credits per period is allowed in order to avoid overload.
- Minimum number of courses: A minimum number of courses per period is required to consider a student as full time.
- Maximum number of courses: A maximum number of courses per period is allowed in order to avoid overload.

In this work, we concentrate on three particular instances of the balanced academic curriculum problem: the three Informatics careers offered by the Technical University Federico Santa María.

### 3 Integer Programming Model

In this section, we present an integer programming model for the balanced academic curriculum problem. Classical examples and very good explanations on the subject of modeling in integer programming can be found in [8].

- Parameters

Let

- $m$ : Number of courses
- $n$ : Number of academic periods
- $\alpha_i$ : Number of credits of course  $i$ ; for all  $i = 1, \dots, m$
- $\beta$ : Minimum academic load allowed per period
- $\gamma$ : Maximum academic load allowed per period
- $\delta$ : Minimum amount of courses per period
- $\varepsilon$ : Maximum amount of courses per period

- Decision variables

Let

- $x_{ij} = 1$  if course  $i$  is assigned to period  $j$ ; 0 otherwise for all  $i = 1, \dots, m$ ; for all  $j = 1, \dots, n$ .
- $c_j$ : academic load of period  $j$ ; for all  $j = 1, \dots, n$ .
- $c$ : maximum academic load for all periods

- Objective function

$$\text{Min } c = \text{Max } \{c_1, \dots, c_n\}$$

- Constraints

- The academic load of period  $j$  is defined by:

$$c_j = \sum_{i=1}^m \alpha_i \times x_{ij} ; \forall j = 1, \dots, n$$

- All courses  $i$  must be assigned to some period  $j$ :

$$\sum_{j=1}^n x_{ij} = 1; \forall i = 1, \dots, m$$

- Course b has course a as prerequisite:

$$x_{bj} \leq \sum_{r=1}^{j-1} x_{ar}; \forall j = 2, \dots, n$$

- The maximum academic load is defined by:

$$c = \text{Max} \{c_1, \dots, c_n\}$$

This can be represented by the following set of linear constraints:

$$c_j \leq c; \forall j = 1, \dots, n$$

- The academic load of period j must be greater than or equal to the minimum required:

$$c_j \geq \beta; \forall j = 1, \dots, n$$

- The academic load of period j must be less than or equal to the maximum allowed:

$$c_j \leq \gamma; \forall j = 1, \dots, n$$

- The number of courses of period j must be greater than or equal to the minimum allowed:

$$\sum_{i=1}^m x_{ij} \geq \delta; \forall j = 1, \dots, n$$

- The number of courses of period j must be less than or equal to the maximum allowed:

$$\sum_{i=1}^m x_{ij} \leq \varepsilon; \forall j = 1, \dots, n$$

## 4 Example

In order to better explain the general model developed in section 3, we present a reduced version taken from an informatics career offered by the Technical University Federico Santa María. Table 1 presents 18 courses assigned to 4 academic periods, each course identified by its code, the number of credits and the precedence relationships.

As parameters, we consider a maximum of 16 credits per period, a minimum of 3 credits per period, a maximum of 6 courses per period, and a minimum of 1 course per period.

Of course, constraints concerning minimum and maximum number of courses and credits per period are satisfied by the curriculum under consideration, so this assignment represents a possible (but maybe not optimal) solution to the model.

Using this solution students have an academic load of 13, 16, 16, and 10 credits per period. The maximum load per period is 16 credits and the maximum difference between loads per period is 6. We guess that this unbalance load does not help students to successfully follow the courses.

**Table 1.** A reduced academic curriculum

Academic period	Code	Credits	Req1	Req2
01	DEW100	1		
01	FIS100	3		
01	HCW310	1		
01	MAT190	4		
01	MAT192	4		
	Total	13		
02	FIS101	5	FIS100	MAT192
02	IWI131	3		
02	MAT191	4	MAT190	
02	MAT193	4	MAT190	MAT192
	Total	16		
03	FIS102	5	FIS101	MAT193
03	HW1	1		
03	IEI134	3	IWI131	
03	IEI141	3	IWI131	
03	MAT194	4	MAT191	MAT193
	Total	16		
04	DEW 0	2	DEW101	
04	HCW311	2	HCW310	
04	IEI132	3	IEI134	
04	IEI133	3	IEI134	
	Total	10		
	Total	55		

This solution can be represented by the values for the decision variables as presented in Table 2.

**Table 2.** Current assignment to variables  $x_{ij}$  for the reduced example

i/j	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

We now explain how constraints are satisfied by this solution.

- All courses  $i$  must be assigned to some period  $j$ :

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i = 1, \dots, m$$

This is verified by the sum of the elements in each column of the assignment matrix presented in Table 3.

**Table 3.** Current solution assigning each course to some period

$i/j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
Sum	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

- Number of courses of period  $j$  greater than or equal to the minimum allowed:

$$\sum_{i=1}^m x_{ij} \geq \delta \quad j = 1, \dots, n$$

This is verified by the sum of the elements in each row of the assignment matrix presented in Table 4.

**Table 4.** Current solution assigning a feasible number of courses per period

$i/j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	$\Sigma$	$\delta$	$\epsilon$
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4	6	1
2	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	4	6	1
3	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	6	6	1
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	4	6	1

- The academic load of period  $j$  must be greater than or equal to the minimum required:

$$c_j \geq \beta \quad \forall j = 1, \dots, 4$$

- The total number of credits per period is obtained by the multiplication of the assignment matrix and the vector containing the number of credits of each course:

$$\begin{pmatrix} 111110000000000000 \\ 000001111000000000 \\ 00000000011110000 \\ 0000000000000001111 \end{pmatrix} \begin{pmatrix} 1 \\ 3 \\ 1 \\ 4 \\ 4 \\ 5 \\ 3 \\ 4 \\ 4 \\ 5 \\ 1 \\ 3 \\ 3 \\ 4 \\ 2 \\ 2 \\ 3 \\ 3 \end{pmatrix} = \begin{pmatrix} 13 \\ 16 \\ 16 \\ 10 \end{pmatrix}$$

These sums also verify the minimum academic load per period allowed  $\beta = 3$ .

- The maximum academic load is defined by:

$$c = \text{Max} \{c_1, c_2, c_3, c_4\}$$

This can be represented by the following linear constraints:

$$c_j \leq c \quad \forall j=1, \dots, 4$$

Table 5 presents academic loads per period and the value  $c$  satisfying this constraint.

**Table 5.** Academic load per period ( $c_i$ )

I	$c_i$	$c = \text{Max } (c_1, c_2, c_3, c_4)$
1	13	16
2	16	16
3	16	16
4	10	16

– Course b has course a as prerequisite:

$$x_{bj} \leq \sum_{r=1}^{j-1} x_{ar} \quad \forall j = 2, \dots, n$$

For example, considering the precedence between MAT191 and MAT190:

$$\begin{aligned} x_{82} &< x_{41} \\ x_{83} &< x_{41} + x_{42} \\ x_{83} &< x_{41} + x_{42} + x_{43} \end{aligned}$$

## 5 Experimental Results and Applications

Solving the reduced version of a balanced academic curriculum problem, presented in section 4, we obtain the optimal solution presented in table 6. This assignment of courses to periods gives now 14, 14, 14, and 13 credits per period. The maximum load per period is 14 credits and the maximum difference between loads per period is 1. That is what we are looking for, the most balanced academic curriculum.

As real-life instances we first considered the three informatics careers at the UTFSM, involving 8, 10, and 12 academic periods, and 36, 42, and 62 courses, respectively. This means that the mathematical model representing each situation involves,  $36 \times 8$ ,  $42 \times 10$ , and  $62 \times 12$  binary variables, respectively. We have solved all these instances of BACP using complete as well as incomplete techniques. We have used `lp_solve` to apply Operations Research techniques and `Oz` to apply Constraint Programming techniques. `lp_solve` is a software toll for solving integer and linear programming models that is available free of charge at [tech.groups.yahoo.com/group/lp\\_solve](http://tech.groups.yahoo.com/group/lp_solve). Details about Operations Research techniques for solving integer programming models can be found in [6]. `Oz` [7] is a software implementing Constraint Programming techniques that is available for free at [www.mozart-oz.org](http://www.mozart-oz.org). Details about Constraint Programming techniques can be found in [1]. Although both complete techniques used have found the optimal solution for all instances very efficiently, we have also implemented a Genetic Algorithm [5] to apply the incomplete approach. We have obtained very good solutions in a very short time, but still not the optimal solutions obtained by the complete techniques.

Considering the successful results obtained dealing with these instances, we have tried to validate the real applicability of this approach. On one hand, BACP has already been used by the UTFSM to improve several academic curricula. On the other hand, a user-friendly interface has been developed to help students to plan their



curricula when they have not approved some courses. Indeed, we have extended the basic model presented in this paper to consider several careers at the same time in order to assign common courses to the same academic period. Due to the higher complexity of this problem, we had to develop hybrid algorithms integrating complete and incomplete techniques because none of the previous ones was able to solve some problems alone. This more complex problem has been very interesting in terms of economic resources for the UTFSM because solutions allow to reduce the number of common courses dictated, assigning some of them in the same academic period, and so reducing the number of lecturers.

**Table 6.** Optimal solution for the reduced academic curriculum

Academic period	Code	Credits	Req1	Req2
01	FIS100	3		
01	MAT190	4		
01	MAT192	4		
01	IWI131	3		
	Total	14		
02	DEW100	1		
02	FIS101	5	FIS100	MAT192
02	MAT191	4	MAT190	
02	MAT193	4	MAT190	MAT192
	Total	14		
03	IEI134	3	IWI131	
03	IEI141	3	IWI131	
03	MAT194	4	MAT191	MAT193
03	DEW 0	2	DEW101	
03	HW1	1		
03	HCW310	1		
	Total	14		
04	FIS102	5	FIS101	MAT193
04	IEI132	3	IEI134	
04	IEI133	3	IEI134	
04	HCW311	2	HCW310	
	Total	13		
	Total	55		

## 6 Conclusions

We have presented a quantitative approach for designing balanced academic curricula. Using solving techniques implemented in software available for free we have been able to solve very efficiently all instances we have considered in this work. We strongly believe that this kind of quantitative approach should be used in order to support several decision problems that arise in the educational field. Many fruitfull experiences have already been carried out for solving, for example, timetabling problems using a similar approach.

**Acknowledgments.** The first author has been partially supported by the Chilean National Science Fund through the project FONDECYT 1070268. The third author has been partially supported by the Chilean National Science Fund through the project FONDECYT 1060373. All authors have also been partially supported by the project INRIA-CONICYT VANANAA II.

## References

1. Apt, K.R.: Principles of Constraint Programming. Cambridge Univ. Press, Cambridge (2003)
2. Castro, C., Manzano, S.: Variable and Value Ordering When Solving Balanced Academic Curriculum Problems. In: Proceedings of The Sixth Workshop of the Working Group on Constraints, ERCIM 2001, Prague, Czech Republic (June 2001)
3. Hnich, B., Kiziltan, Z., Miguel, I., Walsh, T.: Hybrid Modelling for Robust Solving. *Annals of Operations Research* 130(1-4), 19–39 (2004)
4. Mackworth, A.K.: Constraint Satisfaction. In: Shapiro, S.C. (ed.) *Encyclopedia of Artificial Intelligence*, 2nd edn. vol. 1, Addison-Wesley, New York (1992)
5. Michalewicz, Z., Fogel, D.: *How to Solve It: Modern Heuristics*. Springer, Heidelberg (1999)
6. Nemhauser, G.L., Wolsey, L.A.: *Integer and Combinatorial Optimization*. In: Wiley-Interscience Series in Discrete Mathematics, Wiley, New York (1999)
7. Smolka, G.: The Oz Programming Model. In: van Leeuwen, J. (ed.) *Computer Science Today*. LNCS, vol. 1000, pp. 324–343. Springer, Heidelberg (1995)
8. Williams, H.: *Model Building in Mathematical Programming*, 4th edn. J. Wiley and Sons, New York (1999)