

Article

A Game for Learning How to Model in Graph Theory

Alicia Cordero ^{1,*}, Cristina Jordan ¹, Marina Murillo-Arcila ² and Esther Sanabria-Codesal ³

¹ Institute for Multidisciplinary Mathematics, Universitat Politècnica de València, 46022 València, Spain; cjordan@mat.upv.es

² Instituto Universitario de Matemática Pura y Aplicada, Universitat Politècnica de València, 46022 València, Spain; mamuar1@upv.es

³ Departamento de Matemática Aplicada, Universitat Politècnica de València, 46022 València, Spain; esanabri@mat.upv.es

* Correspondence: acordero@mat.upv.es

Abstract: In this article, we show how to introduce students to modeling while exposing the power of graph theory as a modeling tool. For that purpose, we propose a problem aimed at university students based on a game where the objective is to strengthen the learning of reachability and the shortest path algorithms.

Keywords: modeling; weighted graphs; shortest path algorithms; Dijkstra; Bellman–Ford

MSC: 97M10; 97M50; 05C12; 05C22; 97C70



Citation: Cordero, A.; Jordán C.; Murillo-Arcila M.; Sanabria-Codesal E. A Game for Learning How to Model in Graph Theory. *Mathematics* **2022**, *10*, 1969. <https://doi.org/10.3390/math10121969>

Academic Editors: Juan Francisco Sánchez-Pérez, Gonzalo García Ros and Manuel Conesa

Received: 2 May 2022

Accepted: 4 June 2022

Published: 7 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The teaching of graph theory, the great applicability of which stands out, can be approached in very different ways. On too many occasions, following a “traditional” methodology in the classroom encourages students to adopt a passive role, which leads them to being mere recipients of content with no apparent link to reality or critical spirit.

To make this situation more dynamic and increase their interest and motivation for mathematics, in this article we propose mathematical modeling as an effective learning strategy for the subject of graph theory, as shown in [1–3]. Authors such as Braicovich and Cognigni [4] consider the use of graph theory as a conceptual tool in the construction of models; problem solving enables the acquisition and development of skills such as intuition, exploration, discovery, and the design of hypotheses, which contribute to the development of logical thinking, spatial vision, and abstract reasoning in students. Despite this, we have observed that graph theory is sometimes taught without sufficient attention to modeling.

Mathematical modeling in teaching does not have an agreed definition, mainly due to the different ways of approaching this methodology in the classroom. From a real perspective, modeling concentrates mainly on the relationship between the model and the context in which it arises, focusing its interest on incorporating all of the restrictions into the model. A more cognitive perspective on this methodology focuses on the mental and psychological processes that take place during the creation of the model; finally, the more didactic perspective uses these problems to structure and improve the learning process of the students, thus generating interest in introducing new concepts or deepening those already known.

Considering these different approaches, in this paper we adopt a definition of mathematical modeling that we find useful in working with our students, considering it as the description of a real problem that allows the use of mathematical tools for its analysis and total or partial resolution [5]. This description allows us to analyze the process through the modeling scheme (see Figure 1) provided by Blum and Leiss [6] that distinguishes seven phases:

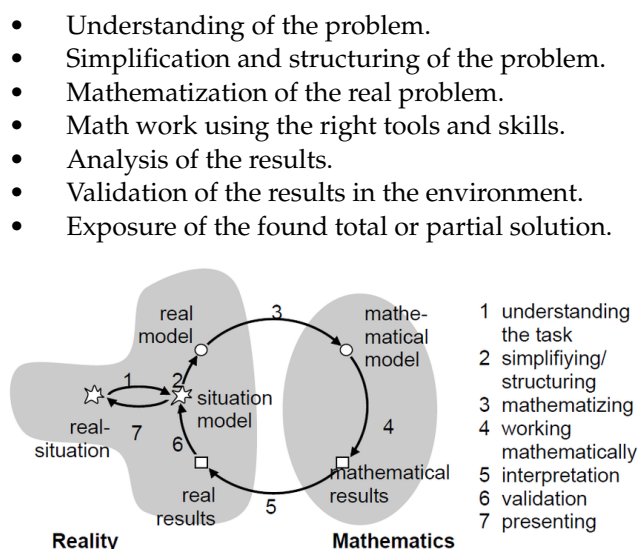


Figure 1. Modeling cycle.

Although these phases are cyclical, the complexity of working with real problems means that there are advances and setbacks in these phases [7–10], as the translation between natural language and the mathematization of the problem sometimes presents great difficulty.

When using mathematical modeling in the classroom, we must take into account the level of involvement of the teaching staff in the process, described in its different degrees by Barbosa [11], which reflects the greater or lesser participation in the elaboration and resolution of the problems.

In the first level, the teacher has greater participation. The teacher chooses the topic, simplifies and elaborates a problematic situation, presents the necessary data for its resolution, elaborates a mathematical model, and lets the students discuss the problem's solution, oriented at each step towards correct modeling.

At the second level, the teacher's direction of the activities is smaller. Although the teacher proposes a problem situation and guides in its solution, it is the students themselves who are in charge of simplifying the problem and collecting the necessary data to obtain the final solution.

In the third level, although the teacher continues to guide the work, students have greater autonomy in all areas. They choose a topic, simplify and elaborate each problem situation, and collect and analyze the necessary data to provide a solution to the proposed modeling.

The choice of one or the other option when implementing modeling in the classroom depends on the teacher's experience and the interests and characteristics of the students. In this work, the focus is mainly on the didactic perspective and the second level.

When modeling, we apply mathematical concepts to different scientific areas, contributing to a vision of mathematics as a helpful tool and not just a set of isolated techniques to be memorized. However, we are aware that modeling in the classroom is a challenge for teachers, as it implies a more significant effort and dedication to planning activities and to evaluating them as well as to updating and revising them.

In the present work, we propose a problem based on the mechanics and esthetics of a medieval game to encourage learning the previous concepts from graph theory via modeling. The game included in this paper is designed to be solved with paper and pencil except for the application of algorithms, for which software support is available.

The problem posed is framed within the use of games in education, which is becoming increasingly popular in higher education. In [12], the authors present survey results showing that engineering students in particular see a great deal of potential in the use of educational games. In [13], the authors show how engineering students positively evalu-

ated the usefulness of games that illustrate course content and mathematical calculations used in STEM courses. A flipped classroom model combined with games, provided in [14], has been shown to strengthen and improve the way Mathematics learning and teaching content, strategies, and approaches are enacted in different levels of education, including higher education. A review collecting the findings of the state-of-the-art literature on the use of games in science education can be found in [15]. Recent meta-analyses conclude that the use of games in education has a significant positive effect on student learning [16,17].

In this paper, the first section is devoted to introducing the notions and concepts related to the learning of weighted graphs, reachability, and the algorithms needed to solve the shortest path problem. In the second section, we present the above-mentioned problem, and provide several questions to ask the students during the resolution of the problem in order to develop reasoning and modeling skills. This problem, together with others related to the real world, was included during the teaching of graphs in the first course of Discrete Mathematics as part of the Data Science degree at the Escola Tècnica Superior d’Enginyeria Informàtica (ETSINF) of the Universitat Politècnica de València (UPV). The students’ perception of their learning of graphs through modeling was highly satisfactory, as reflected in a short quiz answered by them at the end of the course; these results are commented on in the fourth section. Finally, we provide our conclusions in the last section.

2. Preliminaries

In this section, we introduce basic concepts which are needed for solving the proposed problem. A non-directed graph $G = (V, E)$ consists of a nonempty set, V , the elements of which are called vertices or nodes, and a set of edges, E , provided by unordered pairs of different vertices. The graph $G = (V, E)$ is said to be directed if the set E consists of ordered pairs of vertices $u, v \in V$, which we call arcs. In both directed and non-directed graphs, the vertices of an arc or edge are called endpoints. When an arc or an edge has u and v as endpoints, it is simply denoted as (u, v) .

A chain W from a vertex v_0 to a vertex v_l in a graph $G = (V, E)$ is an alternate sequence of vertices and edges (respectively, vertices and arcs), say, $v_0, e_1, v_1, e_2, v_2, \dots, e_l, v_l$, where $e_i = (v_{i-1}, v_i) \in E, 1 \leq i \leq l$. If, in addition, for all $i, j, v_i \neq v_j$ and $e_i \neq e_j$ for all i, j , then W is called a path. In order to simplify notation, we refer to this as $v_0, v_1, v_2, \dots, v_l$. If such a path exists, then we say that v_l is reachable by v_0 . Algorithms for determining reachability include BFS (Breadth First Search) and DFS (Depth First Search) (see [18–20] for more details). A chain $W = v_0, v_1, v_2, \dots, v_l$ is said to be a cycle if $l \geq 2$, all the vertices $v_i, 0 < i < l$ are different from each other, v_0 , and $v_0 = v_l$. If for any pair of different vertices v_i and v_j there exists a path from v_i to v_j , the graph is said to be connected.

In a graph G , if every edge (that is, each respective arc) (u, v) has an associated value $w(u, v)$, it is said to be a weighted graph. The cost matrix of a weighted graph $G = (V, E, w)$ with $|V| = n$ is an $n \times n$ matrix, denoted as $W = (w_{ij})$, the elements of which are provided by

$$w_{ij} = \begin{cases} 0 & \text{si } i = j, \\ w(v_i, v_j) & \text{si } i \neq j \text{ y } (v_i, v_j) \in E, \\ \infty & \text{si } i \neq j \text{ y } (v_i, v_j) \notin E. \end{cases}$$

The weight of a path W in a weighted graph is provided by the sum of all the weights associated with the edges (or arcs) that form it. In a weighted graph $G = (V, E, w)$, the shortest path problem involves finding the path with the least weight between every pair of vertices $u, v \in V$. The objective of the real-world problem we propose in this work consists of deepening the learning of different algorithms that provide the solution for finding the shortest path. These are Dijkstra’s algorithm, which can be applied whenever all the weights are positive, the Bellman–Ford algorithm, applicable when, while the

weights are not necessarily positive, there are no cycles with negative weight, and the Floyd–Warshall algorithm, which provides the shortest path between any pair of distinct vertices in the graph, again with the restriction that there are no cycles with negative weight. Note that these algorithms determine the accessibility between any pair of vertices in the graph. For more information about them, we refer to [10,18–21].

Recall the complexity of an algorithm is a measure of the amount of computational effort expended when the computer solves a problem using that algorithm. The complexity of the BFS and DFS search algorithms is $O(|V| + |E|)$, that of Dijkstra’s is $O(|V|^2)$, and that of Floyd–Warshall and Bellman–Ford’s algorithms is $O(|V|^3)$ [19,21,22].

3. A Teaching Proposal for Shortest Path Algorithms

In this section, we propose a modeling problem the solution to which requires several concepts of graph theory studied in the course of the Discrete Mathematics of the Data Science degree, such as directed graphs, weighted matrices, search algorithms, and shortest path algorithms. The methodology used in this subject consists, first of all, in justifying the need to define the different concepts and algorithms from simple examples of everyday life, thus introducing students gradually in modeling through graph theory. After each concept has been introduced, the student is asked to solve simple modeling problems devoted to practicing the corresponding single concept. References [23,24] that include many of the problems proposed during the development of the subject. After the student is already familiar with this type of exercise, we dedicate a two-hour class to solving the proposed problem in this paper, for the solution of which it is necessary to interrelate all of the concepts studied previously.

Due to the difficulty of the problem, the methodology we follow during its resolution consists of first analyzing the statement of the problem together with the students. Then, a question is proposed, leaving the pupil time to answer it on their own. After discussion between the students and the teacher about the different solutions provided, a new question is posed.

The problems proposed during the course for learning the different algorithms have been solved by hand. However, as the objective of this problem is to deepen in its correct application and not in its resolution, and considering the size of the graph that is posed, the students use our software, “SWGraphs” (see [25]), to obtain the solution. It should be mentioned that these algorithms can be solved in other ways, for example, with NetworkX from Python.

Regarding the evaluation, the proposed problem in this paper has been developed as an activity in class that the students solve under the the professor’s guidance. However, it is not included in student assessments; instead, in order to evaluate their understanding of modeling, students are asked to invent an original problem from a real context, model it with graphs, and solve it using the concepts and algorithms studied during the course.

In the development of the solution presented in this paper, and based on the difficulties encountered by students during our implementation, we include recommendations for teachers when guiding their students in solving the problem, highlighting the points where students may require help.

Problem 1. *In the medieval game “Legends of Al-Bufera”, the player Elner finds himself in the village Rojcatar with a loot of 500 points, a map, and his mare Velatina. In Figure 2, it can be seen that the mentioned map includes the villages Rojcatar and Casue, different isolated houses and inns (labeled as I), and the kilometers separating them from each other.*

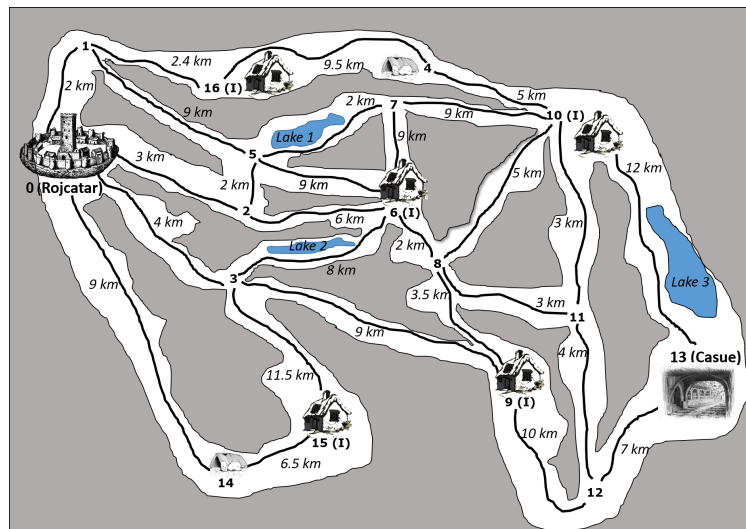


Figure 2. Map of the region of Al-Bufera.

His objective in this screen is to get to the village of Casue, keeping as many points as possible for buying supplies in the village store while bearing in mind the following:

- (i) Velatina travels at 6 km/h, and cannot travel more than 15 km without stopping at one of the inns along the route.
- (ii) Along certain trails there are lakes where Velatina stops for 10 min to drink, then jogs at 8 km/h up to the next crossroad.
- (iii) Elner loses as many points as the number of minutes he takes to travel each route on the map. However, he has the possibility of gaining points by transporting a package on certain routes, as shown in the Table 1:

Table 1. Benefit in points for transporting a package between two directly connected crossroads i and j , represented as $i - j$.

Route	2-6	3-15	5-6	9-8	12-11	16-4
Benefit	72	90	107	25	48	117

- (iv) The inns are located on the boundaries of the feudal lords' lands, and each time they pass through them Elner must pay the tax indicated in Table 2:

Table 2. Tax in points for passing through each feudal lord's lands.

Inn	6	9	10	15	16
Tax	2	5	3	10	5

- (v) Each time Elner passes through an inn, Velatina can travel up to 15 km again.
- (vi) The routes (2, 6), (5, 6), (16, 4), and (12, 11) are dangerous, and Elner will only risk going through them if he transports a package.

Considering the game conditions and using graph theory:

- (a) Help Elner determine the best possible route.
- (b) If Elner decided to avoid stopping at lakes in order to not lose points, what route would he follow in this case? Is this route more advantageous than the one obtained in (a) above?
- (c) Elner manages to overcome the challenge with the maximum number of points possible, and in a later screen, he finds himself with the possibility of using the same map with the added restrictions that the routes (10, 13) and (1, 16) have become inaccessible. Is Elner able to make it from Rojcatar to Casue in these new circumstances? If yes, indicate the best possible route. Otherwise, another challenge must be considered in order to move forward.

Solution 1. First of all, the student is asked to model the problem using a graph $G = (V, E)$, where V is the points marked on the map and the elements of E represent the routes between every two points which do not pass through intermediate points. Considering the different restrictions of the problem, such as the payment of taxes when leaving the inns and the possibility of transporting packages from one point to another in only one direction, the students should consider that G is a directed graph. Next, they must observe that G is a weighted graph with weights that must be decided in each section while taking into account the conditions indicated in each of them.

- (a) In this first section, the weighting is obtained by taking into account kilometers, taxes when leaving inns, stops at the lakes, and transporting of packages. More specifically, the weight of each arc is calculated as the sum of the following values:
- Considering that the average speed at which Velatina travels is 6 km/h, it takes 10 min to travel 1 km, which results in a weight of 10 points for each kilometer traveled.
 - If, at vertex i , there is an inn that charges a tax of c points, (as shown in Table 2), then c points must be added to the weight of each arc (i, j) .
 - If there is a lake in the arc (i, j) , the mare stops for 10 min, which is equivalent to a cost of 10 points. The average speed during the first half of the journey is then 6 km/h, that is, a consumption of 10 points per kilometer. During the second half of the route, following the indications of the problem, the average speed of Velatina is now 8 km/h, which is 7.5 points per kilometer traveled.
 - The transport of packages involves a benefit and not consumption. In order to reflect the fact that this is a gain, we associate the b points obtained by transporting a package from i to j with a negative weight, $-b$, which is added to the weight of the arc (i, j) .

We ask the students to observe that the final weight associated with each arc of the graph can be either positive or negative. From the above considerations, the students must construct the 17×17 matrix of weights (points) C associated with the graph G (see Figure 3) shown below.

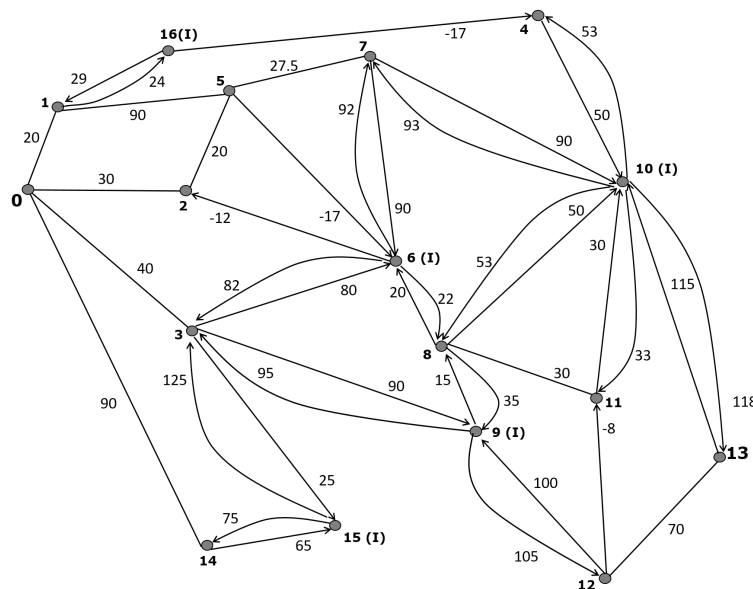


Figure 3. Graph associated with the map of Al-Buferá weighted with the costs obtained from the above calculations.

$$C = \begin{bmatrix} 0 & 20 & 30 & 40 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 90 & \infty & \infty \\ 20 & 0 & \infty & \infty & \infty & \infty & 90 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 24 \\ 30 & \infty & 0 & \infty & \infty & \infty & 20 & -12 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 40 & \infty & \infty & 0 & \infty & \infty & \infty & 80 & \infty & \infty & \infty & 90 & \infty & \infty & \infty & \infty & \infty & 25 & \infty \\ \infty & \infty & \infty & \infty & \infty & 0 & \infty & \infty & \infty & \infty & \infty & \infty & 50 & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & 90 & 20 & \infty & \infty & \infty & 0 & -17 & 27.5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 82 & \infty & \infty & \infty & 0 & 92 & 22 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & 27.5 & 90 & 0 & \infty & \infty & \infty & 90 & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty & 20 & \infty & 0 & 35 & 50 & 30 & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 95 & \infty & \infty & \infty & \infty & \infty & 15 & 0 & \infty & \infty & 105 & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & 53 & \infty & \infty & \infty & 93 & 53 & \infty & 0 & 33 & \infty & 118 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 30 & \infty & 30 & 0 & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 100 & \infty & -8 & 0 & 70 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 115 & \infty & 70 & 0 & \infty & \infty & \infty & \infty \\ 90 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 0 & 65 & \infty \\ \infty & \infty & \infty & 125 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 75 & 0 & \infty \\ \infty & 29 & \infty & \infty & -17 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 0 \end{bmatrix}.$$

We tell the students to approach the problem following an iterative process, taking into account that Velatina cannot travel more than 15 km without stopping at an inn.

We must determine whether or not Casue is reachable from Rojcatar by a route of fewer than 15 km, which we call admissible. As our goal is to find the path from Rojcatar to Casue that consumes the least points, we must analyze the admissible paths from Rojcatar to the different inns. The process continues by obtaining the admissible routes from the inns reached in the previous iteration to Casue and the unreached inns. The iterative process ends when the only admissible paths from the reached inns are those leading to Casue. The path that consumes the least number of points can be obtained by comparing all of the routes obtained by the previous procedure.

We point out to the students that in order to take into account the restriction on the maximum 15 km to be traveled without resting, a 17×17 matrix of weights K must be constructed where the position (i, j) corresponds to the number of kilometers between i and j .

$$K = \begin{bmatrix} 0 & 2 & 3 & 4 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 9 & \infty & \infty \\ 2 & 0 & \infty & \infty & \infty & 9 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 2.4 \\ 3 & \infty & 0 & \infty & \infty & 2 & 6 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 4 & \infty & \infty & 0 & \infty & \infty & 8 & \infty & \infty & 9 & \infty & \infty & \infty & \infty & \infty & \infty & 11.5 & \infty \\ \infty & \infty & \infty & \infty & 0 & \infty & \infty & \infty & \infty & \infty & 5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & 9 & 2 & \infty & \infty & 0 & 9 & 2 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 8 & \infty & \infty & 0 & 9 & 2 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & 2 & 9 & 0 & \infty & \infty & 9 & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & 2 & \infty & 0 & 3.5 & 5 & 3 & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 9 & \infty & \infty & \infty & \infty & 3.5 & 0 & \infty & \infty & 10 & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & 5 & \infty & \infty & 9 & 5 & \infty & 0 & 3 & \infty & 12 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 3 & \infty & 3 & 0 & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 10 & \infty & 4 & 0 & 7 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 12 & \infty & 7 & 0 & \infty & \infty & \infty & \infty \\ 9 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 0 & 6.5 & \infty & \infty \\ \infty & \infty & \infty & 11.5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 6.5 & 0 & \infty \\ \infty & 2.4 & \infty & \infty & 9.5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 0 \end{bmatrix}.$$

Next, we develop the solution to the problem, paying special attention to those details that should be emphasized to help students solve it.

First stage

First, we ask the students which problem in the context of graph theory corresponds to the question posed about determining the best possible path. Bearing in mind that this path is the one that minimizes the number of points consumed, the students should identify that the objective is to find the path of minimum cost (usually called the shortest path) while taking into account the given constraints.

The first of these states that Velatina cannot travel more than 15 km without stopping at an inn; thus, we have to start by determining whether Casue and any of the inns are accessible from Rojcatar by an admissible path. In order to do this while taking into account that all of the weights (kilometers) are positive, we apply Dijkstra’s algorithm to vertex 0 with the matrix of weights K. After applying this algorithm, we can observe that the shortest path from the origin vertex to our destination (vertex 13) has a weight of 25 km, and cannot be reached

without stopping at inns. On the other hand, we can reach the inns located at vertices 6, 9, and 16 in 9, 13, and 4.4 km, respectively. We discard inns 10 and 15 because the distance from Rojcatar to them is 16 and 15.5 km, respectively.

We ask the students whether the shortest paths provided by Dijkstra's algorithm necessarily correspond to those that minimize the number of points consumed. They should realize that the answer is negative because the latter path has to be calculated by considering the weights of the matrix of points C.

Before calculating these paths, we emphasize that matrix C has negative values, and therefore Dijkstra's algorithm is not applicable. It is therefore necessary to make use of the Bellman–Ford or Floyd–Warshall algorithm. Applying one of these algorithms to vertex 0, we can find that the shortest paths to the inns located at vertices 6, 9, and 16 are:

- To the one located in vertex 16: 0-1-16 with a weight of 49 points and 4.4 km.
- To the one located in vertex 9: 0-2-6-8-9 with a weight of 75 points and 14.5 km.
- To the one located in vertex 6: 0-2-6 with a weight of 18 points and 9 km.

We note that certain paths provided by the Bellman–Ford algorithm may not be admissible because they are longer than 15 km; therefore, these are not considered in the present analysis. In that case, a combinatorial study comparing all of the possibilities should be made.

Second stage

As in the previous stage, when looking for the shortest paths in kilometers we apply Dijkstra's algorithm, considering matrix K, to each of vertices 6, 9, and 16 to reach Casue (vertex 13) directly or any of the inns not reached in the previous stage. In this way, we obtain the following paths:

- From the inn located in vertex 6:
 - * **To the inn located in vertex 10: 6-8-10 (7 km)**
 - * To the inn located in vertex 15: 6-3-15 (19.5 Km)
 - * To Casue: 6-8-9-12-13 (22.5 km)
- From the inn located in vertex 9:
 - * **To the inn located in vertex 10: 9-8-10 (8.5 km)**
 - * To the inn located in vertex 15: 9-3-15 (20.5 Km)
 - * To Casue: 9-12-13 (17 km)
- From the inn located in vertex 16:
 - * **To the inn located in vertex 10: 16-4-10 (14.5 km)**
 - * To the inn located in vertex 15: 16-1-0-14-15 (19.9 Km)
 - * To Casue: 16-4-10-13 (26.5 km)

We have highlighted in bold all admissible paths with weightings of fewer than 15 km. We point out to the students that it is not necessary to perform the above study between the already-reached inns i and j ; the shortest path from the origin vertex 0 to an inn i concatenated with the shortest path from inn i to j cannot be shorter than the one provided by Dijkstra's algorithm from 0 to j , as the weights are positive.

Following the same reasoning as in the previous stage, we can apply the Bellman–Ford algorithm using the matrix C to obtain the shortest paths in points from vertices 6, 9, and 16 to 10, which weigh less than 15 km, obtaining:

- From the inn located in 6 up to the one in vertex 10: 6-8-10 with a weight of 75 points and 7 km.
- From the inn located in 9 up to the one in vertex 10: 9-8-10 with a weight of 65 points and 8.5 km.
- From the inn located in 16 up to the one in vertex 10: 16-4-10 with a weight of 36 points and 14.5 km.

We note now that there are no inns left to reach, and the only viable option is to find a route less than 15 km to Casue from inn 10.

We now ask the students whether searching for the best path in points is sufficient to consider the shortest of those routes obtained in the first stage concatenated with the shortest path from the second one. We ask them to think of the fact that the answer is negative. For instance, the shortest path in points obtained in the first stage is the one from 0 to 6 with 18 points, which, concatenated with path 6-10, weighs 93 points. However, if the paths 0-16 and 16-10 are concatenated the total weight is 85 points.

Third stage

At this point, students should be able to continue solving the problem by reasoning as before. Specifically, they must apply Dijkstra's algorithm to obtain the shortest path (in km) from vertex 10 to Casue (vertex 13). The result is path 10-13 of 12 km. Observing that this path is admissible, they should apply the Bellman–Ford algorithm again to obtain the shortest path from 10 to 13 by considering matrix C. This algorithm again provides the admissible path 10-13 with a weight of 118 points.

We ask the students which are the possible routes from Rojcatar to Casue. The answer should be:

- The path 0-2-6-8-10-13, as a result of the concatenation of the paths 0-2-6, 6-8-10, and 10-13, with a cost of 211 ($18 + 75 + 118$) points.
- The path 0-2-6-8-9-8-10-13, as a result of the concatenation of the paths 0-2-6-8-9, 9-8-10, and 10-13, with a cost of 258 ($75 + 65 + 118$) points.
- The path 0-1-16-4-10-13, as a result of the concatenation of the paths 0-1-16, 16-4-10, and 10-13, with a cost of 203 ($49 + 36 + 118$) points.

Therefore, the best possible route is the path 0-1-16-4-10-13, with a cost of 203 points; when taking this route, upon arriving in Casue the player has 297 points to buy supplies at the village store.

- (b) The students should be able to solve this section after the study carried out in section (a). The new cost matrix includes only the package constraints, not the lake ones. Because Elner does not stop at lakes, the weights of the edges containing a lake should be obtained only by considering their distance in kilometers and whether a package is transported from one end to the other. The new cost matrix matches C from the previous section, except for the (5,7) and (7,5) entries, which change from being 27.5 to 20, the (10,13) entry, which changes from 118 to 123, and the (13,10) entry, which changes from 115 to 120. The (3,6) and (6,3) edges keep the same weight, 80.

The procedure from section (a) is repeated, obtaining the same paths again except with different weights. In this case, the admissible paths from Rojcatar to Casue without stopping at lakes are:

- The path 0-2-6-8-10-13, as a result of the concatenation of the paths 0-2-6, 6-8-10, and 10-13, with a cost of 213 ($18 + 75 + 120$) points.
- The path 0-2-6-8-9-8-10-13, as a result of the concatenation of the paths 0-2-6-8-9, 9-8-10, and 10-13, with a cost of 260 ($75 + 65 + 120$) points.
- The path 0-1-16-4-10-13, as a result of the concatenation of the paths 0-1-16, 16-4-10, and 10-13, with a cost of 205 ($49 + 36 + 120$) points.

Therefore, the path Elner should take is 0-1-16-4-10-13, arriving at Casue with 295 points. Although intuitively one might think that not letting Velatina rest at the lakes would decrease the number of points spent, this is not true.

- (c) This section aims to check whether the students have assimilated the techniques, algorithms, and reasoning used in the previous sections. In this case, the matrices to be used to find the costs in points and kilometers coincide with the matrices C and K from section (a), except for the entries (10,13), (13,10), (1,16), and (16,1), which are weighted with ∞ .

On this occasion, we now include a new question related to reachability in graph theory: whether vertex 13 (Casue) can be reached from vertex 0 (Rojcatar). Although this question is solvable using Dijkstra's algorithm, as the weights are all positive, it is convenient to remind the students that using one of the BFS or DFS search algorithms is more appropriate as their

complexity is $O(|V| + |E|)$ and that of Dijkstra's algorithm is $O(|V|^2)$. The application, for example, of the BFS algorithm provides the path 0-3-9-12-13.

After following the same reasoning as in section (a), the following results are obtained:

In the first stage, the inns accessible from the origin vertex are:

- The inn located at vertex 9: 0-2-6-8-9, with a weight of 75 points and 14.5 km.
- The inn located at vertex 6: 0-2-6, with a weight of 18 points and 9 km.

In the second stage, we conclude that the only inn accessible from 6 and 9 is the one located in vertex 10, reached via the following paths:

- From the inn located in 6 up to the one in vertex 10: 6-8-10, with a weight of 75 points and 7 km.
- From the inn located in 9 up to the one in vertex 10: 9-8-10, with a weight of 65 points and 8.5 km.

We note that there are no inns left to reach; thus, the only viable option is to find the shortest path of less than 15 km from the inn located in vertex 10 to Casue.

Applying Dijkstra's algorithm to obtain this path using matrix K , we obtain 10-8-9-12-13, with a weight of 25.5 km. The students must conclude that Elner cannot reach Casue with his mare Velatina from Rojcatar due to the constraints of the game. Therefore, Elner must perform an alternative challenge to continue his adventure.

4. Students' Feedback about Modeling with Graphs

In this section, we include the results of a brief survey circulated to obtain feedback from students in a single group of 72 students after implementing this methodology based on modeling for the teaching of graph theory in a Discrete Mathematics subject in the first year of the degree in data science at the ETSINF of the UPV. The number of students in the group who answered the survey was 51 (70.8%), 34 of them men (66.7%) and 17 women (33.3%). The students followed non-distance education. Of the students who answered the survey, 1.96% were chinese and 98.04% were hispanic. All of the students who answered the survey were taking the course for the first time.

During the development of this subject several real-world problems were proposed, including the one presented in this article. In order to corroborate whether the approach taken when teaching the subject was adequate and to know the students' opinions about it, we asked them their level of agreement with the following assertions:

1. I found the modeling interesting, and it has helped me to understand graph theory better.
2. The methodology applied in the course was adequate.
3. The proposed exercises and tasks have helped improve my learning.

Below, we reflect the level of agreement between 1 and 5, where 1 means "Strongly disagree" and 5 means "Strongly agree".

Figure 4 shows the students' perceptions about assertion 1. It is clear that most of the students agreed or strongly agreed (24% and 62%, respectively) on the positive effect of mathematical modeling for understanding graph theory.

Modeling helped me to understand the graph theory

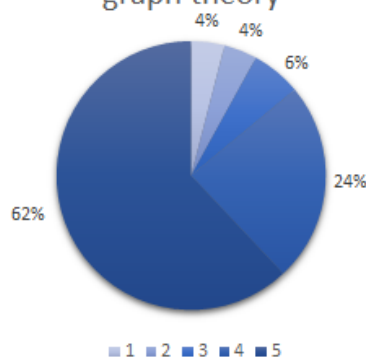


Figure 4. Students' perceptions of assertion 1.

Figure 5 shows that more than half of the students (53%) considered the methodology used in the course very appropriate and 37% of them considered it quite proper.

The methodology applied in the course is adequate

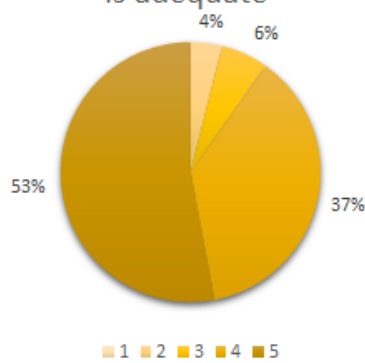


Figure 5. Students' perceptions of assertion 2.

Finally, Figure 6 shows the students' perceptions of whether the proposed mathematical modelling tasks and exercises improved their learning. Most students strongly agreed (27%) or agreed (57%) with this statement.

The proposed exercises and tasks have helped improve my learning

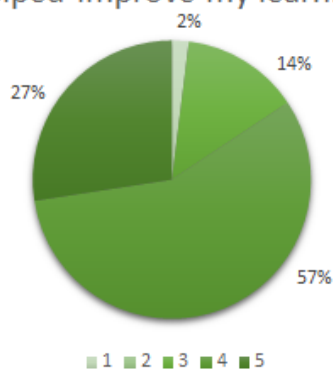


Figure 6. Students' perceptions of assertion 3.

In addition, we encouraged the students to express their opinion about modeling and the learning of graph theory in an open answer question. In general, most of the answers were very positive. We collect here the most representative ones:

- *The realization of graph modeling problems has seemed to be a very interesting aspect because they help understand concepts and put into practice the knowledge acquired.*
- *I have found graphs very interesting. I think I have assimilated and understood all the concepts that have been taught in the course.*
- *The learning methodology used is useful since I consider that I know how to pose, identify and solve graph problems.*
- *I liked the graph theory quite a lot, both for the enjoyable aspect of some problems and for the real applications.*
- *I think that problem modeling methodology helps students a lot and is a great tool to assimilate theoretical concepts through practice.*
- *I found the graph exercises very interesting and important to know the real application of the algorithms, etc.*

5. Conclusions

Graph theory is an area of mathematics that is particularly suitable for introducing students to mathematical modeling, as it is a useful tool for tackling a wide range of problems in diverse areas such as biology, engineering, physics, and computer science. In the case of engineering students with an eminently applied and practical profile, from our point of view, mathematical modeling should be the main methodology applied in the classroom, without forgetting rigor in approaching problems but far from the typical passive, theoretical, and repetitive learning.

The proposed problem in this work encourages students to develop abstraction, deduction, and generalization and facilitates the meaningful development of graph theory concepts, as shown in their answers collected in the previous section. On the other hand, the proposed problem shows students how to apply graph theory in everyday situations and non-academic contexts within their area of interest, providing a practical application of the mathematical concepts worked on in the classroom. The objective of proposing a problem based on a game was to bring mathematics closer to students' hobbies, which helps to attract their attention, as observed during the development of the problem's resolution in class. In addition, based on the correction of the original works presented by the students at the end of the course, we have noticed a higher degree of comprehension of the subject matter than when it is approached from a more purely theoretical point of view.

In this way, as a consequence of our experience, we have succeeded enhancing the positive attitudes of our students towards science, especially in mathematics. From our experience, we should mention that teaching graph theory through modeling supposes a greater workload on the part of professors, as the search for and design of modeling problems is much harder than simply solving theoretical exercises. Another aspect to take into account is the need to dedicate an entire class to solving a single problem, which can be difficult due to time constraints in the completion of established syllabi.

Finally, we would point out that the aim of the present work is to demonstrate an interesting modeling problem in order to serve as a starting point for the application of this methodology by other teachers. The objective of the survey was to obtain a first impression from our students of different aspects of the methodology. As a future work, we intend to repeat the experience in two groups, one of which will follow the same methodology we used in this paper, and the other of which will follow a more theoretical approach without emphasis on modeling. We intend to compare the levels of learning and satisfaction as well as the workload in both groups. Another future approach could be the implementation of the game as a video game.

Author Contributions: All authors have equally contributed to this work. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially supported by grant PGC2018-095896-B-C22 and PGC2018-094889-B-100, funded by MCIN/AEI/10.13039/501100011033 by “ERDF A way of making Europe”, the European Union.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Jordán, C.; Murillo-Arcila, M.; Torregrosa, J.R. The STEM Methodology and Graph Theory: Some Practical Examples. *Mathematics* **2021**, *9*, 3110. [CrossRef]
- Ramle, R.; Rosli, D.I.; Nathan, S.S.; Berahim, M. Question-led approach in designing Dijkstra algorithm game-based learning: A pilot study. *Int. J. Eval. Res. Educ.* **2020**, *9*, 926–933. [CrossRef]
- Sánchez-Torrubia, M.G.; Torres-Blanc, C.; López-Martínez, M.A. PathFinder: A Visualization eMathTeacher for Actively Learning Dijkstra’s Algorithm. *Electron. Notes Theor. Comput. Sci.* **2009**, *224*, 151–158. [CrossRef]
- Braicovich, T.; Cognigni, R.M. Coloreando la geografía del plano al toroide. *Números* **2011**, *76*, 135–148.
- Freudenthal, H. Why to teach mathematics so as to be useful. *Educ. Stud. Math.* **1968**, *1*, 3–8. [CrossRef]
- Blum, W.; Leiss, D. How do Students and Teachers Deal with Modelling Problems? In *Mathematical Modelling: Education, Engineering and Economics*; Haines, E.C., Galbraith, P., Blum, W., Khan, S., Eds.; Horwood: Chichester, UK, 2007; pp. 222–231.
- Årlebäck, J.B. Exploring the Solving Process of Group Solving Realistic Fermi Problems from the Perspective of the Anthropological Theory of Didactics. In *Proceedings of the Seventh Conference of European Research in Mathematics Education (CERME 7)*; Pytlak, M., Rowland, T., Swoboda, W., Eds.; CERME: Rzeszów, Poland, 2009; pp. 1010–1020.
- Blum, W.; Borromeo-Ferri, R. Mathematical Modelling: Can It Be Taught And Learnt? *J. Math. Model. Appl.* **2009**, *1*, 45–58.
- Gallart, C.; Ferrando, I.; García-Raffi, L.M. *Implementación de Tareas de Modelización Abiertas en el aula de Secundaria, Análisis Previo*; González, E., Teresa, M., Myriam, C., David, A., Tomás, O., Eds.; Investigación en Educación Matemática; SEIEM: Salamanca, Spain, 2014; pp. 327–336.
- Jordán, C.; Murillo-Arcila, M.; Seoane-Sepúlveda, J.B. *Teoría de Grafos y Modelización. Problemas Resueltos*; Paraninfo: Madrid, Spain, 2022; ISBN 9788413679280.
- Barbosa, J. Modelagem e Modelos Matemáticos na Educação Científica. *Alexandria Revista de Educação em Ciência e Tecnologia* **2009**, *2*, 69–85.
- Lenz, L.; Stehling, V.; Richert, A.; Isenhardt, I.; Jeschke, S. Of Abstraction and Imagination: An Inventory-Taking on Gamification in Higher Education. In *Proceedings of the 11th European Conference on Game-Based Learning (ECGBL)*, Graz, Austria, 5–6 October 2017.
- Cook-Chennault, K.; Villanueva Alarcón, I.; Jacob, G. Usefulness of Digital Serious Games in Engineering for Diverse Undergraduate Students. *Educ. Sci.* **2022**, *12*, 27. [CrossRef]
- Lameras, P.; Moumoutzis, N. Towards the gamification of inquiry-based flipped teaching of mathematics a conceptual analysis and framework. In *Proceedings of the 2015 International Conference on Interactive Mobile Communication Technologies and Learning (IMCL)*, Thessaloniki, Greece, 19–20 November 2015; pp. 343–347.
- Kalogiannakis, M.; Papadakis, S.; Zourmpakis, A.-I. Gamification in Science Education. A Systematic Review of the Literature. *Educ. Sci.* **2021**, *11*, 22. [CrossRef]
- Sailer, M.; Homner, L. The Gamification of Learning: A Meta-analysis. *Educ. Psychol. Rev.* **2020**, *32*, 77–112. [CrossRef]
- Shurui, B.; Hew, K.F.; Huang, B. Does gamification improve student learning outcome? Evidence from a meta-analysis and synthesis of qualitative data in educational contexts. *Educ. Res. Rev.* **2020**, *30*, 100322.
- Bang-Jensen, J.; Gutin, G. *Digraphs-Theory, Algorithms and Applications*; Springer: Berlin/Heidelberg, Germany, 2002.
- Chartrand, G.; Oellerman, O.R. *Applied and Algorithmic Graph Theory*; McGraw Hill: New York, NY, USA, 1993.
- Gross, J.L.; Yellen, J. *Graph Theory and Its Applications*; CRC Press: Boca Raton, FL, USA, 2006.
- Jordán, C.; Torregrosa, J.R. *Introducción a la Teoría de Grafos y sus Algoritmos*; Reverte-UPV: Valencia, Spain, 1996.
- Christofides, N. *Graph Theory. An Algorithmic Approach*; Academic Press Inc.: Cambridge, MA, USA, 1986.
- Jordán, C.; Burriel, J.; Herráiz, R. Un problema a resolver con los algoritmos de caminos más cortos. *Model. Sci. Educ. Learn.* **2011**, *4*, 263–273. [CrossRef]
- Jordán, C.; Sanabria-Codeçal, E. Aprendiendo a modelizar con grafos. *Pensam. Mat.* **2021**, *11*, 55–66.
- Chamorro Molina, J. Solving With Graphs 2.0. 2014. Available online: <http://hdl.handle.net/10251/36155> (accessed on 1 June 2022).