

## Desenvolvimento de matrizes de horário acadêmico utilizando algoritmos genéticos

Anderson Fernandes Rosa Filho<sup>1</sup>

Jaqueline Brigladori Pugliesi<sup>2</sup>

### Resumo

Em contato com profissionais da área acadêmica foi coletado que atualmente, o desenvolvimento de matrizes de horários tem sido uma tarefa árdua, pois necessita da participação de vários profissionais, ocupando tempo e esforço mental de forma demasiada. A matriz deve contemplar as mais variadas restrições, dentre elas, a título de exemplos, podemos citar: cada curso possui períodos distintos onde ao final devem contar de forma exata com seus semestres e suas disciplinas, cada disciplina possui seus professores e estes possuem suas disponibilidades, ou seja, sempre foi extremamente complexo organizar o horário acadêmico. As variáveis de restrições pessoais e o grande número de combinações possíveis sempre foi um fator limitante para encontrar um resultado satisfatório. Atividades como estas são nomeadas de problemas de otimização combinatória. Foi então que surgiu a iniciativa de desenvolver um aplicativo que, por meio da Inteligência Artificial, que manipula tabelas em um Banco de Dados, tratando as disponibilidades de cada professor, levando em consideração as regras acadêmicas, assim gerando resultados, para realização destes procedimentos utiliza-se uma subárea do conteúdo de IA, o método de Algoritmo Genético. Este procedimento proposto teve como inspiração na ideia de seleção e evolução das espécies, desenvolvida por Darwin, os principais elementos tratados são os cromossomos de um indivíduo, pois, eles são nossos resultados, ou seja, nossa matriz de horários, os termos citados serão explicados no decorrer do trabalho. O presente trabalho tem como objetivo desenvolver uma aplicação capaz de gerar matrizes de horários acadêmicos de forma independente.

**Palavras-chave:** Algoritmo Genético. Inteligência Artificial. Matriz de horário.

### Abstract

*In contact with professionals in the academic field, it was gathered that the development of timetables has been an arduous task since it requires the participation of several professionals, spending too much time and mental effort. The matrix should include the most varied restrictions, such as each course has distinct periods where, in the end, they must accurately count with their semesters and their subjects, each discipline has its teachers and they have their resources, that is, it has always been extremely complex to organize the academic schedule. The personal constraint variables and the large number of possible combinations have always been a limiting factor in finding a satisfactory result. Activities like these are named combinatorial optimization problems. Thus, the initiative came up to develop an*

<sup>1</sup> Graduando em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: andersonfilho@gmail.com.

<sup>2</sup> Doutora em Ciências da Computação pela USP – São Carlos/SP. Docente na Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: jbpugliesi@gmail.com.

*application through Artificial Intelligence, which can manipulate tables in a Database, dealing with the availability of each teacher, taking into account the academic rules. Therefore, generating results to perform these procedures, a sub-area of the AI content, the Genetic Algorithm method has been used. This proposed procedure was inspired by the idea of selection and evolution of the species, developed by Darwin who talked about the main elements treated which are the chromosomes of an individual. For that reason, they are our results, that is to say, our matrix of schedules, which the mentioned terms will be explained in this article. This paper aims at developing an application capable of generating matrices of academic schedule independently.*

**Keywords:** *Genetic Algorithm. Artificial intelligence. Matrix of Schedule.*

## 1 Introdução

Na atual situação onde existe a movimentação de milhares de dados via internet, afetando a todos quando se utilizam e-mails, visualiza-se vídeos, músicas, imagens, acessa-se mídias digitais e principalmente quando se efetua compras na web ou até mesmo no teor das palavras digitadas em um bate-papo, são armazenadas informações. Sendo estas informações manipuladas para vários objetivos, o mais comum é para o setor de marketing, que imediatamente ao coletar uma informação digitada, dispara ofertas de produtos e serviços relacionados ao tema pesquisado ou mesmo digitados em conversas. Estes algoritmos, na maioria das vezes, utilizam subáreas da Inteligência Artificial (IA). Estas subáreas podem ser divididas em três abordagens: simbólica, conexionista e evolucionária.

A princípio a IA simbólica se relaciona com a forma que o ser humano raciocina sendo popularizada com o surgimento dos Sistemas Especialistas e da linguagem Prolog.

Já a conexionista se inspira no comportamento de nossos neurônios, como exemplo de sua utilização, pode-se citar as Redes Neurais.

Tem-se ainda a evolucionária que se inspira na evolução e seleção natural de Darwin, a qual será abordada e tratada neste trabalho.

Evidente o grande benefício que a maioria das instituições acadêmicas terá a cada ciclo de início do semestre letivo, utilizando-se das subáreas da IA para auxiliar no “problema” da elaboração dos horários acadêmicos. A organização e a montagem da matriz de horário tem sido uma tarefa árdua que demanda uma grande quantidade de tempo e esforço mental dos profissionais que a desenvolvem, e pensando neste problema, surgem alguns questionamentos, dentre eles, por que

não tentar desenvolver uma aplicação que possa automatizar esta montagem? O problema agora seria como? E o que utilizar? É possível o desenvolvimento de um programa que gere todas as possibilidades existentes para esse problema? Se não todas, mas algumas que resolvam este problema? Seria viável caso houvesse uma grande quantidade de respostas? Ou, se teríamos tempo para testar todas essas respostas? Essas são algumas questões que iremos tentar responder neste artigo.

Inicialmente, na montagem de matriz de horários, há vários fatores que devem ser determinados, atualmente cada instituição escolhe um responsável e este estabelece qual será a melhor forma para o seu desenvolvimento, muitas vezes são baseadas em matrizes anteriores ou dando preferência para disponibilidades dos docentes, contudo, este processo feito manualmente é bastante demorado, além de proporcionar várias tentativas e geralmente poucos acertos.

Dando continuidade, nestes processos podem-se obter várias soluções por caminhos diferentes, para este trabalho vamos utilizar o método inspirado no darwinismo, na seleção e na evolução das espécies.

Misturando biologia com computação seria possível automatizar e conseguir resolver problemas de matrizes de horários acadêmicos utilizando-se de Inteligência Artificial? A resposta para essa questão é SIM, pois, analisando como a evolução das espécies funciona e modelando este método para um algoritmo computacional, seria possível verificar sua viabilidade e se suas respostas trariam soluções adequadas para serem implantadas, mas por onde começar?

Diante desta questão, a princípio será criada uma população inicial, com um número pré-estabelecido de indivíduos, estes serão selecionados pela sua aptidão de sobrevivência em um ambiente, possivelmente os mais aptos irão sobreviver e se reproduzir e teremos algumas alterações em seus cromossomos, esta nova geração irá novamente se reproduzir e gerar outras gerações e assim por diante, e é assim que aplicaremos este método em nosso “problema” até que alguma geração nos dê uma boa solução dentre várias que ele trará disponível conforme nossas restrições.

O método utilizado para realização deste trabalho é conhecido tecnicamente por alguns como busca estocástica, ou seja, um conjunto formando um domínio, que por sua vez se relaciona com outro conjunto denominado contradomínio, sendo geradas imagens desta relação, tudo por meio dos elementos aplicados a uma função, onde as imagens serão as respostas obtidas, já que que o aplicativo busca

de forma incessante melhorar suas próprias respostas, sempre conferindo a avaliação das notas de cada indivíduo.

Vale lembrar, que o artigo trará a relação dos materiais e métodos de desenvolvimento que serão utilizados, como parte de algoritmos, Banco de Dados, linguagem utilizada, e alguns pseudocódigos dos dados e suas modelagens para o problema.

Ainda serão apresentados seus resultados e discussões, tempo de execução e o que pode ser trabalhado em projetos futuros. Assim espera-se que o leitor consiga ter uma imagem clara da aplicação do método ao “problema” e de como ele pode ser utilizado para tantas outras questões semelhantes.

Vale ressaltar, que após a confecção deste estudo e reflexão ampla sobre o conceito, surgiu uma frase que poderia definir o programa de Algoritmo Genético (AG) e caso este tenha objetivos claros e consiga realizar o que foi proposto em suas resoluções, então, poderia ser definido pela frase, “Faça o que eu mando, obtenha o que eu preciso”.

## 2 Algoritmos Genéticos

Em resumo Russel e Norvig (2013, p. 194) definem Algoritmo Genético como “uma busca de subida de encosta estocástica em que é mantida uma grande população de estados. Novos estados são gerados por mutação e por cruzamento, que combinam pares de estados da população.” Em outras palavras, pode-se expressar que este tipo de busca se inicia por soluções aleatórias e então são realizadas modificações nestas para tentar-se melhorar os resultados e até mesmo poder solucionar o problema.

Para deixar ainda mais clara a ideia do autor sobre a denominação de Algoritmo Genético, segue nova citação por ele publicada.

Um Algoritmo Genético é uma variante de busca em feixe estocástica na qual os estados sucessores são gerados pela combinação de dois estados pais, em vez de serem gerados pela modificação de um único estado. A analogia em relação à seleção natural é a mesma que se dá na busca em feixe estocástica, exceto pelo fato de agora estarmos lidando com a reprodução sexuada, e não com a reprodução assexuada (RUSSEL e NORVIG, 2013, p. 166).

O presente estudo poderia ser considerado, como uma técnica computacional para problemas de otimização e busca, e se baseia na forma de como os seres vivos sobrevivem, no princípio de seleção natural proposto por Charles Darwin.

Complementando a ideia de Algoritmos Genéticos, Ikeda (2010) ensina em seu trabalho, que estes algoritmos se baseiam na evolução das espécies que evoluem para alcançar a solução de um problema, onde a competitividade da natureza impõe uma evolução constante para melhor chance de sobrevivência, em uma evolução constante. Vale ainda ressaltar, que este processo de evolução é aleatório e são guiados por uma seleção que se baseia nas estruturas individuais de cada indivíduo.

A princípio o método de seleção natural desenvolvido por Charles Darwin, em sua obra *A Origem das espécies* (DARWIN, 2003), explica que as espécies mais aptas sobrevivem em seu ambiente e tendem a se reproduzir, gerando filhos que herdaram aspectos específicos de seus pais, com uma certa mutação (ou não) em sua geração. Ou seja, quanto melhor o indivíduo for capaz de se adaptar em seu ambiente, logo este terá melhor chance de sobreviver e se reproduzir, segue outro exemplo extraído da doutrina de Darwin.

Suponhamos um lobo que se nutrisse de diferentes animais, apoderando-se de uns pela astúcia, de outros pela força, de outros, enfim, pela agilidade. Suponhamos ainda que a sua presa mais rápida, o gamo por exemplo, tinha aumentado em número após algumas alterações sobrevindas no país, ou que os outros animais, de que se nutre ordinariamente, tinham diminuído durante a estação do ano em que o lobo está mais apertado pela fome. Nestas circunstâncias, os lobos mais ágeis e mais velozes têm mais probabilidades de sobreviver do que os outros (DARWIN, 2003, p. 104).

Acresce que os Algoritmos Genéticos trabalham de maneira similar a teoria de Darwin da seleção natural, mas, de onde surgiu este método? Quem foi o originário deste algoritmo?

## 2.1 Breve histórico

Foi publicado em 1975, por Jhon Holland, em *Adaptation in Natural and Artificial Systems*, sendo essencial no estudo de conduzir os processos naturais adaptáveis para serem modelados para os sistemas de computadores. Outra importante obra sobre o método é a de David Goldberg, 1989, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Graças a estes e outros autores que

publicaram trabalhos na área, o método de Algoritmo Genético pode ser aplicado a vários problemas de otimização combinatória.

Além disso, segundo Souza (2019, online), os problemas de otimização combinatória são:

Esse nome meio "assustador" esconde um conceito bastante simples. Suponha que você possui um conjunto de itens e uma série de regras que podem ser usadas para selecionar alguns elementos (itens) desse conjunto. Usando essas regras, há várias maneiras diferentes de escolher os elementos e criar outros conjuntos menores (ou subconjuntos). Se a cada elemento estiver associado um custo, os subconjuntos criados também terão um custo que é dado, por exemplo, pela soma dos custos de seus elementos. O problema de Otimização Combinatória, em geral, se resume a encontrar, dentre todos os possíveis subconjuntos, aquele cujo custo seja o menor possível.

Mas como aplicar isto a um algoritmo computacional? Ainda mais em um problema? Deve-se dar início a prática dos métodos de Algoritmos Genéticos como códigos que realizam regras de busca e otimização. Sobre estas técnicas, Lacerda e Carvalho (1999) apresentam em comum um espaço de busca, onde estão todas as possíveis soluções para um problema, porém, estas são geradas aleatoriamente, bem como uma função objetivo, conhecida também como função de avaliação, ela avalia as soluções produzidas definindo uma nota a solução, sendo possível verificar se a solução é boa ou ruim.

## 2.2 Exemplo prático

Um exemplo simples para iniciar a prática ao método seria encontrar o máximo da função contido na Figura 1, mostrando o ponto da função que seria considerada

$$f(x) = x^2$$

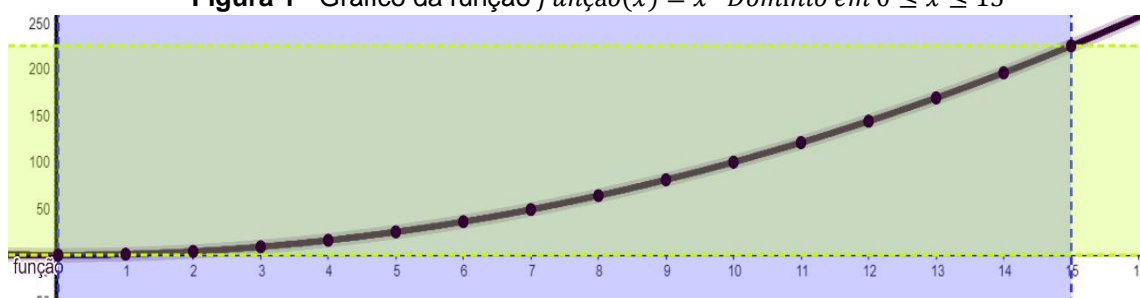
Em que o domínio da função deve estar entre:

$$0 \leq x \leq 15$$

Ou matematicamente expressando:

$$\mathbb{D}f(x) = \{x \in \mathbb{R} | 0 \leq x \leq 15\}.$$

**Figura 1** - Gráfico da função  $f(x) = x^2$  Domínio em  $0 \leq x \leq 15$



Fonte: Autor.

Do mesmo modo, este método se baseia na evolução natural das espécies e com isto foram utilizadas nomenclaturas características deste conceito. Segundo Silveira e Amabis (2004) e Estranho (2011), os organismos dos seres vivos são constituídos por genes e de suas sequências são formadas os DNA, sendo estes um conjunto de moléculas que carregam as informações genéticas de cada ser vivo, possuindo característica de um fio com 2 metros de comprimento que ficam supercondensados, enrolados como molas dentro do cromossomo. Exercendo os três (gene, DNA, cromossomos) o mesmo objetivo, que é levar suas características moldando cada espécie. Os cromossomos podem sofrer mutações quando há cruzamento das espécies.

Acrescenta-se ainda que, quando se utiliza este conceito na computação para aplicar a problemas, usa-se termos para trabalhar com certos elementos destes, conforme são mostrados na Tabela 1.

**Tabela 1** - Termos utilizados em Algoritmos Genéticos.

Termo utilizado	Descrição
Indivíduo	Qualquer solução possível
Cromossomo	Codificação das características de um indivíduo
População	Grupo de todos os indivíduos
Espaço de busca	Todas soluções possíveis para o problema
Genoma	Coleção de todos os cromossomos de um indivíduo

Fonte: Baseado em IKEDA, 2010.



### 2.3 Exemplo de representação e indivíduo

O cromossomo, em Algoritmo Genético, é uma possível solução para o problema. Ele pode ter várias representações, sendo que as mais comuns são, em listas que podem ser binária, string (alfabeto incluindo números), valores reais ou inteiros. No exemplo, é utilizado o binário e representado na Figura 2.

Figura 2 - Exemplo de cromossomo.



representação do número 10 dez em binário  
na lista cromossomo cromossomo = [1,0,1,0]

Fonte: Autor.

Desde já, alerta-se que para a criação dos cromossomos, deve-se iniciar a população e depois avaliá-la. A população irá gerar vários indivíduos aleatórios com seus devidos cromossomos. Em nosso exemplo, o cromossomo, que é um vetor ou lista que representa o indivíduo, também pode ser desmembrado e recombinado com partes de cromossomos de outro indivíduo, como herda a ideia da evolução das espécies.

Esta recombinação, ou melhor, cruzamento de indivíduos, que são selecionados através de suas avaliações, definidas por uma função comum, denominada “função de avaliação”, recebendo sua própria nota. Em alguns trabalhos, como o de (IKEDA, 2010), essa função é chamada de “função objetivo”. Neste trabalho, será chamada de “função de avaliação”, que define o valor de cada indivíduo, sua avaliação poderá determinar se ele é uma boa adaptação em seu ambiente ou não, observando sua proximidade da solução do problema, definindo sua probabilidade de ser selecionado.

### 2.4 Exemplo da função de avaliação

No exemplo, apresentado na Figura 3, vamos supor que o indivíduo e seu cromossomo é o representado na imagem e a função de avaliação funcionará atribuindo 1 ponto para o indivíduo para cada número 1 que estiver na lista ou



cromossomo. Portanto, o cromossomo possui dois números 1, e a nota da avaliação deste indivíduo será 2.

**Figura 3** - Exemplo da função avaliação.



pela a função de avaliação definida será  
Nota da avaliação do cromossomo: 2

Fonte: Autor.

Assim, prosseguindo com o processo de seleção, serão selecionados os pais para realizar o cruzamento do cromossomo. Os pais selecionados tendem a ter melhores notas devido ao método de seleção. O método mais utilizado é o da roleta. Para fins didáticos e facilitação da leitura de como utilizar a roleta, será exibido um gráfico, com a frequência acumulada dos valores das notas. Dessa forma, a Figura 4 exemplifica o tratamento, e a Tabela 2 e a Figura 5 apresentam o processo estatístico.

**Figura 4** - Cromossomos / Notas.

Indivíduo	Cromossomos	Nota
I	1 0 1 0	2
II	0 0 1 0	1
III	1 0 1 1	3
IV	1 1 0 0	2

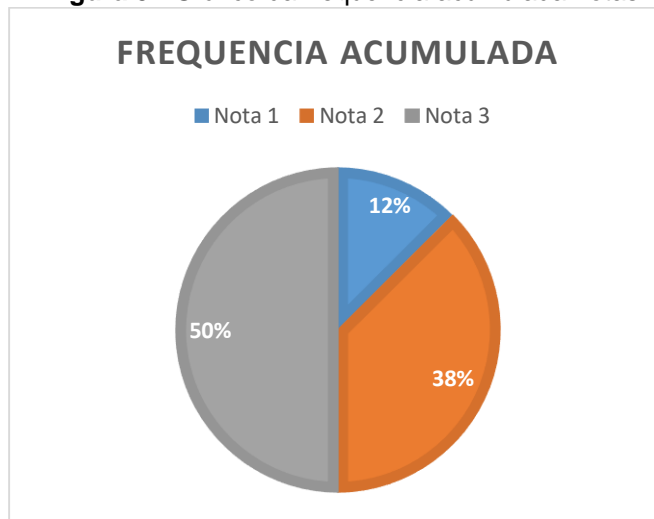
Fonte: Autor.

**Tabela 2** - Notas dos pais

Valor Nota	Frequências
Nota 1	1
Nota 2	2
Nota 3	1

Fonte: Autor.

Figura 5 - Gráfico da frequência acumulada notas



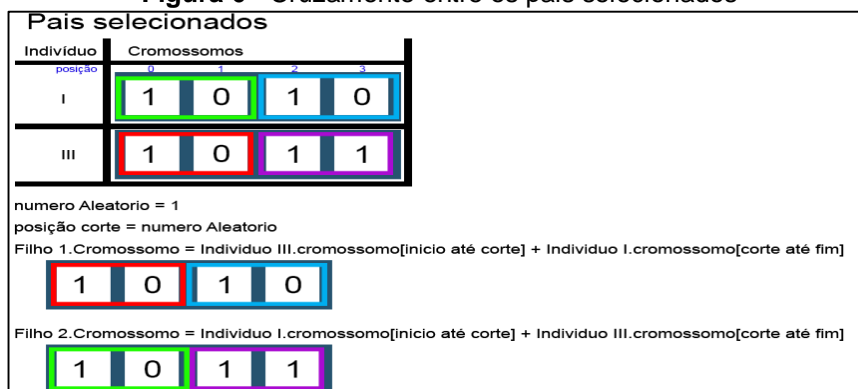
Fonte: Autor.

Ainda, para ter uma boa solução, utiliza-se a frequência acumulada, já que a roleta terá uma maior probabilidade de escolher um pai com uma avaliação melhor.

### 2.5 Exemplo da função *crossover*

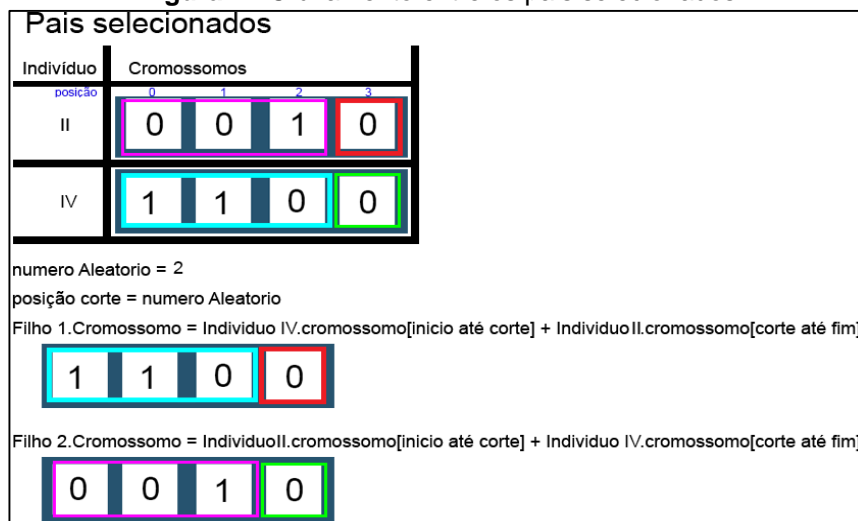
Após a seleção dos pais, inicia-se o *crossover*. Este processo trata do cruzamento dos pais selecionados pela roleta, gerando novos filhos. Para realizar este método, devem-se decidir quais elementos serão utilizados, ou seja, quais genes dos cromossomos serão usados, sendo necessário ser sorteado um corte. As Figuras 6 e 7 apresentam casais escolhidos, na roleta, para reprodução, e diferentes pontos de corte.

Figura 6 - Cruzamento entre os pais selecionados



Fonte: Autor.

Figura 7 - Cruzamento entre os pais selecionados



Fonte: Autor.

Logo após os pais serem selecionados, será sorteado um ponto de corte e os filhos herdam as características dos pais. Chamam-se esses cruzamentos de *crossover aritmético*, que é o *crossover* onde há um número de corte, representando a parte que cada filho irá herdar do cromossomo dos pais. Também existem outros tipos de funções de cruzamentos como aponta Ikeda (2010, p. 4):

*Crossover* em 1 ou mais pontos

Para que tal cruzamento seja possível, 2 fatores devem ser analisados:

- saber se a combinação será efetivamente realizada através da taxa de *crossover*
- determinar em qual(is) ponto(s) o *crossover* será feito Por exemplo, se a taxa for de 0.7 (70%), considerando uma escala entre 0 e 1, e o valor sorteado for maior, então o *crossover* não é realizado, ou seja, os filhos serão idênticos aos pais. Caso contrário, é feito o corte de cruzamento em ponto(s) determinado(s) pelo algoritmo, criando indivíduos com a combinação dos genes dos pais. No exemplo abaixo, o *crossover* ocorre em apenas 1 ponto, resultando em filhos.

*Crossover* uniforme

Cada bit do descendente é escolhido de acordo com uma máscara de cruzamento gerada aleatoriamente. Onde houver 1 na máscara, o bit virá do primeiro pai, caso contrário, virá do segundo. O processo é repetido com os pais trocados para produzir o segundo filho. Para cada cruzamento, uma nova máscara é criada.

## 2.6 Exemplo de uma função mutação

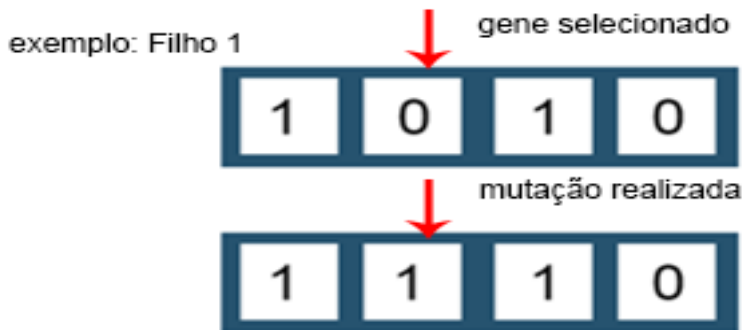
Posteriormente ao cruzamento tem-se a função de mutação. Esta é uma das funções mais básicas do algoritmo, que terá uma taxa de mutação como parâmetro, e percorrerá o cromossomo passando de gene em gene do indivíduo, realizando testes para verificar essa taxa de mutação. Se esta taxa for menor que o valor

sorteado, então ele irá alterar aquele valor para uma string oposta ou, em alguns casos, para valores aleatórios dentro do domínio. No exemplo, se a posição sorteada do cromossomo for “zero”, o número será alterado para “um”, e caso o número seja “um”, então ele mudará para do número “zero”. Se o número sorteado não ultrapassar o da taxa de mutação, então prosseguirá sem alterações.

Figura 8 - Função de mutação

### Mutação

verifica na posição ou “gene”  
se número aleatório maior que a taxa de mutação então  
troque o valor do “gene”

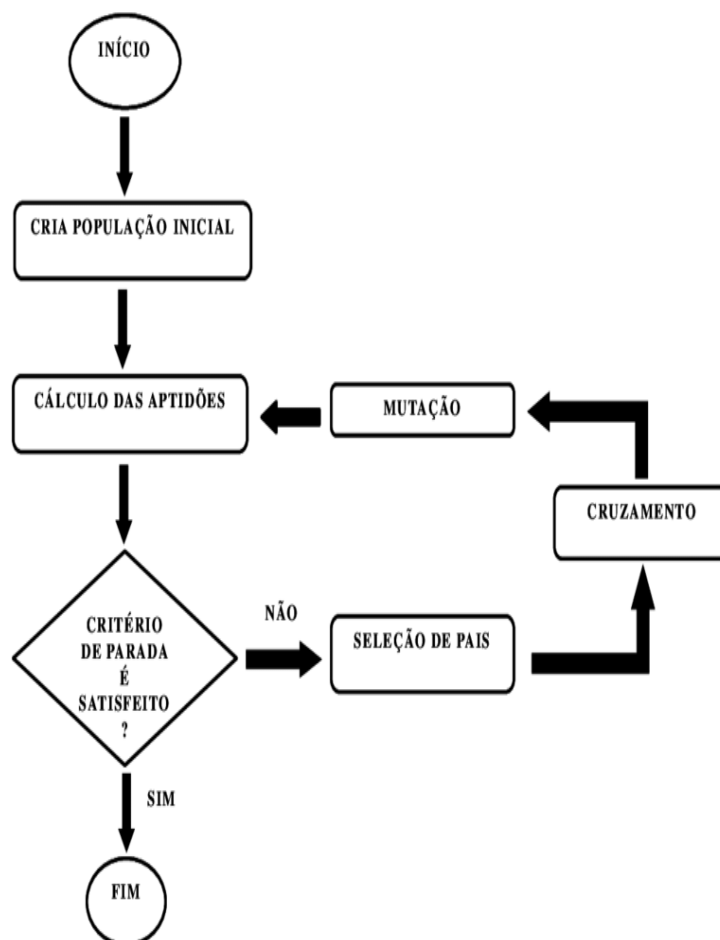


Fonte: Autor.

## 2.7 Geração de uma nova população

Após a criação da população inicial, deve-se começar o processo de nova geração, e após todos os passos serem realizados teremos uma nova população, que servirá de base para a reiteração do algoritmo, consecutivamente buscando em cada geração uma boa solução para o problema, assim como na evolução dos seres, com a estratégia de “gerar-e-testar” (IKEDA, 2010). A Figura 9 apresenta um digrama para melhor representar o fluxo das operações.

Figura 9 - Fluxograma de Algoritmo Genético



Fonte: NRIA, *et al.*, 2013, online.

Assim, mesmo que o exemplo utilizado seja simples, fez-se necessário sua simulação para exemplificar a ideia do funcionamento do Algoritmo Genético. Não obstante, este algoritmo pode ser aplicado em problemas muito complexos, como já citado neste trabalho.

### 3 Desenvolvimento

A princípio, o desenvolvimento de matrizes de horários acadêmicos é um problema complexo de várias soluções possíveis. Este tipo de problema demanda tempo e certo conhecimento específico sobre o tema para se ter uma possível solução, e nem sempre será possível se chegar em uma boa resposta. Com o objetivo de oferecer aos responsáveis pelo desenvolvimento destas matrizes uma maior agilidade e variadas possibilidades de solução, pontuando suas avaliações, é necessário aplicar uma técnica para sua resolução, sendo esta denominada de

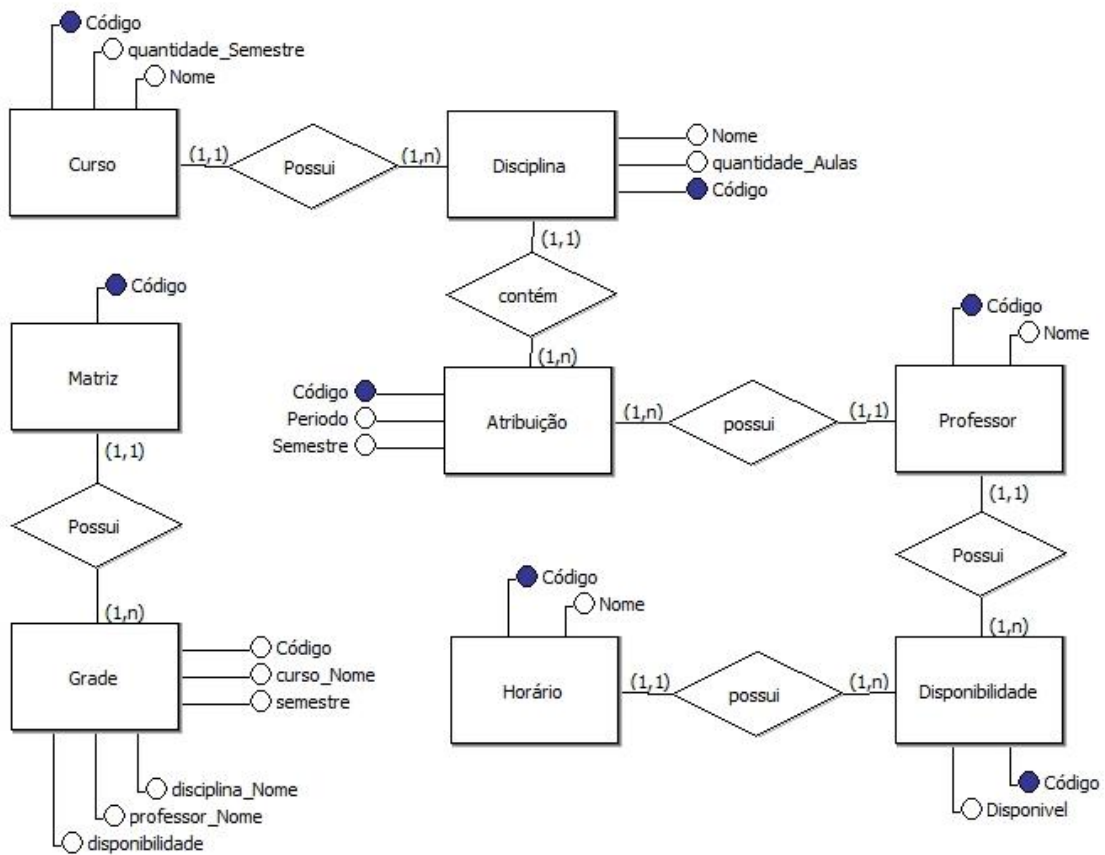
Algoritmo Genético, que será aplicado neste trabalho para auxiliar a montagem destas matrizes.

Nesta seção, explica-se como foi o desenvolvimento de uma geração de grades de horários, aplicada à FATEC Franca – Thomaz Novelino. Além disso, relacionam-se os passos de montagem do algoritmo com o problema apresentado.

Em primeiro lugar, definem-se os dados, quais são as informações e como elas devem ser manipuladas. Na instituição citada, a título de exemplo, no ano atual 2019, no primeiro semestre, existem três cursos em andamento: Análise e Desenvolvimento de Sistemas (ADS), Gestão da Produção Industrial (GPI) e Gestão de Recursos Humanos (GRH). Os cursos de ADS e GPI possuem aulas no período matutino e noturno, e 6 semestres com 800 minutos semanais cada período. No caso do GRH, as aulas são ministradas apenas no período noturno, contendo 800 minutos semanais em cada semestre, e tendo, até o momento, somente 2 semestres.

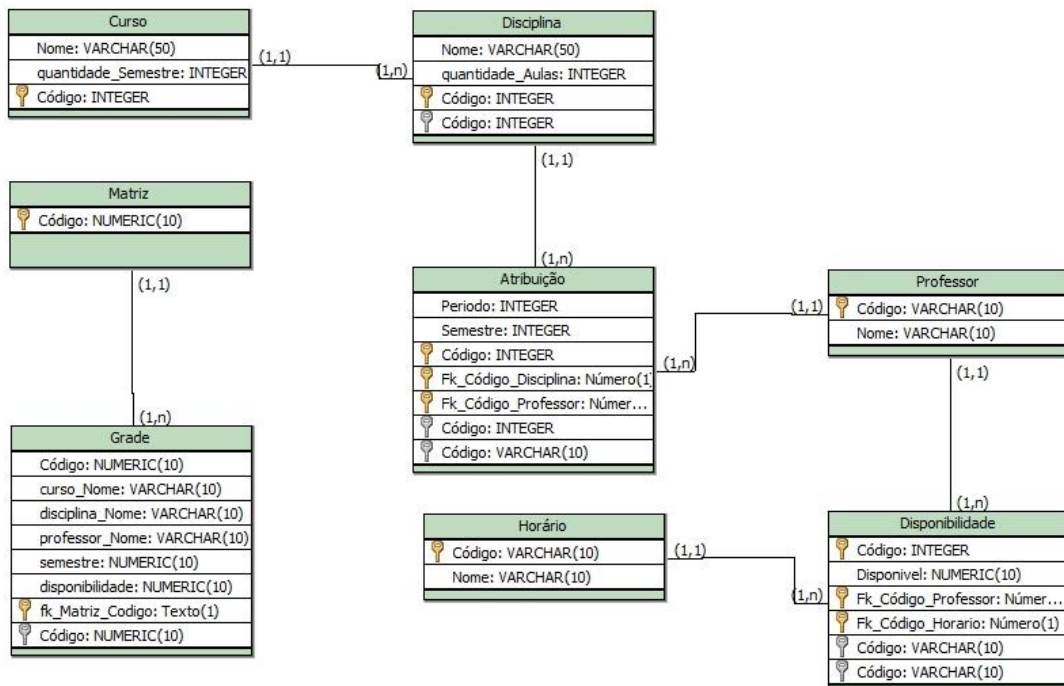
Além disso, cada curso possui várias disciplinas, cada disciplina possui suas especificidades, e para cada disciplina podem existir professores distintos para lecioná-las, em cada período. Com estas informações já se consegue material para relacionar as informações e abastecer o Banco de Dados. Para o desenvolvimento do modelo conceitual e lógico do Banco de Dados, utilizou-se o software gratuito brModelo 3.0, que é apresentado nas Figuras 10 e 11.

Figura 10 - Modelo conceitual do Banco de Dados



Fonte: Autor.

Figura 11 - Modelo lógico do Banco de Dados



Fonte: Autor.



### 3.1 Modelando o indivíduo no problema proposto

Após a montagem do Banco de Dados e das atribuições de valores inseridos nas tabelas, os horários passam a ser divididos em blocos de horários conforme mostra a Figura 12.

**Figura 12** - Tabela de Horários da Fatec Franca

Período	código	Nome	código	Nome	código	Nome	código	Nome	código	Nome	código	Nome
<b>Período Matutino</b>												
1	1	Segunda - 07:40   - 09:20	4	Terça - 07:40   - 09:20	7	Quarta - 07:40   - 09:20	10	Quina - 07:40   - 09:20	13	Sexta - 07:40   - 09:20	-	-
1	2	Segunda - 09:30   - 11:10	5	Terça - 09:30   - 11:10	8	Quarta - 09:30   - 11:10	11	Quinta - 09:30   - 11:10	14	Sexta - 09:30   - 11:10	-	-
1	3	Segunda - 11:20   - 13:00	6	Terça - 11:20   - 13:00	9	Quarta - 11:20   - 13:00	12	Quinta - 11:20   - 13:00	15	Sexta - 11:20   - 13:00	-	-
<b>Período Noturno</b>												
2	16	Segunda - 19:00   - 20:40	18	Terça - 19:00   - 20:40	20	Quarta - 19:00   - 20:40	22	Quina - 19:00   - 20:40	24	Sexta - 19:00   - 20:40	26	Sábado - 09:30   - 11:10
2	17	Segunda - 20:50   - 22:30	19	Terça - 20:50   - 22:30	21	Quarta - 20:50   - 22:30	23	Quinta - 20:50   - 22:30	25	Sexta - 20:50   - 22:30	27	Sábado - 11:20   - 13:00

Fonte: Autor.

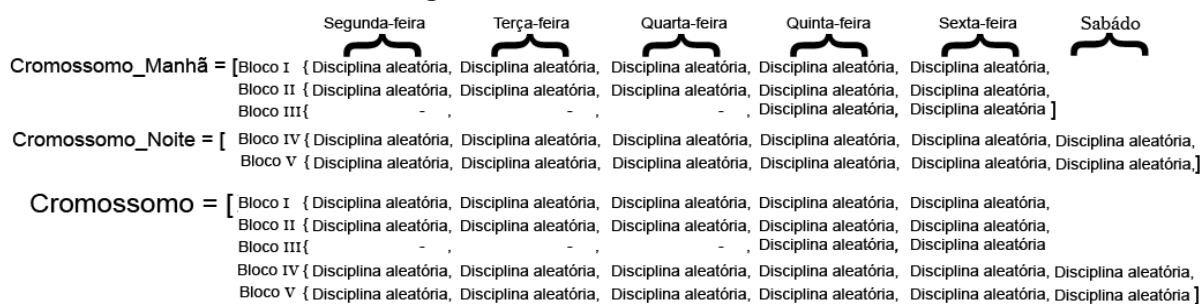
Conforme a tabela de horários da Figura 12, no caso da instituição estudada e respeitando as informações obtidas, os horários serão definidos como o cromossomo do nosso indivíduo, sendo que cada indivíduo terá uma variável nota e tempo de inicialização de população. Neste exemplo, iniciamos os horários da primeira posição até a décima quinta onde serão tratados os horários do matutino. Já do décimo sexto ao vigésimo sétimo serão tratados o período noturno e sua junção formará uma lista que será denominada cromossomo. Em cada posição serão atribuídos uma disciplina e um professor, porém, em cromossomos distintos. Cada uma destas posições será denominada de gene.

$$Cromossomo\_Matutino = arredondado\_para\_cima(tamanho(Cromossomo) \div 2) + 1$$

$$Cromossomo\_Noturno = tamanho(cromossomo) - tamanho(Cromossomo\_Matutino)$$

Prosseguindo, para a inicialização e desenvolvimento do cromossomo, em cada gene do período matutino, será atribuída uma disciplina, sendo verificado se a ela ainda poderia ser atribuída pela sua quantidade, isso irá se repetir até que a quantidade de aulas atribuídas seja igual ao número do somatório de uma lista das quantidades de aulas. O mesmo será feito na parte do cromossomo noturno. Depois será feita a junção destas partes, como mostrado na Figura 13.

**Figura 13 - Cromossomo utilizado**



Fonte: Autor.

### 3.2 Método de avaliação

Posteriormente, será aplicado o método de avaliação, que foi desenvolvido para pontuar com o número um cada bloco de horário que for encaixado um professor e este estiver com este bloco disponível, então adicionado 1 (um) em sua soma. Neste trabalho utilizaremos 27 blocos de horários, destes 24 são utilizáveis, 12 são no período matutino e também no período noturno. Deste modo, a somatória da avaliação perfeita teria 144, nos 2 períodos do curso ADS, assim como no curso de GPI, sendo o total esperado de 288. Porém, tem mais 12 blocos para cada semestre disponibilizados para o curso de GRH, como estava disponível na data de realização deste trabalho contando com 2 semestre com 12 aulas cada, temos mais 24 blocos a serem avaliados, resultando em um total de 312. Portanto, 312 seria o somatório perfeito e a solução perfeita para nosso problema e a mínima seria 0 (zero).

### 3.3 Método de cruzamento

Dando continuidade, na função de cruzamento depois de selecionados os pais pelas avaliações, eles se cruzam por um número de corte aleatório, recebendo características de seus dois genitores. Neste caso, a função irá retornar dois filhos e será realizada uma nova geração, definindo o valor desta.

### 3.4 Método de mutação

Ao iniciar a execução do método de mutação, em sua chamada levará um parâmetro, sendo este definido como a taxa de mutação, nela será atribuído um

valor, que definirá se o gene terá uma mutação ou não, dependendo do teste e do tipo de mutação a ser feita. Tanomaru (1995, p. 9) cita e recomenda que uma boa taxa de mutação é de 0,01 para não modificar muito o cromossomo que é um possível resultado do problema.

### 3.5 Geração de população

Enfim, após os filhos serem gerados e implantado o método de mutação, serão selecionados os filhos pelo método da roleta, utilizando neste caso a ideia de frequência acumulada, medindo as melhores notas dentre todas as gerações realizando o somatório e deste somatório, utilizaremos uma porcentagem para selecionar os novos pais cada qual com suas respectivas notas, tendo assim uma boa chance de um bom indivíduo ser selecionado até a origem de uma próxima geração, e assim sucessivamente.

Depois de realizados todos os processos, tem-se uma nova população que sustentará as próximas gerações, desencadeando um ciclo, que buscará boas soluções.

## 4 Resultados e discussão

Considera-se que os resultados obtidos foram satisfatórios, pela geração de várias soluções em curto espaço de tempo, modificando estes resultados para uma possível mudança para melhor. Pelos testes com 100 gerações de 20 indivíduos as notas ficam em torno de 304 a 308 levando 45 segundos para se obter 20 resultados, levando em consideração que temos uma base de disponibilidade dos professores aleatória.

Caso fosse almejado testar todas as possibilidades do problema para a instituição teríamos um novo arranjo de contagem onde em cada semestre de cada curso surgiria possibilidades de encaixes de disciplinas e blocos de horários.

*Semestre\_matutino*

= {-, -, -, disciplina\_1, disciplina\_1, disciplina\_2, disciplina\_2, disciplina\_3, disciplina\_3, disciplina\_4, disciplina\_5, disciplina\_6, disciplina\_6, disciplina\_7}

Assim, temos 15 elementos, e 15 espaços disponíveis, se cada elemento não puder ser utilizado mais de uma vez, então teremos uma combinação de 15! (quinze fatorial), isso para o período matutino de um semestre de um curso. No período noturno temos os seguintes

*Semestre\_Noturno*

= {disciplina\_1, disciplina\_1, disciplina\_2, disciplina\_2, disciplina\_3, disciplina\_3, disciplina\_4, disciplina\_5, disciplina\_6, disciplina\_6, disciplina\_7}

Neste caso, temos 12 elementos, com 12 espaços a serem preenchidos, e como estes elementos após serem atribuídos não podem ser repetidos, teremos assim um arranjo de 12! (doze fatorial). Adicionando com a contagem do primeiro período teremos 15! + 12!, na instituição utilizada para realizar o trabalho em 2 cursos que possuem 6 semestres e 1 curso com 2 semestres, sendo este disponibilizado apenas no período noturno. Havendo a mesma quantidade de aulas e elementos nos semestres, portanto, teremos uma contagem total de:

$$\begin{aligned} \text{numero\_Total\_Contagem} &= \text{Curso\_ADS} + \text{Curso\_GPI} + \text{Curso\_GRH} \\ \text{numero\_Total\_Contagem} &= (15! + 12!) \times 6 + (15! + 12!) \times 6 + (12!) \times 2 \end{aligned}$$

Vale ressaltar que este número de contagem não leva em consideração algumas restrições como: mesmo bloco de horários, um mesmo professor, horários disponíveis no bloco sorteado e se o horário está legalizado de acordo com as leis trabalhistas de jornada de trabalho dos professores.

Na Figura 14 está representado parte do resultado, na qual se encontra o horário dos três primeiros semestres do curso de ADS, gerado pela aplicação. Este resultado obteve a nota de avaliação igual a 306 pontos dos 312 possíveis, sendo obtido pelas informações manipuladas do Banco de Dados, com 98% de aproveitamento. Assim, analisando o resultado obtido, conclui-se que a aplicação pode ser de grande auxílio na resolução deste problema.

Figura 14 - Exemplo de resultado matriz de horários

Horários Gerados nota:306					
ADS Matutino 1º semestre					
semestre:1 ADS- Matematica Discreta Silvia Regina Viel	semestre:1 - -	semestre:1 ADS- Administração Geral Ana Claudia Nery Salomão	semestre:1 ADS- Laboratorio de Hardware Carlos Eduardo de França Roland	semestre:1 - -	semestre:1 ADS- Administração Geral Ana Claudia Nery Salomão
semestre:1 Programação em Microinformatica Ely Fernando do Prado	semestre:1 ADS- Matematica Discreta Silvia Regina Viel	semestre:1 ADS- Algoritmo e logica de Programação Jaqueline Brigadori Pugliesi	semestre:1 ADS- Arquitetura e Organização de Computadores Antonio Clementino Neto	semestre:1 ADS- Algoritmo e logica de Programação Jaqueline Brigadori Pugliesi	semestre:1 ADS- Laboratorio de Hardware Carlos Eduardo de França Roland
semestre:1 ADS- Administração Geral Ana Claudia Nery Salomão	semestre:1 ADS- Arquitetura e Organização de Computadores Antonio Clementino Neto	semestre:1 - -	semestre:1 ADS- Inglês I Marcia Helena Venâncio Faleiros	semestre:1 Programação em Microinformatica Ely Fernando do Prado	
ADS Noturno 1º semestre					
semestre:1 ADS- Algoritmo e logica de Programação Jaqueline Brigadori Pugliesi	semestre:1 ADS- Arquitetura e Organização de Computadores Antonio Clementino Neto	semestre:1 ADS- Algoritmo e logica de Programação Jaqueline Brigadori Pugliesi	semestre:1 ADS- Arquitetura e Organização de Computadores Antonio Clementino Neto	semestre:1 ADS- Arquitetura e Organização de Computadores Antonio Clementino Neto	
semestre:1 ADS- Arquitetura e Organização de Computadores Antonio Clementino Neto	semestre:1 ADS- Inglês I Flavia Herker Lopes Bernabé	semestre:1 ADS- Administração Geral Ana Claudia Nery Salomão	semestre:1 Programação em Microinformatica Jorge Luís Takahashi Hattori	semestre:1 ADS- Matematica Discreta Mario Tomoyoshi Tsuchiya	
ADS Matutino 2º Semestre					
semestre:2 ADS- Contabilidade Tessia Berber Teixeira Maranhã	semestre:2 ADS- Engenharia de Software I Fernando Martins	semestre:2 ADS- Comunicação e Expressão Liene Cunha Viana Bittar	semestre:2 ADS- Comunicação e Expressão Liene Cunha Viana Bittar	semestre:2 - -	semestre:2 ADS- Calculo Mario Tomoyoshi Tsuchiya
semestre:2 ADS- Linguagem de Programação Daniel Facciolo Pires	semestre:2 ADS- Calculo Mario Tomoyoshi Tsuchiya	semestre:2 ADS- Inglês II Marcia Helena Venâncio Faleiros	semestre:2 - -	semestre:2 ADS- Engenharia de Software I Fernando Martins	semestre:2 ADS- Calculo Mario Tomoyoshi Tsuchiya
semestre:2 ADS- Linguagem de Programação Daniel Facciolo Pires	semestre:2 ADS- Calculo Mario Tomoyoshi Tsuchiya	semestre:2 ADS- Sistema de Informação Carlos Eduardo de França Roland	semestre:2 ADS- Sistema de Informação Carlos Eduardo de França Roland	semestre:2 - -	
ADS Noturno 2º semestre					
semestre:2 ADS- Inglês II Flavia Herker Lopes Bernabé	semestre:2 ADS- Comunicação e Expressão Liene Cunha Viana Bittar	semestre:2 ADS- Engenharia de Software I Fernando Martins	semestre:2 ADS- Sistema de Informação Carlos Eduardo de França Roland	semestre:2 ADS- Sistema de Informação Carlos Eduardo de França Roland	
semestre:2 ADS- Engenharia de Software I Fernando Martins	semestre:2 ADS- Linguagem de Programação Fausto Gonçalves Cintra	semestre:2 ADS- Linguagem de Programação Fausto Gonçalves Cintra	semestre:2 ADS- Comunicação e Expressão Liene Cunha Viana Bittar	semestre:2 ADS- Contabilidade Tessia Berber Teixeira Maranhã	
ADS Matutino 3º Semestre					
semestre:3 ADS- Estrutura de dados Jaqueline Brigadori Pugliesi	semestre:3 ADS- Estatística Aplicada Maria Luisa Cervi Uzun	semestre:3 ADS- Engenharia de Software II Ely Fernando do Prado	semestre:3 - -	semestre:3 ADS- Estrutura de dados Jaqueline Brigadori Pugliesi	semestre:3 ADS- Estatística Aplicada Maria Luisa Cervi Uzun
semestre:3 ADS- Sociedade e Tecnologia Marcelo de Moraes Montanar	semestre:3 - -	semestre:3 ADS- Interação Humano Computador Carlos Eduardo de França Roland	semestre:3 ADS- Interação Humano Computador Carlos Eduardo de França Roland	semestre:3 ADS- Estatística Aplicada Maria Luisa Cervi Uzun	semestre:3 ADS- Sociedade e Tecnologia Marcelo de Moraes Montanar
semestre:3 ADS- Sistema Operacionais I Antonio Fernando Traina	semestre:3 - -	semestre:3 ADS- Economia e Finanças Sergio Ishikawa	semestre:3 ADS- Engenharia de Software II Ely Fernando do Prado	semestre:3 ADS- Inglês III Marcia Helena Venâncio Faleiros	
ADS Noturno 3º semestre					
semestre:3 ADS- Interação Humano Computador Carlos Eduardo de França Roland	semestre:3 ADS- Engenharia de Software II Ely Fernando do Prado	semestre:3 ADS- Economia e Finanças Sergio Ishikawa	semestre:3 ADS- Sistema Operacionais I Antonio Fernando Traina	semestre:3 ADS- Engenharia de Software II Ely Fernando do Prado	
semestre:3 ADS- Estatística Aplicada Maria Luisa Cervi Uzun	semestre:3 ADS- Interação Humano Computador Carlos Eduardo de França Roland	semestre:3 ADS- Estrutura de dados Jaqueline Brigadori Pugliesi	semestre:3 ADS- Inglês III Flavia Herker Lopes Bernabé	semestre:3 ADS- Estrutura de dados Jaqueline Brigadori Pugliesi	
ADS Matutino 3º Semestre					
semestre:4 ADS- Metodologia da Pesquisa Científica-Tecnologica Carlos Eduardo de França Roland	semestre:4 Programação Orientada a Objetos Daniel Facciolo Pires	semestre:4 Programação Web Fausto Gonçalves Cintra	semestre:4 ADS- Engenharia de Software III Antonio Fernando Traina	semestre:4 ADS- Sistema Operacionais II Antonio Fernando Traina	semestre:4 ADS- Sistema Operacionais II Antonio Fernando Traina
semestre:4 ADS- Sistema Operacionais II Antonio Fernando Traina	semestre:4 ADS- Engenharia de Software III Antonio Fernando Traina	semestre:4 - -	semestre:4 Programação Web Fausto Gonçalves Cintra	semestre:4 - -	semestre:4 ADS- Engenharia de Software III Antonio Fernando Traina
semestre:4 ADS- Banco de Dados Claudio Eduardo Paiva	semestre:4 - -	semestre:4 Programação Orientada a Objetos Daniel Facciolo Pires	semestre:4 ADS- Inglês IV Flavia Herker Lopes Bernabé	semestre:4 ADS- Banco de Dados Claudio Eduardo Paiva	
ADS Noturno 3º semestre					
semestre:4 ADS- Banco de Dados Claudio Eduardo Paiva	semestre:4 Programação Web Fausto Gonçalves Cintra	semestre:4 ADS- Engenharia de Software III Antonio Fernando Traina	semestre:4 Programação Orientada a Objetos Daniel Facciolo Pires	#REF!	
semestre:4 ADS- Inglês IV Flavia Herker Lopes Bernabé	semestre:4 ADS- Banco de Dados Claudio Eduardo Paiva	semestre:4 ADS- Sistema Operacionais II Antonio Fernando Traina	semestre:4 Programação Orientada a Objetos Daniel Facciolo Pires	semestre:4 ADS- Metodologia da Pesquisa Científica- Tecnologica Carlos Eduardo de França Roland	

Fonte: Autor.

Ao realizar a execução da aplicação em 20 indivíduos foram gastadas 100 gerações, cujo tempo de execução foi otimizado, gastando aproximadamente 26s.30ms (vinte e seis segundos e 30 milésimos de segundos), obtendo o número de avaliações entre 306 a 302. O número de gerações e indivíduos pode variar dependendo do problema. Neste caso, o número de possibilidade seria muito alto e,

pelos cálculos de tempo de execução, para realizar um teste com o número total de possibilidades a espera seria imensa. Como exemplo poderíamos citar um computador, com processador i7 8ª geração com 2.20 Ghz, o tempo de espera seria de aproximadamente 113 décadas. Mesmo assim não significa que o algoritmo iria gerar todas as soluções possíveis, pois ele trabalha de forma aleatória, mas em certos casos poderia conseguir gerar soluções com melhores notas.

### Considerações finais

O artigo desenvolvido propôs uma aplicação para auxiliar a montagem de matrizes de horários acadêmicos, uma vez que esta tarefa é um trabalho que demanda enorme espaço de tempo, sem dizer das diversas variáveis que devem ser consideradas, como disponibilidades dos professores em blocos de horários, não ser o mesmo docente que atua em um mesmo bloco de horário entre dois ou mais cursos, considerar a legislação trabalhista na área específica, dentre outras.

Importante frisar que toda instituição acadêmica necessita realizar este processo para o início de suas atividades e mesmo depois de todo trabalho realizado e muitas vezes realizado por tentativa e erro, ainda assim, nem sempre se consegue obter um bom resultado. Partindo desta premissa poderíamos denominar este exercício como um problema enfrentado pelas instituições acadêmicas.

Então, após a análise do problema enfrentado pelas instituições de ensino, nasceu a motivação para realização este trabalho, que poderá representar uma possível solução para esta tarefa, utilizando-se de métodos de Inteligência Artificial, os Algoritmos Genéticos, partindo dos princípios desenvolvidos por Darwin em seus estudos sobre a seleção natural, contudo, aplicados à computação e moldado para obter soluções de vários problemas.

Neste método se destacam funções essenciais como desenvolvimento do indivíduo e seus cromossomos, avaliação, cruzamento e mutação, e após realização destes processos ocorre a seleção dos filhos que reinicializam o processo com a nova geração.

Em seguida, para modelarmos o problema da instituição, foi desenvolvida uma ligação entre as informações de cursos, semestres, disciplinas, professores e horários. Estas ligações aconteceram em níveis, como no caso dos cursos semestrais, que cada semestre possui suas respectivas disciplinas, com seus

respectivos professores, e suas mais diversas especificidades e disponibilidades. Então, após o modelo desenvolvido neste trabalho foram utilizados dois cromossomos, um com os blocos de horários designando cada curso com seu respectivo código de identificação e outro específico aos professores com a mesma identificação.

Neste processo de desenvolvimento, houve enorme dificuldade em vários fatores como a modelagem do Banco de Dados e quais seriam suas relações. Vale ressaltar que a princípio a aplicação foi pensada para funcionar com disciplinas semelhantes a vários cursos como acontece em alguns casos com disciplinas comuns, como o cálculo, mas, há existência de algumas diferenças de curso para curso em certas disciplinas, viabilizando a programação e manipulação dos dados. Também houve problema após o cruzamento, pois se houvessem disciplinas com quantidades de aulas acima de seu modelo original, geraria incompatibilidades e a matriz seria inutilizável. Desta forma, foram feitas funções para verificar e corrigir este possível erro. Para finalizar foi identificado o problema de horário onde um professor que teria realizado a aula no último bloco de horário no período noturno não poderia realizar a próxima aula no primeiro período matutino, ou seja, no primeiro bloco de horário do dia seguinte.

A aplicação deste programa trouxe resultados de maneira rápida e eficaz, gerando vários resultados que poderiam ser selecionados para melhor atender as necessidades de cada instituição, porém, as informações geradas estão restritas ao Banco de Dado, não gerando interfaces gráficas.

### **Agradecimentos**

Agradeço primeiramente à Deus que permitiu que tudo isso acontecesse, ao longo de minha vida por ter me dado saúde e força para superar minhas dificuldades, por minha vida, não somente nestes anos como universitário, mas que em todos os momentos pois é o maior mestre que alguém pode conhecer. Ao meu pai que apesar de todas as dificuldades me fortaleceu e que para mim foi muito importante, além de ser paciente no trabalho de revisão da redação. À Instituição pelo ambiente criativo e amigável que proporcionou. À minha orientadora Dra. Jaqueline Brigladori Pugliesi pelo empenho e dedicação e o entusiasmo pelo desenvolvimento deste trabalho, e ao professor Me. Cláudio Eduardo Paiva, pela



orientação, apoio e confiança. A todos os outros professores por me proporcionar o conhecimento não apenas racional, mas a manifestação do caráter e afetividade da educação no processo de formação profissional, por tanto que se dedicaram a mim, não somente por terem me ensinado, mas por terem me feito aprender. A palavra mestre, nunca fará justiça aos professores dedicados aos quais sem nominar terão os meus eternos agradecimentos.

### Referências

DARWIN, C. **A Origem das Espécies**. Tradução de Mesquita Paul. PORTO: [s.n.], v. 1, 2003. 572 p. Disponível em: <<https://drive.google.com/open?id=1enN2Dsuu5iSU1p6Zvndul8CIhr1AYyYb>>. Acesso em: 21.mar. 2019.

ESTRANHO, R. M. Qual a diferença entre DNA, gene e cromossomo? **Ciência, Mundo Estranho**, p. 4, 18 Abril 2011. Disponível em: <<https://super.abril.com.br/mundo-estranho/qual-a-diferenca-entre-dna-gene-e-cromossomo/>>. Acesso em: 26.mar.2019.

IKEDA, P. A. Introdução aos Algoritmos Genéticos, 2010. Disponível em: <<https://www.ime.usp.br/~gold/cursos/2009/mac5758/PatriciaGenetico.pdf>>. Acesso em: 25.mar.2019.

LACERDA, E. G. M. D.; CARVALHO, A. C. P. L. F. D. INTRODUÇÃO AOS ALGORITMOS. In: GALVÃO, C. O.; VALENÇA, M. **Sistemas inteligentes**. Porto Alegre: ABRH., 1999.

NRIA, S. et al. Roteamento de Veículos Utilizando Otimização por Colônia de Formigas e Algoritmo Genético, 2013. ISSN 10.7436. Disponível em: <[https://www.researchgate.net/publication/300661461\\_Roteamento\\_de\\_Veiculos\\_Utilizando\\_Otimizacao\\_por\\_Colonia\\_de\\_Formigas\\_e\\_Algoritmo\\_Genetico](https://www.researchgate.net/publication/300661461_Roteamento_de_Veiculos_Utilizando_Otimizacao_por_Colonia_de_Formigas_e_Algoritmo_Genetico)>. Acesso em: 28.mar.2019.

RUSSEL, S.; NORVIG, P. **Inteligência Artificial**. Tradução de Regina Célia Simille. 3ª. ed. Rio de Janeiro: Elsevier Editora Ltda, 2013. 1323 p.

SILVEIRA, R. V. M. D.; AMABIS, J. M. COMO OS ESTUDANTES DO ENSINO MÉDIO RELACIONAM OS CONCEITOS. **IV ENCONTRO NACIONAL DE PESQUISA EM EDUCAÇÃO EM CIÊNCIAS**, São Paulo, 2 Junho 2004. 12. Disponível em: <<http://fep.if.usp.br/~profis/arquivos/ivenpec/Arquivos/Orais/ORAL052.pdf>>. Acesso em: 26.mar.2019.

SOUZA, C. C. D. Otimização Combinatória, 2019. Disponível em: <<http://www.ic.unicamp.br/~cid/intro-comb-opt.html>>. Acesso em: 26.mar.2019.

TANOMARU, J. Motivação, Fundamentos e Aplicações de Algoritmos Genéticos. II **Congresso Brasileiro de Redes Neurais**, Curitiba, 01 Novembro 1995. 31. Disponível em: <<https://www.dca.ufrn.br/~meneghet/FTP/Motiva%E7%E3o%20Fundamentos%20e%20Aplica%E7%F5es%20de%20Algoritmos%20Gen%E9ticos.pdf>>. Acesso em: 26.mar.2019.