

Article

# Application of Graph Theory and Automata Modeling for the Study of the Evolution of Metabolic Pathways with Glycolysis and Krebs Cycle as Case Studies

Carlos De Las Morenas Mateos <sup>1</sup> and Rafael Lahoz-Beltra <sup>2,\*</sup>

<sup>1</sup> Department of Biodiversity, Ecology and Evolution (Biomathematics), Faculty of Biological Sciences, Complutense University of Madrid, 28040 Madrid, Spain

<sup>2</sup> Modeling, Data Analysis and Computational Tools for Biology Research Group, Complutense University of Madrid, 28040 Madrid, Spain

\* Correspondence: lahozraf@ucm.es

**Abstract:** Today, graph theory represents one of the most important modeling techniques in biology. One of the most important applications is in the study of metabolic networks. During metabolism, a set of sequential biochemical reactions takes place, which convert one or more molecules into one or more final products. In a biochemical reaction, the transformation of one metabolite into the next requires a class of proteins called enzymes that are responsible for catalyzing the reaction. Whether by applying differential equations or automata theory, it is not easy to explain how the evolution of metabolic networks could have taken place within living organisms. Obviously, in the past, the assembly of biochemical reactions into a metabolic network depended on the independent evolution of the enzymes involved in the isolated biochemical reactions. In this work, a simulation model is presented where enzymes are modeled as automata, and their evolution is simulated with a genetic algorithm. This protocol is applied to the evolution of glycolysis and the Krebs cycle, two of the most important metabolic networks for the survival of organisms. The results obtained show how Darwinian evolution is able to optimize a biological network, such as in the case of glycolysis and Krebs metabolic networks.

**Keywords:** evolution of metabolic networks; glycolysis; Krebs cycle; enzyme evolution; electronic enzyme; evolutionary graph theory



**Citation:** De Las Morenas Mateos, C.; Lahoz-Beltra, R. Application of Graph Theory and Automata Modeling for the Study of the Evolution of Metabolic Pathways with Glycolysis and Krebs Cycle as Case Studies. *Computation* **2023**, *11*, 107. <https://doi.org/10.3390/computation11060107>

Academic Editors: Gennady Bocharov, Fabrizio Mafessoni and Rainer Breitling

Received: 8 May 2023  
Revised: 22 May 2023  
Accepted: 25 May 2023  
Published: 28 May 2023



**Copyright:** © 2023 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

Graph theory is one of the main modeling tools used in biology [1] as it enables the mathematical analysis of networks composed of proteins, genes, transduction signals, metabolites, etc. As a modeling method, graph theory has provided, along with other modeling techniques, a procedure for the study of complex biological systems by means of a bottom-up approach. An example is the study of metabolic pathways (Figure 1), i.e., the succession of biochemical reactions, where an initial substrate is transformed into one or more final products through a series of intermediary molecules or metabolites. The different reactions of all metabolic pathways are catalyzed by enzymes, i.e., by proteins that accelerate the rate of a reaction. In DNA, genes encode proteins, and since an enzyme is a protein, the expression of a gene results in an enzyme. Consequently, metabolism is ultimately governed by genetic control.

Graph theory is very useful in the analysis of complex systems consisting of objects connected to each other, e.g., metabolites. Therefore, through the use of graphs, we can model the biochemical transformations involved in a metabolic pathway. In graph theory, it is usual to represent a metabolic pathway as a directed graph in which the nodes represent the metabolic substances and the edges represent the biochemical reactions catalyzed by

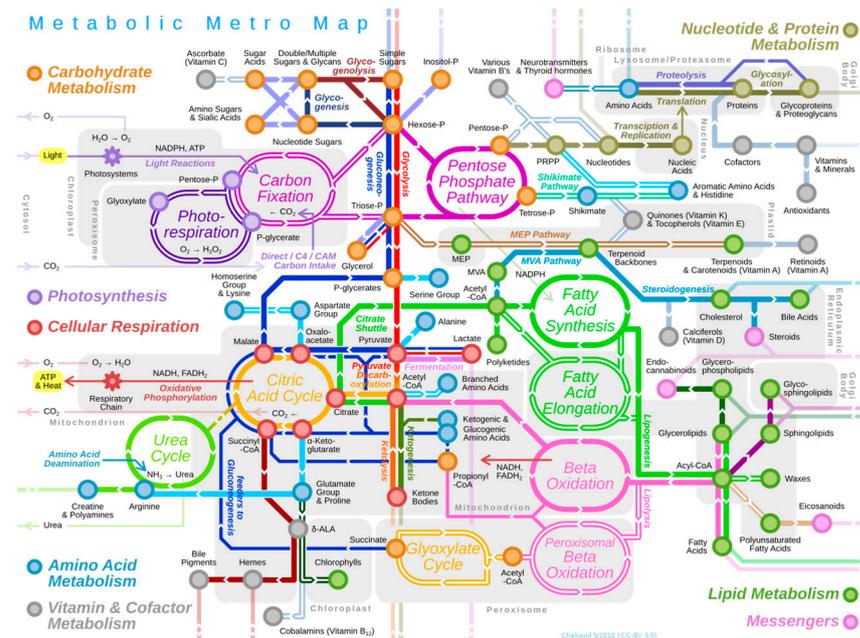
the enzymes. Thus, each of the enzymes involved in the metabolic pathway is identified by a label.

In agreement with [2], the use of graphs in biochemistry dates back to 1949. However, the study of metabolic networks as dynamic systems ‘within organisms’ is an issue that is not straightforward to address [3,4]. The current view of metabolism is based on the consideration of two opposing approaches. On the one hand, [5] views the metabolic network of an organism as a hierarchical organization of highly interconnected minor metabolic pathways. On the other hand, according to [6], the metabolic network of an organism exhibits the characteristics of a small-world network model similar to computer networks, neural networks, and certain social networks. According to the latter approach, the biochemical reactions would be represented in the nodes of a graph and the metabolites in the connections between nodes.

Now, while a graph captures the structure of a metabolic network, its behavior can be studied by means of a system of ordinary differential equations that model the rates of change of the concentrations of the different metabolites comprising the metabolic pathway. Given a vector  $S$  of metabolite concentrations and a vector  $v$  of reaction rates (or quantity of metabolic substance that is transformed in a given biochemical reaction per unit of time), the equation that describes the system of equations modeling the behavior of a metabolic pathway is as follows [7]:

$$\frac{dS}{dt} = N \cdot v \tag{1}$$

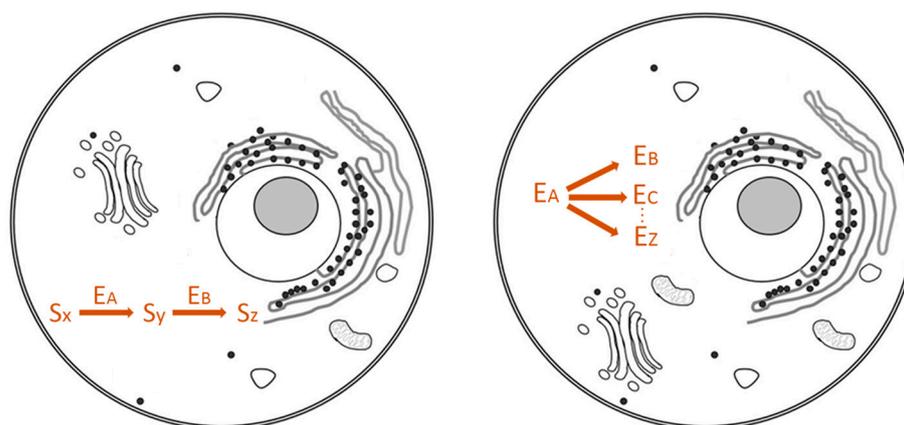
where  $N$  is referred to as the stoichiometric matrix, i.e., the matrix whose elements  $n_{ij}$  represent the amount of metabolite  $i$  formed in reaction  $j$ . Therefore, a metabolic network can be represented either as a graph or as a stoichiometric matrix ( $N$ -matrix), the latter representation being equivalent to a hypergraph [8].



**Figure 1.** Metabolic ‘Metro’ map (From Chakazul, 2016. Wikipedia, “[https://en.wikipedia.org/wiki/File:Metabolic\\_Metro\\_Map.svg](https://en.wikipedia.org/wiki/File:Metabolic_Metro_Map.svg)” (accessed on 27 May 2023)).

In this paper, using a simulation model we studied how each of the enzymes involved in a metabolic pathway could evolve during the course of Darwinian evolution, thereby, how the arcs or edges of a graph connecting one node (metabolite) to the next in a metabolic pathway evolved. In summary, how did metabolism evolve to yield today’s metabolic networks?

There are currently two hypotheses (Figure 2) about how metabolic networks could have evolved [9,10]. One of these hypotheses, known as the ‘retrograde model’, postulates that when a metabolite that acts as the substrate  $S_y$  of a metabolic reaction catalyzed by an  $E_B$  enzyme is depleted, then the  $E_B$  enzyme evolves, resulting in an  $E_A$  enzyme that precedes it, catalyzing the reaction that will produce the depleted substrate  $S_y$ . In contrast, the hypothesis that is referred to as the ‘patchwork hypothesis’ conjectures that primitive enzymes were quite unspecific, catalyzing the transformation of a wide range of metabolites or substrates. Over time, these nonspecific enzymes would have evolved into other specialized descending enzymes. It is now believed that this hypothesis could explain the topology of the metabolic networks of living organisms.



**Figure 2.** Evolution of metabolism: (left) retrograde evolution and (right) patchwork evolution hypotheses (see text for explanation).

In order to be able to answer this question, we assume that a metabolic network is a (a) dynamic system whose (b) structure is represented by a graph, studying through simulation the emergence during the evolution of the different enzymes involved in the catalysis of metabolic reactions. Likewise, in the graph depicting a metabolic pathway, (c) nodes represent the molecules or metabolites, and (d) arcs or edges represent the chemical reactions mediated by enzymes.

Of the set of metabolic pathways present in living organisms (Figure 1), we have chosen two of the most representative metabolic pathways: glycolysis and the Krebs or tricarboxylic acid (TCA) cycle. These are two metabolic pathways of great evolutionary significance in living organisms. Glycolysis is an ancestral metabolic pathway, i.e., one of the oldest in evolution, responsible for the oxidation of glucose into pyruvic acid. The result is the production of energy in the cell, and this metabolic pathway occurs in both oxygen-breathing and non-oxygen-breathing organisms. In oxygen-breathing organisms, the pyruvic acid resulting from glycolysis will undergo a series of chemical transformations in the so-called Krebs cycle or TCA cycle. The purpose of this cycle is to continue the oxidation of the metabolic intermediates, with the electrons from this oxidation being captured by citric acid, again resulting in additional energy for the cell. The final balance is the production of energy from glucose in the form of three molecular species (4 ATP, 10 NADH, and 2 FADH<sub>2</sub>).

Finally, and although this is an additional matter for the purpose of this paper, we also simulate the evolution of the enzymes involved in the alcoholic fermentation pathway because of their importance in the biotechnological industry.

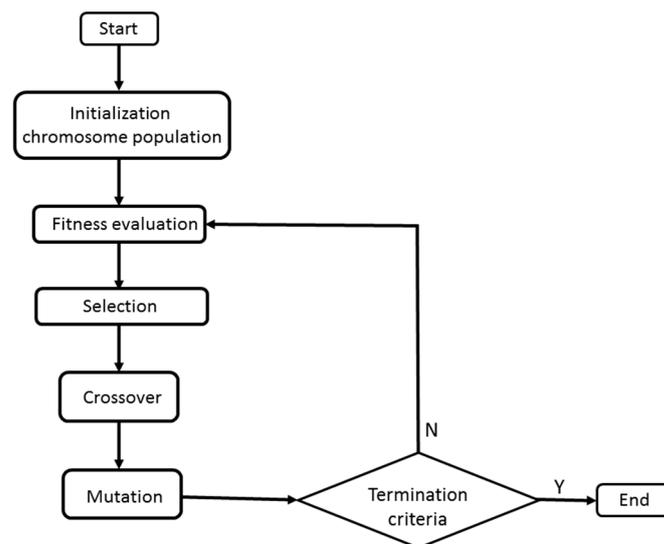
## 2. Materials and Methods

### 2.1. Simulating Evolution with a Genetic Algorithm

Genetic algorithms (GAs) are stochastic optimization methods inspired by the principles of natural selection [11] established by Darwin’s evolutionary theory in 1859. Currently,

they are used in optimization and search problems, being trial-and-error methods in which the solution space is continuously explored and evaluated until an optimum (global or local) is found. The search for an optimum simulates the survival of the most valid solutions or “individuals” in an environment or “problem space”. Since they were introduced by Holland [12] and popularized by Goldberg [13], GAs represent today one of the most widely used optimization techniques in the field of Artificial Intelligence (AI) due to their many applications in engineering, market research, commerce, operations development, biology, sociology, game theory, computer science, etc. [14].

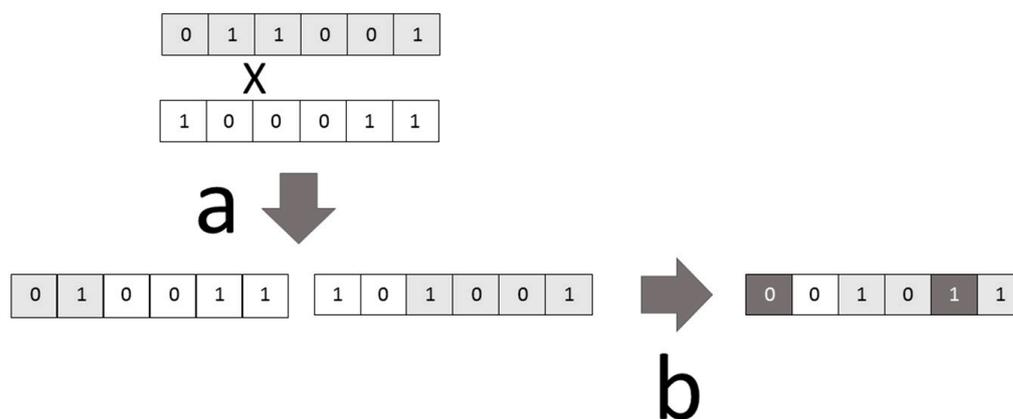
A GA involves the following steps or stages (Figure 3). First, (a) a set of solutions (individuals) is randomly generated. Individuals are represented by a vector that models the chromosomes, that is, the structure storing the genetic information (DNA) of a living being. In the present model, chromosomes were defined as binary strings. Afterward, (b) the fitness of chromosomes, goodness or validity of the solutions—a metaphor of adaptation to the environment—is evaluated. Those solutions with a higher fitness value will have a higher probability of surviving and, therefore, of passing to the next generation (gene propagation). The solutions that pass to the next generation will experience the effect of genetic mechanisms, in particular (c) mutations and genetic crossover, which are simulated in the GAs. The goal of these genetic mechanisms is to introduce variability in the population or set of solutions by promoting the search for new solutions. The search process is repeated over and over again until (d) an optimal solution is found.



**Figure 3.** Genetic algorithm (see text for explanation).

The search for new solutions, i.e., the exploration of the problem space, is the result of the simulation of mutation and crossover or recombination. In a GA, these mechanisms are simulated by means of the so-called genetic operators. In the present work, we have used the recombination operator at one point and the mutation operator for chromosomes defined as a binary string (flip-by-bit method).

According to Figure 4a, the one-point recombination was implemented as follows. Once the two parental chromosomes are chosen, a cutoff point is randomly selected, i.e., in the present example, at the second position, and the chromosome segments are then exchanged. The result of this mechanism is an offspring consisting of two recombinant chromosomes as they share the information of the parental chromosomes. A value of the  $p_r$  recombination rate is set at the beginning of the simulation experiment.



**Figure 4.** Genetic operators (see text for explanation). (a) One-point recombination. (b) Flip-by-bit mutation.

Mutation was simulated (Figure 4b) by randomly choosing one of the offspring chromosomes, e.g., the right one. Next, we randomly selected two positions, for example, the first and the fifth, flipping the value of the binary number ( $0 \rightarrow 1$ ,  $1 \rightarrow 0$ ). In a manner similar to recombination, the population mutation rate  $p_m$  is set at the beginning of the simulation experiment.

Simulation experiments were conducted using a simple genetic algorithm, SGA, customizing the algorithm to the goals of the present model. The code in Python language is available in [15].

## 2.2. Vectorization: Binary Representation of Metabolites

The first step in GA is to initiate the population or set of solutions, in our case, the enzymes that will transform one metabolite into another in a metabolic pathway. In the present paper, the formal representation of enzymes and metabolites is referred to as vectorization. Metabolites are molecules, and vectorization was performed as binary matrices according to the method developed in [16].

In the present model, the nodes represent the metabolic intermediates, i.e., sugars and other glycolytic molecules. It is important to note that the representation of a molecule of metabolism as a binary matrix is actually not a new approach, but the criterion adopted in this work is a novelty differing from the usual approach.

In 1974, ref. [17] translated the molecular topology into a matrix, taking an element  $a_{ij}$  with the value of 1 when the edges are adjacent or 0 otherwise. However, unlike this procedure, the method we have followed in this work assigns a 5-bit word to the functional groups of the molecule. To this end, we defined a table or “Rosetta stone” of cellular metabolism. According to Figure 5, we assigned a 5-bit word to each functional group (group of atoms in a molecule with distinctive chemical properties), ordering the most frequent functional groups in the metabolic intermediates depending on their redox potential (the tendency of a functional group to acquire electrons). Note how the vectors were classified by their redox potential from the most reduced form corresponding to the alkyl group to the most oxidized form represented by  $\text{CO}_2$ .

Decimal	Binary	"Functional group"		Red-Ox scale
0	000 00		} NULL VALUES	<div style="display: flex; align-items: center; justify-content: center;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">+Red -Ox</div> <div style="width: 100%; height: 100%; background: linear-gradient(to top, black 48%, white 48% 52%, black 52%);"></div> <div style="border: 1px solid black; padding: 5px; margin-left: 10px;">-Red -Cx</div> </div>
1	000 01			
2	000 10			
3	000 11			
4	001 00		} NULL VALUES	
5	001 01			
6	001 10			
7	001 11			
8	010 00	-CH <sub>3</sub>	} Alkyle	
9	010 01	-CH <sub>2</sub> -		
10	010 10	-CH-		
11	010 11	-C-		
12	011 00	=CH <sub>2</sub>	} Alkene	
13	011 01	=CH-		
14	011 10	=C-		
15	011 11	=C=		
16	100 00	-CH <sub>2</sub> OH	} Alcohol	
17	100 01	H-C-OH (D)		
18	100 10	HO-C-H (L)		
19	100 11	-C-OH		
20	101 00	-CH <sub>2</sub> O-P	} Ester	
21	101 01	H-C-O-P		
22	101 10	=C-O-P		
23	101 11	-C-O-P		
24	110 00	-C(=O)H	} Carbonyle (aldehyde, ketone)	
25	110 01	} NULL VALUES		
26	110 10			
27	110 11			
28	111 00	-C(=O)OH	} Carboxile, Ether, CO <sub>2</sub>	
29	111 01	-C(=O)O-P		
30	111 10	-C(=O)O-P		
31	111 11	CO <sub>2</sub>		

Figure 5. "Rosetta stone" of the cellular metabolism (see [16]).

In accordance with the above, the substrate *S* and the product *P* of a biochemical reaction catalyzed by an enzyme *E* were encoded in a matrix whose elements are zeros or ones [16]. It is interesting to note that all metabolites  $M_{i \times 5}$  in glycolysis and the Krebs cycle are molecules with *i* equal to 3, 4, 5, or 6 carbons. Thus, depending on the number of carbons in the molecule, we will represent a metabolite as follows:

$$\begin{aligned}
 M_{C_3} &= \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \end{pmatrix} & M_{C_4} &= \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \end{pmatrix} \\
 M_{C_5} &= \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{pmatrix} & M_{C_6} &= \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \\ a_{61} & a_{62} & a_{63} & a_{64} & a_{65} \end{pmatrix} \tag{2}
 \end{aligned}$$

It is important to indicate that given a value  $i$ , the row vector  $a_{i1}, a_{i2}, \dots, a_{i5}$  represents a functional group, either of the substrate  $s_{i5}$  or product  $p_{i5}$ . That is, each row of any of the above matrices represents a carbon atom of the molecule, providing that, according to Figure 5, a total of thirty-two possible binary vectors (from 00000 to 11111). However, since Krebs cycle metabolites are the result of the arrangement of twenty-two functional groups, then the ten binary vectors in Figure 5 are chemically meaningless for Krebs cycle metabolites. In the present work, in order to conduct simulation experiments including CO<sub>2</sub> and acetyl-CoA molecules, both molecules were modeled as a row vector  $v$  and an  $M_{2 \times 5}$  matrix:

$$v_{CO_2} = (1 \quad 1 \quad 1 \quad 1 \quad 1) \quad M_{acetyl-CoA} = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} \tag{3}$$

### 2.3. Modeling an Enzyme as Automaton

Once the metabolite molecules were represented as binary matrices, the enzymes that catalyze the transformation of one metabolite into another were modeled as automata.

Enzymes are proteins with catalytic properties that can increase and regulate the reaction rate. In this work, and as already mentioned above, we simulated the evolution of the enzymes and, subsequently, of the appearance or emergence of each of the arcs of the graph representing the metabolic network.

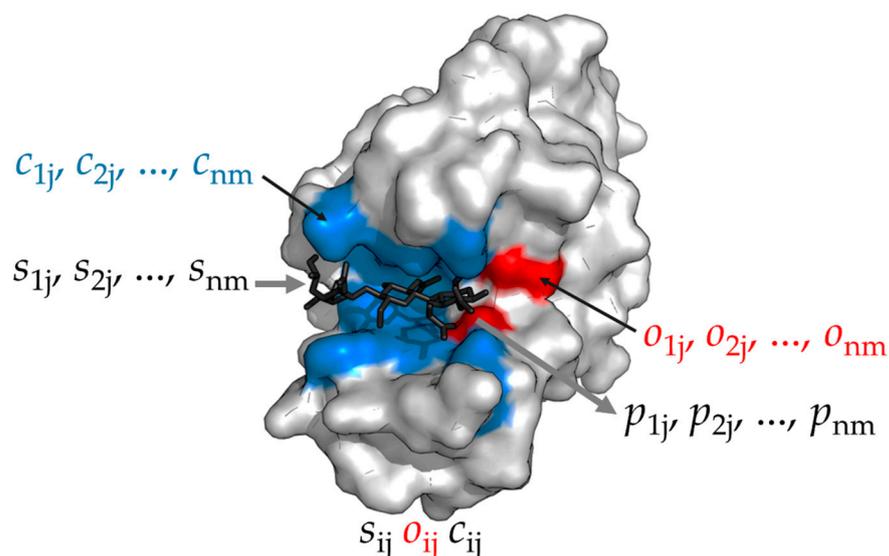
Since enzymes are proteins, unlike metabolites, which are sugars, their vectorization does not represent their chemical composition. An enzyme has the organization depicted in Figure 6. The region of the enzyme where the substrate binds is called the active site, is the place where the chemical reaction occurs. The conformational state of the active center is the molecular shape of the enzyme in the active center. Therefore, the conformational state determines the affinity between the enzyme and the substrate. Assuming an analogy to the key-lock concept, the active center of the enzyme would be the lock and the substrate the key. Once the substrate  $S$  binds to the active center of the enzyme  $E$ , and then the substrate is transformed into a product  $P$  by breaking and forming chemical bonds:



In (4), the amino acids, i.e., the basic building subunits of proteins, are responsible for the transformation of the substrate  $S$  into the product  $P$  of the reaction. From a biochemical point of view, these amino acids create an environment in the active center with different chemical characteristics (hydrophobic or hydrophilic, acidic or basic). Furthermore, temperature and pH can affect the enzymatic reaction, although this fact has not been included in the present model.

The role of enzymes is of such importance in biochemistry that they have frequently been the subject of study and computer simulation using a wide repertoire of modeling

and simulation techniques. Whether using the Michaelis–Menten differential equation, the Monte Carlo method, a molecular dynamics method, or the cellular automata approach, all of these techniques have attempted to capture the primary function and features of the enzymes. In the present work, we have modeled enzymes as automata [18].



**Figure 6.** Organization of the active center of an enzyme. The conformational state of the substrate binding site (blue) is modeled with the sequence  $c_{1j}, c_{2j}, \dots, c_{nm}$  while the catalytic site (red) is represented by  $o_{1j}, o_{2j}, \dots, o_{nm}$ . The substrate  $s_{1j}, s_{2j}, \dots, s_{nm}$  is the black colored molecule that will be transformed into a product  $p_{1j}, p_{2j}, \dots, p_{nm}$  by the  $s_{ij} o_{ij} c_{ij}$  rule (The figure corresponds to lysozyme and the substrate is peptidoglycan. Modified from Thomas Shafee, 2015. [https://commons.wikimedia.org/wiki/File:Enzyme\\_structure.svg](https://commons.wikimedia.org/wiki/File:Enzyme_structure.svg)).

Cellular automata are discrete space and time models that have been used to model biological systems [11]. A cellular automaton in two dimensions ( $D = 2$ ) is a checkerboard where each cell is called a “finite automaton” because the cell is in any of a finite number of states. The state of a given automaton  $S_t(x)$  located at cell  $x$  at time  $t$  changes depending on the states of neighboring automata  $S_t(x+x_0), S_t(x+x_1), \dots, S_t(x+x_{n-1})$  by applying a state transition rule (5) to update its state [11,19]:

$$S_{t+1}(x) = F(S_t(x + x_0), S_t(x + x_1), \dots, S_t(x + x_{n-1})) \tag{5}$$

Applying this approach, enzyme kinetics have been modeled on several occasions [18,20–22], including the simulation of metabolic networks, for instance, the mitogen-activated protein kinase signaling cascade [23], the hypothetical unidirectional network of enzymatic reactions studied by [24], etc.

In the field of molecular biology, the technique of cellular automata has been successfully used to simulate the dynamic activities inside cells. For example, this technique was useful in the 1990s, enabling the construction of simulation models in structural biology in which the dynamic behavior of the system being simulated was mediated by enzymes and/or polymerization and self-assembly phenomena [25]. This technique enables the construction of models, solving many of the limitations of classical differential equation models, such as the 3D graphical representation of a system composed of different classes of molecules. An example is the model [26] in which, using molecular automata, it was possible to simulate *Escherichia coli* membrane assembly, the operation of ATP synthase as well as the assembly of the bacterial flagellar motor.

### Electronic Enzyme Model

Enzymes were modeled as automata according to a class of automata introduced by [27], referred to as electronic enzymes. The electronic circuit modeling of an electronic enzyme is a hardware-level model of the active center of an enzyme, which is configurable via a genetic algorithm (see Section 2.1). The possible applications of this class of hardware were patented [28].

We define this kind of automata as follows. An electronic enzyme  $E_m$  is a vector resulting from concatenating two strings of  $n$  bits length, respectively:

$$E_m: (c_{1j}, c_{2j}, \dots, c_{nm}, o_{1j}, o_{2j}, \dots, o_{nm}) \tag{6}$$

where  $c_{1j}, c_{2j}, \dots, c_{nm}$  is an  $n$ -bit string ( $i = 1, 2, \dots, n$ ) representing the ‘conformational state’ of the enzyme  $E_m$  ( $j = 1, 2, \dots, m$ ). Thus, in the active center then  $n$ -bit string would be defining the binding sites and substrate orientation (Figure 6). In addition,  $o_{1j}, o_{2j}, \dots, o_{nm}$  are the Boolean operators AND ( $o_{ij} = 1$ ) or XOR ( $o_{ij} = 0$ ) modelling the amino acids of the active center that are involved in the reaction which converts the substrate  $S$  into a product  $P$ . As a consequence, the  $n$ -bit string  $o_{1j}, o_{2j}, \dots, o_{nm}$  would define the ‘catalytic site’ of the active center of the enzyme (Figure 6).

According to the  $E_m$  definition, an enzymatic reaction, i.e.,  $S_m + E_m \rightarrow P_m$ , is a transformation rule defined by the following logical operations:

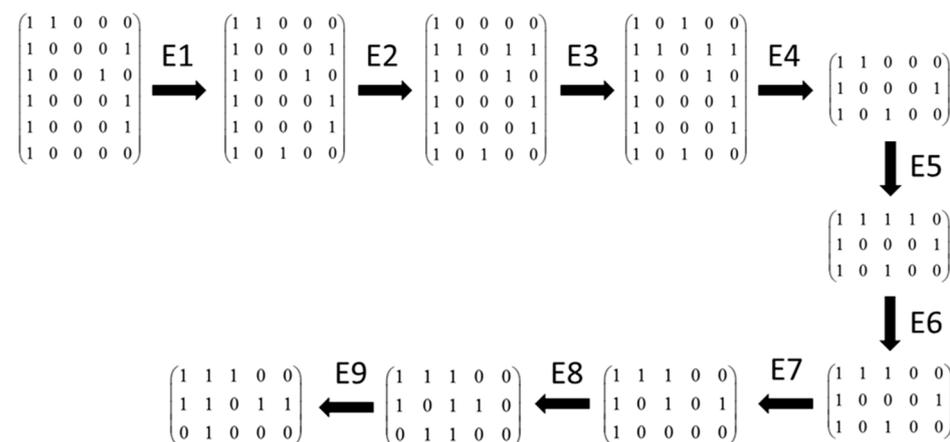
$$p_{1j} = s_{1j} o_{1j} c_{1j}, p_{2j} = s_{2j} o_{2j} c_{2j}, \dots, p_{ij} = s_{ij} o_{ij} c_{ij} \tag{7}$$

where  $s_{ij}$  and  $p_{ij}$  are the substrate and product resulting from the reaction (4).

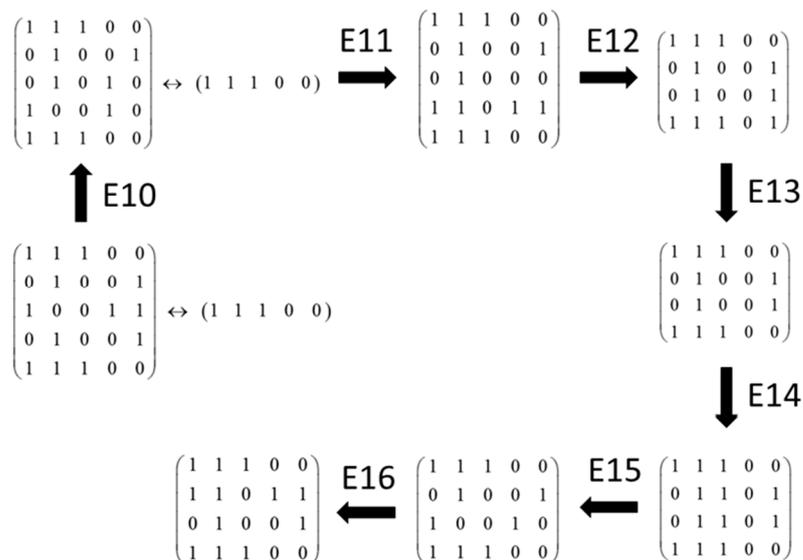
For example, let 1110 be the substrate  $S$  ( $i = 4$ ) and  $E$  be the enzyme (of reaction  $j$ ) defined by 10101000. In this example:  $p_{1j} = 1 \text{ AND } 1 = 1, p_{2j} = 1 \text{ XOR } 0 = 1, p_{3j} = 1 \text{ XOR } 1 = 0, p_{4j} = 0 \text{ XOR } 0 = 0$ , being the product  $P$  obtained from the reaction is 1100.

### 2.4. Binary Model of Glycolysis and Krebs Cycle

Applying the concepts and definitions described above, the glycolysis pathway was modeled, as shown in Figure 7. Likewise, using the same methodology, the Krebs cycle was modeled as has been depicted in Figure 8.



**Figure 7.** Binary encoding of the glycolysis pathway. E1: Glucose → Glucose-6-phosphate, E2: Glucose-6-phosphate → Fructose-6-phosphate, E3: Fructose-6-phosphate → Fructose-1,6-biphosphate, E4: Fructose-1,6-biphosphate → Glyceraldehyde-3-phosphate, E5: Glyceraldehyde-3-phosphate → 1,3-biphosphateglycerate, E6: 1,3-biphosphateglycerate → 3-phosphoglycerate, E7: 3-phosphoglycerate → 2-phosphoglycerate, E8: 2-phosphoglycerate → phosphoenolpyruvate, E9: phosphoenolpyruvate → pyruvate.

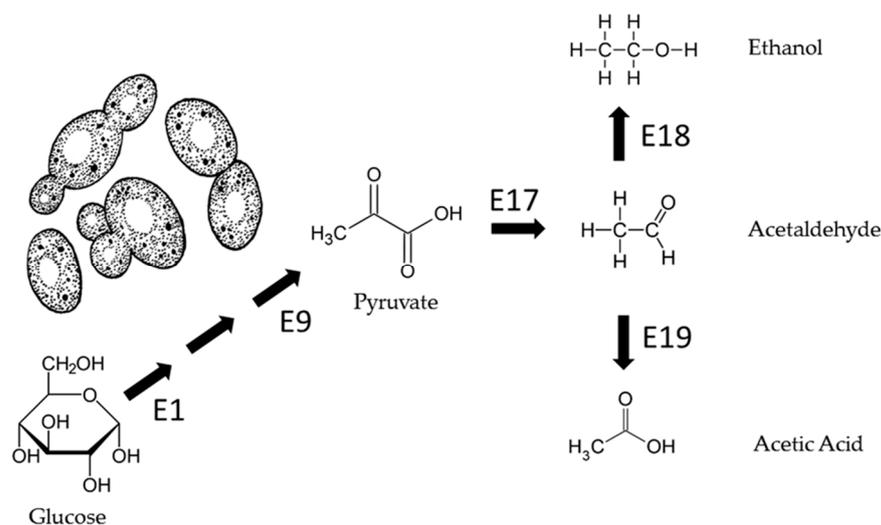


**Figure 8.** Binary encoding of Krebs cycle. E10: Citrate → isocitrate, E11: Isocitrate → α-ketoglutarate, E12: α-ketoglutarate → SuccinilCoA, E13: SuccinilCoA → Succinate, E14: Succinate → Fumarate, E15: Fumarate → malate, E16: Malate → oxazalate.

2.5. Simulating the Evolution of the Metabolic Pathway of Alcoholic Fermentation

After the simulation of glycolysis and Krebs or TCA cycle evolution, an additional experiment was conducted to simulate the evolution of the enzymes involved in the metabolic pathway known as alcoholic fermentation. Alcoholic fermentation is the metabolic pathway carried out in yeasts and some bacteria that is applied as a biotechnological process to convert sugars into ethyl alcohol and carbon dioxide.

We simulated the metabolic pathway illustrated in Figure 9, where each reaction is catalyzed by E17, E18, and E19 enzymes, respectively.



**Figure 9.** Biochemical model of alcohol fermentation in yeast. In the metabolic pathway of glycolysis, glucose is transformed by the action of successive enzymes (E1 . . . E9) into pyruvic acid. Fermentation begins with a first biochemical reaction catalyzed by the enzyme E17, in which pyruvate is transformed into acetaldehyde. The acetaldehyde is then transformed into ethanol through the reaction catalyzed by the E18 enzyme, or alternatively, the acetaldehyde is converted into acetic acid by the E19 enzyme. In the present simulation, we have not considered for simplicity that the reaction between acetaldehyde and ethanol is reversible.

## 2.6. Simulation Experiments

The enzymes  $E_m$  were defined as 10-bit vectors, the first 5 bits being the conformational state and the remaining 5 being the Boolean operators. The optimal binary string of the vectors modeling the substrates is searched evolutionarily via a genetic algorithm. An enzyme will be optimal if it is satisfied that the transformation of one or more of the rows of the matrix representing the substrate results in the reaction product matrix corresponding to a step within a metabolic pathway.

In the present article, we have omitted for simplicity the name, properties, and detailed explanation of the mechanism of action of the enzymes  $E$  involved in the biochemical reactions because the interested reader will find this information in any general biochemistry textbook.

From the set of matrices modeling the intermediates of glycolysis and Krebs cycle, the initial population ( $t = 0$ ) was obtained according to the following method.

An initial population  $P(0)$  consisting of 100 enzymes (binary chromosomes of 10-bit length) was obtained with a random number generator  $U [0, 1]$  with uniform distribution. The population  $P(t)$  will evolve during 200 generations such that once the 200th generation is reached, the genetic algorithm ends, obtaining the optimal solutions or chromosomes and, therefore, the optimal enzyme configurations. In each simulation experiment, 100 replicates were conducted, and since it is a stochastic optimization method, more than 2,000,000 solutions or enzymes were obtained from the initial population  $P(0)$ .

Based on the  $P(0)$  population of randomly generated chromosomes, reproduction, recombination and mutation were simulated by means of a genetic algorithm. In each generation, the configuration of the enzymes was evaluated, and its fitness value  $f_E$  was obtained according to the following evaluation function:

$$f_E = \alpha - \left( \beta \sum_{i,j=1}^l H(p_{kj}^*, s_{kj}) \right) \quad (8)$$

In the above function (8), indexes  $i, k$  represent the rows and  $j$  the column in the matrix  $S$ ,  $\alpha$  is a coefficient whose value was 100 in the simulation experiments,  $\beta$  is equal to the  $\frac{2^\alpha}{l}$  term, and  $l$  is the chromosome length. Thus, the enzyme fitness value is calculated by simulating the enzyme–substrate reaction (4), i.e., the operation between substrate and enzyme (7), and then comparing the  $p_{kj}^*$  product obtained from the reaction with the target product  $s_{kj}$ , i.e., the substrate of the next step or reaction of the metabolic pathway. The similarity between the obtained product and the target product or next substrate was calculated using the Hamming distance  $H(p_{kj}^*, s_{kj})$ . The Hamming distance is the number of positions in which two binary vectors or matrices with identical length or number of rows and columns differ, respectively. For this purpose, the chromosomes were decoded, resulting in the configuration of each enzyme, i.e., the conformational state ( $c_{1j}, c_{2j}, \dots, c_{nj}$ ) and the catalytic site ( $o_{1j}, o_{2j}, \dots, o_{nj}$ ) involved in the reaction.

Note that the genetic algorithm that simulates Darwinian evolution finds the appropriate configuration of the active center of an enzyme when the sum of the fitness function is zero. Then, once the chromosomes were evaluated, the new generation of chromosomes was obtained by applying the wheel parents selection algorithm [11]. In the chromosomes of the new generation, genetic variability is simulated through the genetic mechanisms of recombination and mutation. As explained above, recombination was performed with the one-point recombination operator, and mutation was simulated by applying the mutation operator based on the flip-by-bit method.

Based on the described methodology and using the fitness function (8), we conducted simulation experiments to find the optimal configurations of the enzymes involved in the metabolic pathways of glycolysis, Krebs cycle and alcoholic fermentation. The experiments were performed by assigning the following parameters in the SGA: population size  $N$  equal to 100 enzymes, maximum number of generations  $G_{\max}$  equal to 200, number of replicates

$r$  per experiment equal to 100, recombination rate  $p_r = 0.20$ , mutation rate  $p_m = 0.01$  (rate of change of the bit value  $p_b = 0.05$ ).

Once the results of the simulation experiments were collected, the average fitness of each generation was calculated, obtaining the performance graph of the genetic algorithm. In addition, an  $M_B$  molecular biodiversity value (in bits) was also obtained for the optimal enzymes (see Appendix B, Table A1) by applying the following expression:

$$M_B = \log_2 \frac{n!}{n_1! n_2! \dots n_k!} \quad (9)$$

where  $n$  is the total number of optimal enzymes, and  $n_1, n_2, \dots, n_k$  is the number of different molecular forms of the enzyme. Thus,  $n$  is the number of enzymes with the same conformational state and catalytic site found by the genetic algorithm, and  $n_k$  is the number of enzymes of class  $k$ . In expression (9) the factorials were approximated by Stirling's formula.

Since the different molecular forms (1, 2, ...  $k$ ) of the enzymes share a common catalytic activity they are called isoenzymes. Therefore, the above formula is a measure of molecular biodiversity, i.e., the richness and number of different enzymes catalyzing the same biochemical reaction.

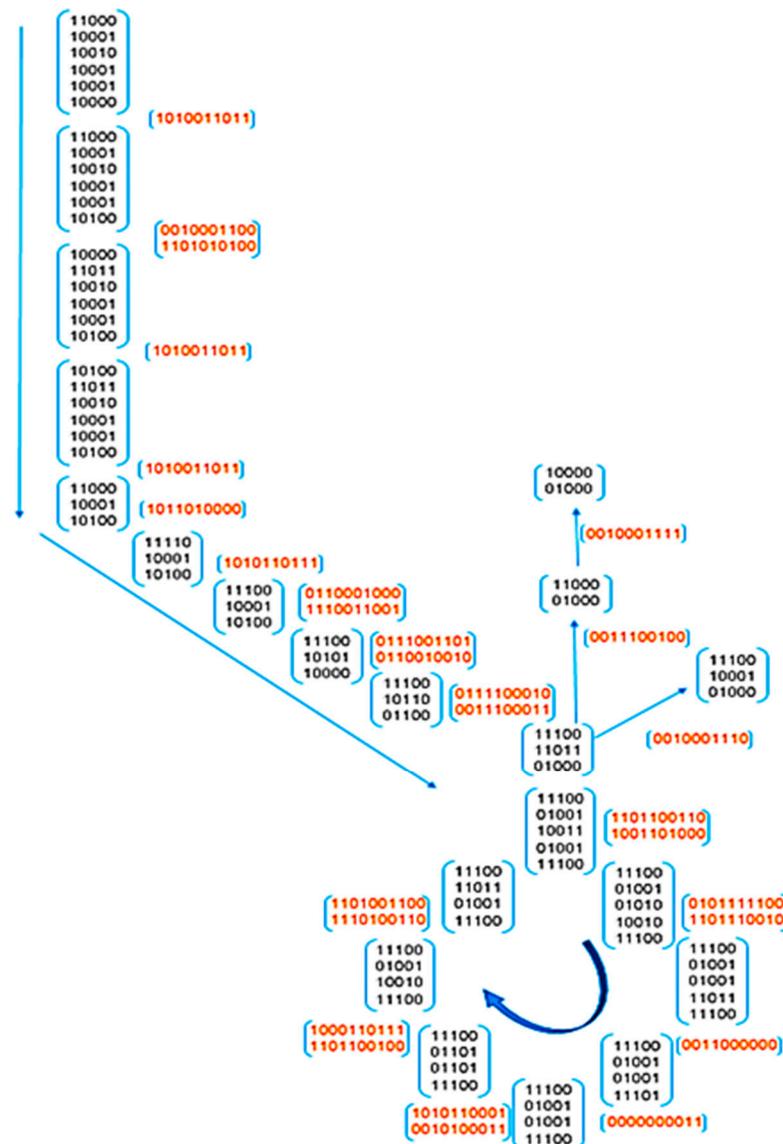
### 3. Results

The obtained results suggest that the simulations conducted capture essential elements in the evolution of metabolic pathways, and in particular, those related to intermediary carbon metabolism.

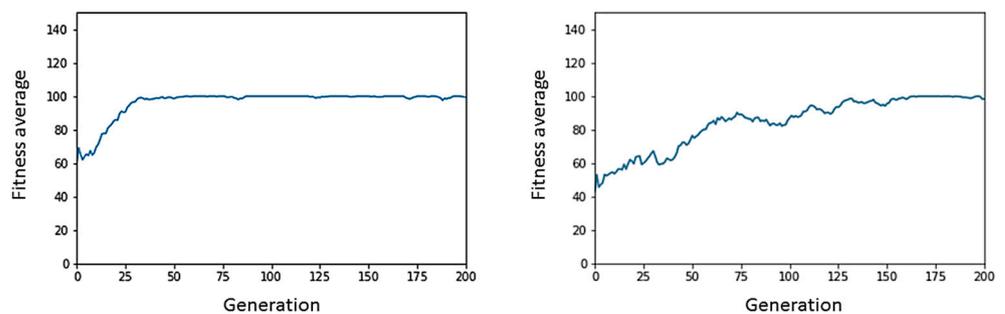
Figure 10 represents in a graph and using binary notation the flow of information in bits that would take place in a 'digital electronic circuit' that would be the hardware implementation of the central carbon metabolism, i.e., glycolysis and Krebs cycle. In Figure 11, we illustrate the classical performance graph of a genetic algorithm, thereby proving at the simulation level the Darwinian evolutionary convergence of enzymes in a hypothetical organism. Tables A2 and A3 (Appendix B) show the results obtained in the simulation of glycolysis and Krebs cycle, respectively.

Similar results were obtained for the evolution of the metabolic pathway of alcoholic fermentation. Table A4 (Appendix B) shows the results of the metabolic pathway of the simulated fermentation in yeast.

In addition, it is interesting to note that the Boolean operations modeling the catalytic site of the enzymes have an associated cost. The cost was expressed as energy consumption, penalizing the presence of XOR gates in the active center in the fitness value of the enzyme. Hence, in the simulations, whereas for the AND gate, the cost was equal to 1, the cost for the XOR gate was 1.5. In these simulation experiments, the result was a decrease in the biodiversity of optimal enzymes, emerging along the evolutionary pathway only those enzymes with the lowest cost and, therefore, with the lowest energy cost. We found only eight enzymes that included the XOR operator in their active center. However, in the general experiments where this restriction is not present in the fitness calculation, we obtained 54 optimal enzymes capable of catalyzing the biochemical reaction, despite the fact that their active center includes a greater number of XOR operators.



**Figure 10.** Simulation of the evolution of enzymes involved in the reaction steps of glycolysis and Krebs cycle. The 5-bit black vectors denote the functional groups of the metabolites (3, 4, 5 and 6 carbons) represented by matrices. In the graph, the 10-bit orange vectors stand for the optimal enzymes according to the notation (6). When an enzyme is represented by a  $2 \times 10$  matrix it is because this enzyme transforms two functional groups in the corresponding biochemical reaction.



**Figure 11.** Performance graph. (Left) Evolution in glycolysis of the E1 enzyme that catalyzes the reaction Glucose  $\rightarrow$  Glucose-6-phosphate. (Right) Evolution in the Krebs cycle of the E12 enzyme responsible for the reaction  $\alpha$ -ketoglutarate  $\rightarrow$  SuccinilCoA.

#### 4. Discussion

From the simulation experiments conducted with the evolution of glycolysis, Krebs cycle and alcoholic fermentation networks, we can conclude that Darwinian evolution is able to optimize a biological network, such is the case of a metabolic network. However, in our study, the evolution of the enzymes involved in a metabolic network is not simulated by taking into account the stoichiometry of the reactions and the kinetic parameters of the enzymes. Although the latter is the usual protocol, it has not been the case in the present work. In our model, the evolution of the enzymes, and therefore of the edges of the graph representing the metabolic network under study, is simulated with the assumption that the active center of an enzyme can be modeled as an automaton. This approach is one of the main novelties of the present paper.

Now, what does graph theory contribute to our understanding of the evolution of metabolic networks in living beings? According to [3,29] and in a similar manner to many complex non-biological systems, metabolic networks exhibit characteristics that are typical of small-world networks. For example, they exhibit a power-law degree distribution, high cluster coefficients (a feature of robust networks being metabolic networks resilient to damage) and small network diameter (close distance between metabolites). These characteristics would be a consequence of the way in which biochemical reactions have been assembled with each other under the Darwinian mechanism of natural selection. Obviously, this assembly between reactions, i.e., the connection between the nodes (metabolites) of the graph (metabolic networks), is embodied by the evolution of enzymes. Thus, the evolution of the enzymes involved in the catalysis of the biochemical reactions (edges of the graph) transforms one metabolite (node) into the next. This point of view would be in agreement with the model [30] that describes the evolution of a metabolic network as a Markov process in which the insertion and deletion of reactions take place. The above approach is plausible since the survival of an organism depends on the characteristics exhibited by its metabolic networks. In this sense, the production of energy is crucial for a living organism. For example, the transformation of the most important sugar, glucose, into energy in the form of the ATP molecule was a critical step in the evolution of life on Earth. Thus, evolution allowed the optimization of the glycolytic pathway [31] by adopting the maximum production of ATP as the principle of optimization.

However, the above view about the evolution of metabolic networks contradicts the view of [6], observing that most important metabolic networks in an organism, i.e., glycolysis, Krebs cycle, and the pentose phosphate cycle, are distributed or split into different subsets or modules. These modules would represent the functional units of metabolism. In the case of the TCA or Krebs cycle, this modularity may be a consequence of the fact that TCA is the largest metabolic cycle found in organisms [4]. Consequently, and according to [32], the large network that forms the metabolism could be broken down into modules: the metabolism is organized around several core modules that interact strongly with each other, being responsible for the basic functions of metabolism. These central modules would interact in turn with other peripheral modules responsible for relatively independent metabolic functions. In other words, and according to [32], there would be a correlation between function, evolution, and topological modularity, which would suggest that the architecture of metabolic networks is a result of their evolutionary history.

Nevertheless, and with respect to the two previous approaches, small-world versus modularity, there are opinions [33] on the extent to which the properties of metabolism deduced by applying graph theory really reflect the biological organization present in living beings or, on the contrary, are merely an artifact resulting from the graph representation. Of course, methodologies other than graph theory can be applied. For example, glycolysis can be modeled and simulated with differential equations [34] or graph theory using Petri nets [35]. Alternatively, as we have undertaken in this work, we can use automata theory, e.g., electronic enzymes or, inspired by artificial neural networks, we could also use for this purpose the so-called 'biochemical neurons' [36]. However, no matter the method used,

the simulation of the evolution of a metabolic network is a complex problem that, in many cases, will lead to queries similar to [33].

In the field of biology, answering the question of how the evolution of enzymes involved in glycolysis and the Krebs cycle could have taken place is still a matter of debate. At present, there is still a debate between the retrograde model and the patchwork model [9,10], although there are classical papers that support the results of our work. In particular, ref. [36] argues that the glycolytic pathway appears to be the result of the chance assembly of independent evolved enzymes. Indeed all glycolytic enzymes are variations with a common motif: they all have a core of  $\beta$ -sheet protein chains surrounded by  $\alpha$ -helices chains. However, although the experimental data are important, perhaps the most satisfactory explanation comes from theoretical biology. During the 1990s, studies were carried out in which the evolution of a metabolic pathway was considered to be an optimization problem, as has been addressed in the present paper. For example, refs. [37,38] found in a model known as the ‘pentose game’, whose results apply to the evolution of the Krebs cycle, that the evolution of a metabolic pathway must satisfy certain requirements. These requirements were included under the common term of simplicity. The simplicity principle was summarized in a simple idea: the metabolite production occurs by the shortest route and, therefore, with the smallest possible number of reaction steps. Therefore, although the molecular biodiversity of the number of optimal enzymes we have calculated with (9) is so high; ultimately, living beings operate with only a few enzymes. This is because principles such as the simplicity principle we have just enunciated, which are a consequence of Darwinian natural selection, impose a strong constraint on the selection of a few enzymes.

The study of biological evolution is a problem that requires powerful mathematical and computational tools. In the particular case of the evolution of metabolic pathways, graph theory is useful for both studying the structures [39] present in organisms as well as for the analysis of the dynamic processes exhibited by living organisms [40]. Moreover, the application of graph theory together with heuristic methods such as genetic algorithms [41] is a promising approach to the study not only of metabolic networks but also complex networks. Complex networks are not easy to study because they behave as dynamic systems that evolve and change over time, e.g., social networks, the brain, the Internet, the immune system, electric power distribution networks, insect colonies, etc.

This paper is an example that illustrates a common problem in biology. In many biological studies, it is not enough to have powerful computers and sophisticated mathematical modeling tools, as is the case of graph theory. In many cases, the problem is the lack of solid working hypotheses, assumptions, and, therefore, of a proper experimental framework that would make it possible to search for satisfactory answers. Consequently, and to date, we still do not have a satisfactory ‘picture’ or global model to explain how the evolution of metabolism took place. However, despite these limitations, the study of the principles of the evolution of metabolic networks can be approached through simulation experiments, i.e., under a computational paradigm or framework [42]. Thus, experiments could be conducted *in silico* with a representation in a simplified form of the main components of biochemical reactions such as enzymes, fitness function, metabolites and even genes. The results of the simulation experiments [42] or those obtained by us in this paper would be congruent with the patchwork hypothesis (Figure 2), i.e., modern and highly specific enzymes would come from ancestral enzymes less specific in recognizing a substrate or metabolite.

Although our results may shed light on the evolution of metabolism, it is the methodology we introduce here that represents a novel contribution within the so-called computational paradigm. Nevertheless, this paradigm is not new, its theoretical foundation being based on the so-called ‘Artificial Chemistry’ [43] of the 1990s and on the earlier contributions of Kauffman [44]. Under an Artificial Life theoretical approach, the molecules are modeled as binary structures or strings, defining chemical reactions as ‘reaction rules’. These rules of reaction are deployed in different approaches, either by Boolean opera-

tions [44], lambda calculus [45], finite state machines, Turing machines, or even electronic hardware, as the one implemented for our enzyme [27,28].

A striking feature of the methodology introduced by artificial chemistry is that the results obtained from the simulations are similar although applying a different paradigm and methodology to those obtained with ordinary differential equations [25].

In conclusion, we suggest that the methodology introduced in the present work may encourage the search for new tools with which it will be possible to study and contrast through simulation experiments the hypotheses currently in vogue on the origin and evolution of metabolic networks.

## 5. Conclusions

In this paper, we study how the evolution of two of the most emblematic metabolic networks of living organisms, glycolysis and the Krebs cycle, could have taken place by Darwin's mechanism of natural selection. One of the novelties of the work has been to model enzymes as automata and metabolites as binary matrices, simulating the evolution of an enzyme with a simple genetic algorithm. The results obtained may contribute to shedding light on and, therefore, providing a plausible explanation to the origin of life and the evolution of metabolic networks. The general approach of the work can perhaps be generalized with appropriate changes to the study of the evolution of complex networks other than metabolic networks.

**Author Contributions:** C.D.L.M.M. has collaborated in the discussion, revision and carried out simulation experiments that were used in his Master's Degree in Industrial and Environmental Biotechnology 2016–2017, Complutense University of Madrid. R.L.-B. has devised the general problem and the model and written the Python routines of the artificial immune system and genetic algorithm. He has also supervised the work of the first author and written this paper. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Acknowledgments:** R.L.-B. also thanks the Department of Genetics, Physiology and Microbiology and the Departmental Section of Biochemistry and Molecular Biology for their collaboration through their Master's students with the Biomathematics Unit of the Faculty of Biology, Complutense University of Madrid.

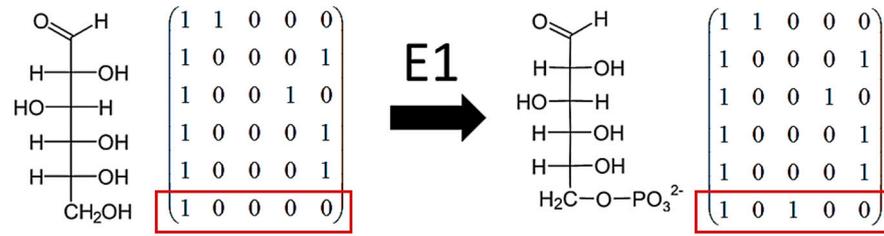
**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

In this section, we will illustrate with an example how to obtain the optimal configuration of the E1 enzyme responsible for catalyzing the biochemical reaction transforming Glucose into Glucose-6-phosphate (Figure A1). First, we obtain the two metabolites represented with their respective binary matrices. For this purpose, we apply Figure 5 by translating the functional groups of the molecules into their equivalent binary representation as a five-bit string. In accordance with Figure A1 we observe that the transformation of Glucose to Glucose-6-phosphate involves the chemical conversion of the CH<sub>2</sub>OH functional group of glucose to another CH<sub>2</sub>-O-PO<sub>3</sub><sup>2-</sup> group of glucose-6-phosphate. Secondly, when using the five-bit binary string representation of a functional group, we found that the goal is to perform the conversion 10000 → 10100. Thus, in the present example, the central bit of the binary string representing the glucose functional group must be inverted.

Once the Python code of the SGA [15] adapted to the experiments described in this article was downloaded, we encoded (Figure A2) the substrate *S* (Glucose) and the product *P* (Glucose-6-phosphate) of the reaction  $S + E \rightarrow P$ . Note that we only encode the functional groups that are part of the transformation of *S* to *P*. Additionally, note that the genome length is 10 bits, as in the experiment of example  $E_1$ : ( $c_{11}, c_{21}, c_{31}, c_{41}, c_{51}, o_{11}, o_{21}, o_{31}, o_{41}, o_{51}$ ).

Following, and as shown in Figure A1, we set the values of the recombination rate ( $p_r = 0.20$ ) as well as the mutation rate ( $p_m = 0.01$ ) and the rate of change of the bit value per chromosome ( $p_b = 0.05$ ).



**Figure A1.** What is the proper configuration of the E1 enzyme that converts Glucose (Left) to Glucose-6-phosphate (Right)? (see text for explanation).

Finally, and after running the program, we will choose those solutions (enzymes) with maximum fitness, i.e., with Hamming distance equal to zero. For example, in the current experiment, two optimal enzymes *E* are 1011011010 and 1010011010, since both convert the substrate, i.e., the functional group 10000 of *S*, into the functional group 10100 of the product *P*. Thus, in the case of enzyme 1011011010: 1 AND 1 = 1, 0 AND 0 = 0, 1 XOR 0 = 1, 1 AND 0 = 0 and 0 XOR 0 = 0 being 10100 the functional group of the product *P*. Note that by performing the appropriate operations, the second enzyme leads to the same result.

```

import math
import random
import numpy as np
import matplotlib.pyplot as plt

PopSize=100      # Population size
GenomeLength=10  # Genome length
generation_max= 200 # Iteration

def Init_population():
def Show_population():

#####

substrate[0]=1
substrate[1]=0
substrate[2]=0
substrate[3]=0
substrate[4]=0

product_e[0]=1
product_e[1]=0
product_e[2]=1
product_e[3]=0
product_e[4]=0

#####

#####
# SIMPLE GENETICALGORITHM
#
#####
def Simple_GA():
generation=0
print("-----GENERATION: ",generation,"-----")
print()
Init_population()
Show_population()
Fitness_evaluation(generation)

while (generation<generation_max):
print("The best of generation [",generation+1,"], best_chrom[generation])
print()
print("-----GENERATION: ",generation+1,"-----")
print()
# Select individuals, update generation number and obtain fitness
wheel_p_selection()
generation=generation+1
Fitness_evaluation(generation)

# Apply genetic operators
crossover(0.2)
mutation(0.01,0.05)
    
```

**Figure A2.** Python code for the simulation of the evolution of the metabolic pathway shown in Figure A1. (1) Experiment conditions, setting the population size, i.e., the number of enzymes to evolve, the length of the genome and the number of generations or iterations. (2) Setting the functional groups of the substrate *S* and product *P*. (3) Parameter values for recombination and mutation genetic operators.

## Appendix B

In this appendix we present the tables with the results of the experiments conducted in this work.

**Table A1.** Example of molecular biodiversity calculation. In the present experiment, we obtained  $n = 1,491,009$  enzymes and 54 isoenzymes, resulting in a  $M_B = 18,897,212$  bits (18.89 Mbits).

Enzyme	Total Solutions	Enzyme	Total Solutions
[1110011011]	48,282	[0110001011]	32,163
[1010011011]	70,791	[0111001011]	13,332
[1111011011]	25,516	[0110101011]	41,907
[1011011011]	33,188	[0010001011]	62,800
[1110111011]	39,949	[0011001011]	26,704
[1010111011]	38,660	[0011101011]	27,756
[1011111011]	30,986	[0010101011]	37,271
[1111111011]	20,669	[0111101011]	8880
[1110011010]	59,802	[0111001010]	2953
[1010011010]	50,292	[0011001010]	22,935
[1011011010]	53,968	[0010001010]	31,149
[1111011010]	51,644	[0110001010]	15,073
[1010011001]	64,083	[0110101001]	13,142
[1110111001]	20,859	[0010001001]	19,458
[1110011001]	30,405	[0010101001]	12,299
[1010111001]	41,591	[0110001001]	20,436
[1110011000]	23,481	[0110001000]	13,581
[1010011000]	32,714	[0010001000]	8765
[1011010011]	10,060	[0011100011]	12,204
[1011110011]	25,532	[0011000011]	23,857
[1010010011]	42,754	[0010000011]	52,353
[1010110011]	38,666	[0010100011]	10,710
[1010010010]	20,181	[0010000010]	44,475
[1011010010]	9416	[0011000010]	9944
[1010010001]	15,749	[0010000001]	5629
[1010110001]	10,357	[0010100001]	2981
[1010010000]	5432	[0010000000]	3225

**Table A2.** Glycolysis. Simulation results.

ENZYME	Functional Group Change	Coding	Different Solutions	Total Solutions	$M_B$ (bits)	Fitness
E1	CH <sub>2</sub> OH → CH <sub>2</sub> OP	10000 → 10100	54	1,491,009	18,897,212	98.6%
E2	CHO → CH <sub>2</sub> OH	11000 → 10000	108	1,438,457	21,049,471	91.8%
	CHOH → CO	10001 → 11011	12	1,418,899	17,243,935	99.0%
E3	CH <sub>2</sub> OH → CH <sub>2</sub> OP	10000 → 10100	54	1,479,813	19,219,669	99.2%
E4	CHOH → CHO	10001 → 11000	36	1,413,161	19,463,340	95.4%
E5	CHO → COOP	11000 → 11110	12	1,432,677	17,058,760	98.8%
E6	COOP → COOH	11110 → 11100	48	1,368,925	20,644,997	97.6%

**Table A2.** *Cont.*

ENZYME	Functional Group Change	Coding	Different Solutions	Total Solutions	$M_B$ (bits)	Fitness
E7	CHOH → CHOP	10001 → 10101	36	1,452,560	18,788,783	98.8%
	CH <sub>2</sub> OP → CH <sub>2</sub> OH	10100 → 10000	108	1,412,987	21,405,506	96.2%
E8	CHOP → COP	10101 → 10110	24	1,393,199	18,936,236	99.8%
	CH <sub>2</sub> OH → CH <sub>2</sub>	10000 → 01100	18	1,462,342	17,279,598	100%
E9	COP → CO	10110 → 11011	8	1,385,557	16,988,969	99%
	CH <sub>2</sub> → CH <sub>3</sub>	01100 → 01000	108	1,402,062	21,414,808	100%

**Table A3.** Krebs cycle. Simulation results.

ENZYME	Functional Group Change	Coding	Different Solutions	Total Solutions	$M_B$ (bits)	Fitness
E10	COH → CH	10011 → 01010	24	1,392,887	19,018,497	94.0%
	CH <sub>2</sub> → CHOH	01001 → 10010	12	1,387,318	17,779,168	95.8%
E11	CH → CH <sub>2</sub>	01010 → 01001	36	1,400,246	19,419,492	93.0%
	CHOH → CO	10010 → 11011	12	1,404,398	17,434,613	97.8%
E12	CO → CO <sub>2</sub> S	11011 → 11101	16	1,387,545	18,278,171	97.2%
E13	CO <sub>2</sub> S → CHOH	11101 → 11100	48	1,349,784	20,882,154	98.6%
E14	CH <sub>2</sub> → CH	01001 → 01101	36	1,405,740	19,077,282	95.6%
	CH <sub>2</sub> → CH	01001 → 01101	36	1,395,211	19,413,333	100%
E15	CH → CH <sub>2</sub>	01101 → 01001	72	1,335,197	21,537,852	91.8%
	CH → CHOH	01101 → 10010	8	1,366,984	17,320,683	98.85%
E16	CH <sub>2</sub> → CO	01001 → 11011	12	1,384,225	17,831,490	97.2%
	CHOH → CH <sub>2</sub>	10010 → 01001	12	1,411,206	7,335,462	93.2%

**Table A4.** Alcoholic fermentation. Simulation results.

ENZYME	Functional Group Change	Coding	Different Solutions	Total Solutions	$M_B$ (bits)	Fitness
E17	CHO → COH	11011 → 11000	48	1,342,367	20,973,898	98.8%
E18	COH → CH <sub>2</sub> OH	11000 → 10000	108	1,430,457	21,049,471	97.6%
E19	COH → COOH	11000 → 11100	36	1,452,560	18,788,783	98.8%

## References

- Pavlopoulos, G.A.; Secrier, M.; Moschopoulos, C.N.; Soldatos, T.G.; Kossida, S.; Aerts, J.; Schneider, R.; Bagos, P.G. Using graph theory to analyze biological networks. *BioData Min.* **2011**, *4*, 10. [\[CrossRef\]](#)
- Le Novère, N.; Hucka, M.; Mi, H.; Moodie, S.; Schreiber, F.; Sorokin, A.; Demir, E.; Wegner, K.; Aladjem, M.I.; Wimalaratne, S.M.; et al. The systems biology graphical notation. *Nat. Biotechnol.* **2009**, *27*, 735–741. [\[CrossRef\]](#) [\[PubMed\]](#)
- Wagner, A.; Fell, D.A. The small world inside large metabolic networks. *Proc. R. Soc. Lond. B* **2001**, *268*, 1803–1810. [\[CrossRef\]](#) [\[PubMed\]](#)
- Becker, M.Y.; Rojas, I. A graph layout algorithm for drawing metabolic pathways. *Bioinformatics* **2001**, *17*, 461–467. [\[CrossRef\]](#) [\[PubMed\]](#)
- Ravasz, E.; Somera, A.L.; Mongru, D.A.; Oltvai, Z.N.; Barabasi, A.L. Hierarchical organization of modularity in metabolic networks. *Science* **2002**, *297*, 1551–1555. [\[CrossRef\]](#) [\[PubMed\]](#)
- Ma, H.-W.; Zhao, X.-M.; Yuan, Y.-J.; Zeng, A.-P. Decomposition of metabolic network into functional modules based on the global connectivity structure of reaction graph. *Bioinformatics* **2004**, *20*, 1870–1876. [\[CrossRef\]](#)
- Heinrich, R.; Schuster, S. *The Regulation of Cellular Systems*; Chapman & Hall: New York, NY, USA, 1996.
- Wang, L.; Dash, S.; Ng, C.Y.; Maranas, C.D. A review of computational tools for design and reconstruction of metabolic pathways. *Synth. Syst. Biotechnol.* **2017**, *2*, 243–252. [\[CrossRef\]](#)
- Lacroix, V.; Cottret, L.; Thebault, P.; Sagot, M.-F. An introduction to metabolic networks and their structural analysis. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2008**, *5*, 594–617. [\[CrossRef\]](#) [\[PubMed\]](#)

10. Emiliani, G.; Fondi, M.; Liò, P.; Fani, R. Chapter 2. Evolution of metabolic pathways and evolution of genomes. In *Geomicrobiology: Molecular and Environmental Perspective*; Barton, L., Mandl, M., Loy, A., Eds.; Springer: Dordrecht, The Netherlands, 2010; pp. 37–68.
11. Lahoz-Beltra, R. *Bioinformática: Simulación, Vida Artificial e Inteligencia Artificial*; Ediciones Díaz de Santos: Madrid, Spain, 2004; pp. 237–323.
12. Holland, J.H. *Adaptation in Natural and Artificial Systems*, 1st ed.; University of Michigan Press: Ann Arbor, MI, USA, 1975.
13. Goldberg, D.E. *Genetic Algorithms in Search, Optimization, and Machine Learning*; Addison-Wesley: Reading, MA, USA, 1989.
14. Davis, L. (Ed.) *Handbook of Genetic Algorithms*, 1st ed.; Van Nostrand Reinhold: New York, NY, USA, 1991.
15. Lahoz-Beltra, R.; De Las Morenas Mateos, C. Simulating the evolution of a metabolic network in Python. *Figshare. J. Contrib.* **2023**. [[CrossRef](#)]
16. Recio Rincon, C.; Cordero, P.; Castellanos, J.; Lahoz-Beltra, R. A new method for the binary encoding and hardware implementation of metabolic pathways. *Int. J. Inf. Theor. Appl.* **2014**, *21*, 21–29.
17. Randic, M. On the recognition of identical graphs representing molecular topology. *J. Chem. Phys.* **1974**, *60*, 3920–3927. [[CrossRef](#)]
18. Kier, L.B.; Cheng, C.-K.; Testa, B.; Carrupt, P.-A. A cellular automata model of enzyme kinetics. *J. Mol. Graph.* **1996**, *14*, 227–231. [[CrossRef](#)] [[PubMed](#)]
19. Sayama, H. Definition of Cellular Automata. Available online: <https://math.libretexts.org/@go/page/7829> (accessed on 1 May 2023).
20. Marashi, S.-A.; Behrouzi, R. Modeling direct ligand passage toward enzyme active site by a ‘double cellular automata’ model. *Biochem. Biophys. Res. Commun.* **2005**, *333*, 1–4. [[CrossRef](#)]
21. Eibinger, M.; Zahel, T.; Ganner, T.; Plank, H.; Nidetzky, B. Cellular automata modeling depicts degradation of cellulosic material by a cellulase system with single-molecule resolution. *Biotechnol. Biofuels* **2016**, *9*, 56. [[CrossRef](#)]
22. Kier, L.B.; Cheng, C.-K. A cellular automata model of an anticipatory system. *J. Mol. Graph. Model.* **2000**, *18*, 29–32. [[CrossRef](#)]
23. Kier, L.B.; Bonchev, D.; Buck, G. Modeling biochemical networks: A cellular-automata approach. *Chem. Biodivers.* **2005**, *2*, 233–243. [[CrossRef](#)]
24. Weimar, J.R. Cellular automata approaches to enzymatic reaction networks. In Proceedings of the Cellular Automata: 5th International Conference on Cellular Automata for Research and Industry, ACRI 2002, Geneva, Switzerland, 9–11 October 2002; Springer: Berlin/Heidelberg, Germany, 2002; pp. 294–303.
25. Lahoz-Beltra, R. Molecular automata modeling in structural biology. In *Advances in Structural Biology*; Malhotra, S.K., Tuszyński, J.A., Eds.; Elsevier: Amsterdam, The Netherlands, 1999; Volume 5, pp. 85–101. [[CrossRef](#)]
26. Lahoz-Beltra, R. Molecular automata assembly: Principles and simulation of bacterial membrane construction. *BioSystems* **1997**, *44*, 209–229. [[CrossRef](#)] [[PubMed](#)]
27. Lahoz-Beltra, R. Evolving hardware as model of enzyme evolution. *BioSystems* **2001**, *61*, 15–25. [[CrossRef](#)]
28. Lahoz-Beltra, R. Enzima Electronico, Con Sitio Activo Reconfigurable. P200000174, 27 January 2000. Oficina Española de Patentes y Marcas. Available online: <http://invenes.oepm.es/InvenesWeb/detalle?referencia=P200000174> (accessed on 1 May 2023).
29. Fell, D.A.; Wagner, A. The small world of metabolism. *Nat. Biotechnol.* **2000**, *18*, 1121–1122. [[CrossRef](#)]
30. Mithani, A.; Preston, G.M.; Hein, J. A stochastic model for the evolution of metabolic networks with neighbor dependence. *Bioinformatics* **2009**, *25*, 1528–1535. [[CrossRef](#)] [[PubMed](#)]
31. Heinrich, R.; Montero, F.; Klipp, E.; Waddell, T.G.; Melendez-Hevia, E. Theoretical approaches to the evolutionary optimization of glycolysis: Thermodynamic and kinetic constraints. *Eur. J. Biochem.* **1997**, *243*, 191–201. [[CrossRef](#)]
32. Zhao, J.; Ding, G.H.; Tao, L.; Yu, H.; Yu, Z.H.; Luo, J.H.; Cao, Z.W.; Li, Y.X. Modular co-evolution of metabolic networks. *BMC Bioinform.* **2007**, *8*, 311. [[CrossRef](#)]
33. Zhou, W.; Nakhleh, L. Properties of metabolic graphs: Biological organization or representation artifacts? *BMC Bioinform.* **2011**, *12*, 132. [[CrossRef](#)] [[PubMed](#)]
34. Van Zwieten, D.A.J.; Rooda, J.E.; Armbruster, H.D.; Nagy, J.D. *Discrete Event Simulations for Glycolysis Pathway and Energy Balance. SE Report*; Technische Universiteit Eindhoven: Eindhoven, The Netherlands; Volume 2010-02, Available online: <https://pure.tue.nl/ws/portalfiles/portal/3865367/376484356450067.pdf> (accessed on 7 May 2023).
35. Oyelade, J.; Isewon, I.; Rotimi, S.; Okunoren, I. Modeling of the glycolysis pathway in *Plasmodium falciparum* using Petri nets. *Bioinform. Biol. Insights* **2016**, *10*, 49–57. [[CrossRef](#)] [[PubMed](#)]
36. Fothergill-Gilmore, L.A. The evolution of the glycolytic pathway. *Trends Biochem. Sci.* **1986**, *11*, 47–51. [[CrossRef](#)]
37. Meléndez-Hevia, E.; Waddell, T.G.; Montero, F. Optimization of metabolism: The evolution of metabolic pathways toward simplicity through the game of the pentose phosphate cycle. *J. Theor. Biol.* **1994**, *166*, 201–220. [[CrossRef](#)]
38. Meléndez-Hevia, E.; Waddell, T.G.; Cascante, M. The puzzle of the Krebs citric acid cycle: Assembling the pieces of chemically feasible reactions, and opportunism in the design of metabolic pathways during evolution. *J. Mol. Evol.* **1996**, *43*, 293–303. [[CrossRef](#)]
39. Diudea, M.V.; Gutman, I.; Jantschi, L. *Molecular Topology*; Nova Science Publishers: Huntington, NY, USA, 2001.
40. Guzzi, P.H.; Roy, S. *Biological Network Analysis: Trends, Approaches, Graph Theory, and Algorithms*; Elsevier: Amsterdam, The Netherlands, 2020.
41. Jäntschi, L. Genetic Algorithms and Their Applications. Ph.D. Thesis, University of Agricultural Sciences and Veterinary Medicine Cluj-Napoca, Cluj, Romania, 2010.

42. Flamm, C.; Ullrich, A.; Ekker, H.; Raden, M.; Hoegerl, D.; Rohrschneider, M.; Sauer, S.; Scheuermann, G.; Klemm, K.; Hofacker, I.; et al. Evolution of metabolic networks: A computational frame-work. *J. Syst. Chem.* **2010**, *1*, 4. [[CrossRef](#)]
43. Dittrich, P.; Ziegler, J.; Banzhaf, W. Artificial chemistries—A review. *Artif. Life* **2001**, *7*, 225–275. [[CrossRef](#)]
44. Kauffman, S.A. Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol.* **1969**, *22*, 437–467. [[CrossRef](#)]
45. Fontana, W.; Buss, L. The arrival of the fittest: Toward a theory of biological organization. *Bull. Math. Biol.* **1994**, *56*, 1–64. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.