

1 2 9 0



UNIVERSIDADE D
COIMBRA

FUNDAMENTOS DE INTELIGÊNCIA
ARTIFICIAL

Trabalho Prático 2

REALIZADO POR:

Ana Carolina Morais N°2021222056

- email: anamorais@student.dei.uc.pt

Fernanda Fernandes N°2021216620

- email: mrfernandes@student.dei.uc.pt

Pedro Castro N°2021198420

- email: uc2021198420@student.uc.pt

Turma PL4

Turma: PL4

Turma: PL5

INTRODUÇÃO

O presente relatório pretende abordar as componentes de um algoritmo evolucionário, para criar um veículo motorizado. Ao longo deste trabalho, exploraremos os conceitos essenciais envolvidos na concepção e no desenvolvimento de um algoritmo evolucionário, com foco no desenvolvimento de veículos adequados ao cenário em questão. O algoritmo irá construir o veículo com base num conjunto de parâmetros (número de rodas, tamanho das rodas, etc).

Para isso, foi utilizado um simulador com recurso ao 'Software' *Unity*. É através deste simulador, que podemos testar o desempenho do veículo em vários cenários, incluindo trajetos com buracos e quedas.

META 1 – MODELAÇÃO E DESENVOLVIMENTO

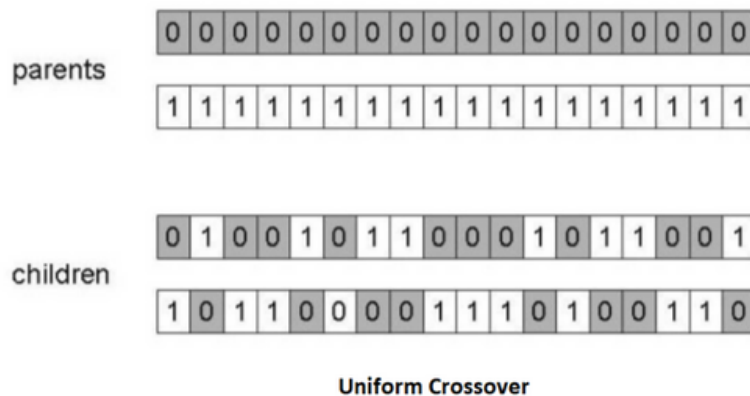
Neste capítulo explicamos todas as etapas envolvidas neste trabalho para o desenvolvimento do Algoritmo Genético. Assim, para esta meta, foi necessário a implementação de funcionalidades básicas do algoritmo, sendo estas, **recombinação**, **seleção de progenitores**, **mutação**, **aptidão** e ainda definir os **parâmetros** do algoritmo.

• RECOMBINAÇÃO (CROSSOVER)

A etapa de recombinação, é o momento onde se recombina os genes para a formação de uma nova geração. Existem diversas técnicas de recombinação, como, por exemplo:

- RECOMBINAÇÃO COM UM PONTO DE CORTE
- RECOMBINAÇÃO COM $N(N > 1)$ PONTOS DE CORTE
- RECOMBINAÇÃO UNIFORME
- Entre outras.

Após a análise dos vários tipos, concluímos que o melhor a implementar seria o operador de recombinação uniforme. Na recombinação uniforme, cada gene do descendente tem a mesma probabilidade de ser herdado de qualquer um dos pais. Isso é diferente de outros métodos de recombinação, como o crossover de um ponto de corte ou de dois pontos, onde blocos de genes são copiados de um único pai para os descendentes. Em contraste, na recombinação uniforme, cada gene é considerado separadamente, e para cada gene, escolhe-se aleatoriamente de qual pai ele será copiado. Abaixo, encontra-se uma figura que representa esta mesma recombinação.



Esta é uma técnica muito simples de implementar, porém, tem algumas limitações: por vezes, o indivíduo pode herdar um mau gene de um progenitor e o melhor gene pode ser perdido. No entanto, se tivermos dois pais bons, tipicamente, teremos alta probabilidade de ter um bom filho e, no caso de ele não ser bom, será eliminado na próxima geração (seleção natural).

• MUTAÇÃO

A mutação é um elemento crucial nos algoritmos genéticos, cujo objetivo é manter e introduzir diversidade genética na população de soluções ao longo de gerações. Em suma, a mutação modifica aleatoriamente partes do genótipo (ou seja, os genes) de um indivíduo, o que pode resultar na descoberta de novas e, possivelmente, melhores soluções. Após a recombinação genética é aplicado um processo de mutação. Tal como a recombinação existem vários tipos de mutação, tal como:

- MUTAÇÃO DE BIT FLIP
- MUTAÇÃO UNIFORME
- MUTAÇÃO GAUSSIANA
- Entre outras.

Após a análise dos vários tipos, concluímos que o melhor a implementar seria o algoritmo de **mutação gaussiana**. Pois, esta mutação envolve adicionar um valor aleatório retirado de uma distribuição normal e converge mais rapidamente para soluções ótimas. Neste algoritmo, cada gene dos cromossomos de um indivíduo têm probabilidade igual de sofrer uma mutação, ao qual o novo gene é calculado na base de uma função gaussiana[1].

• SELEÇÃO DE PROGENITORES

A seleção de progenitores é uma etapa importante de um algoritmo genético porque determina quais indivíduos da população atual terão a oportunidade de se reproduzir e transmitir os seus genes para a próxima geração.

A escolha dos pais é um ponto crucial na convergência do nosso algoritmo evolucionário, pois bons pais têm tendência a gerar filhos melhores e mais adequados ao meio. Vários métodos de seleção são utilizados, cada um com características específicas que podem afetar o desempenho do algoritmo.

- SELEÇÃO POR ROLETA
- SELEÇÃO POR TORNEIO
- MÉTODO DE AMOSTRAGEM UNIVERSAL ESTOCÁSTICA
- Entre outras.

Após a análise dos vários tipos, concluímos que o melhor a implementar seria a **seleção por roleta**.

A seleção por roleta em algoritmos genéticos é um método onde a oportunidade de um indivíduo ser escolhido para reprodução é proporcional à sua aptidão. Em outras palavras, indivíduos mais aptos têm maiores segmentos na "roleta" imaginária e, portanto, maior probabilidade de serem selecionados. Este método simples favorece os melhores indivíduos, o que pode acelerar a convergência para soluções de alta qualidade. No entanto, uma desvantagem é que pode levar à perda de diversidade genética, pois indivíduos altamente aptos podem dominar rapidamente a população.

• APTIDÃO

A aptidão é uma métrica crucial em algoritmos evolucionários para avaliar se um indivíduo possui características genéticas adequadas para desempenhar bem a tarefa designada. Esta função, chamada de função de aptidão, é essencial para orientar todo o processo de seleção, mutação e recombinação, selecionando indivíduos com alta aptidão para reprodução. Isso favorece a criação de novas gerações que possuem características genéticas mais propensas a resolver o problema proposto.

No momento da avaliação de um indivíduo, obtemos um valor que corresponde à qualidade do mesmo. Este valor será maior quanto melhor for o desempenho do indivíduo. É bastante importante que esta função esteja bem definida, para evitar excluir indivíduos que representem uma boa solução, o que irá, posteriormente, influenciar a evolução deste mesmo algoritmo.

Neste contexto, elaborámos uma função de aptidão que contém todas as variáveis disponíveis do indivíduo, dando diferentes pesos a cada uma, uma vez que, por exemplo, a **distância** e o **tempo decorrido** são indicadores diretos do desempenho do veículo, outro caso, o **número de rodas** e a **massa do carro**: são variáveis que podem afetar a estabilidade e a eficiência do veículo. Embora possam parecer menos importantes à primeira vista, elas podem ter implicações significativas no comportamento dinâmico do carro, influenciando como ele responde em diferentes cenários.

Ponderámos descartar algumas variáveis para simplificar a função, mas resultaria na perda de informação crucial que afeta a precisão da simulação.

Assim, ao construir, desta forma, a função de aptidão, não só amplia a profundidade e a aplicabilidade dos resultados da simulação, mas também aumenta a capacidade de alcançar um 'design' otimizado e funcionalmente completo do veículo.

```
// QUANTO MAIS LONGE O CARRO CHEGA, MELHOR É O SEU DESEMPENHO.
FLOAT DISTANCEWEIGHT = 1.0F;
FITNESS += DISTANCE * DISTANCEWEIGHT;

// QUANTO MAIS TEMPO O CARRO LEVA PARA COMPLETAR A PISTA, PIOR É O SEU DESEMPENHO.
FLOAT TIMEWEIGHT = -0.5F;
FITNESS += ELAPSEDTIME * TIMEWEIGHT;

// CARROS COM MAIS RODAS PODEM SER MAIS PESADOS E MENOS EFICIENTES
FLOAT WHEELPENALTY = -0.1F;
FITNESS += NUMBEROFWHEELS * WHEELPENALTY;

// CARROS MAIS PESADOS PODEM SER MAIS DIFÍCEIS DE CONTROLAR
FLOAT MASSPENALTY = -0.05F;
FITNESS += CARMASS * MASSPENALTY;

// CARROS QUE COMPLETAM A PISTA TÊM UM PESO EXTRA
FLOAT COMPLETIONBONUS = 100.0F;
FITNESS += ROADCOMPLETED * COMPLETIONBONUS;

FLOAT AVERAGEVELOCITYWEIGHT = 0.2F;
FLOAT AVERAGEVELOCITY = SUMVELOCITIES / VELOCITIES.COUNT;
FITNESS += AVERAGEVELOCITY * AVERAGEVELOCITYWEIGHT;

// ACELERAÇÕES MÉDIAS PODEM INDICAR UM BOM USO DE POTÊNCIA.
FLOAT AVERAGEACCELERATIONWEIGHT = 0.1F;
FLOAT AVERAGEACCELERATION = SUMACCELERATIONS / ACCELERATIONS.COUNT;
FITNESS += AVERAGEACCELERATION * AVERAGEACCELERATIONWEIGHT;

// FORÇAS MÉDIAS PODEM INDICAR UM BOM USO DE POTÊNCIA.
FLOAT AVERAGEFORCEWEIGHT = 0.1F;
FLOAT AVERAGEFORCE = SUMTOTALFORCES / FORCES.COUNT;
FITNESS += AVERAGEFORCE * AVERAGEFORCEWEIGHT;
```

• **PARAMETRIZAÇÃO**

Para esta fase são especificados os parâmetros do algoritmo evolucionário. Assim, para este projeto foram utilizados os seguintes parâmetros:

- Probabilidade de Mutação
- Tamanho do Elitismo
- Probabilidade de Crossover
- Número de Gerações

Estes parâmetros visam gerir como o algoritmo evolucionário se irá comportar. Assim, é possível alterar os valores ao longo das experiências de forma mais fácil.

• **EXPERIMENTAÇÃO E ANÁLISE – GAPROAD**

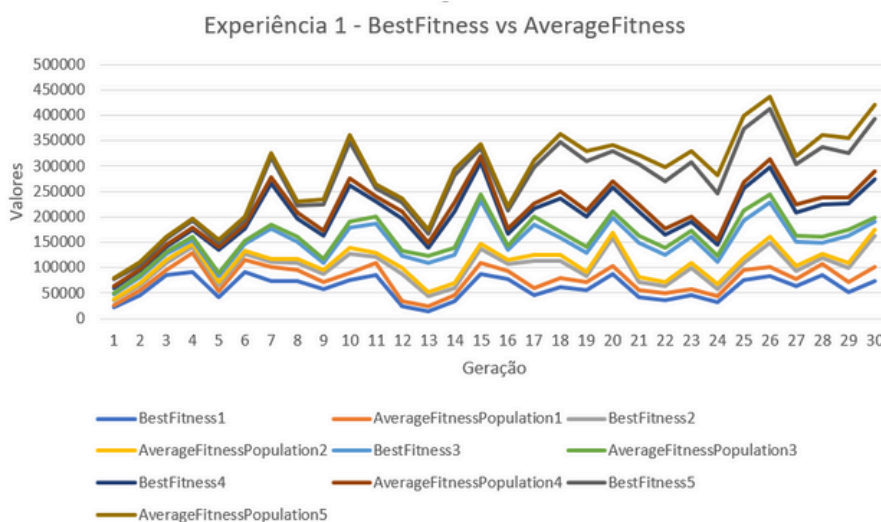
Após termos as funções devidamente implementadas, é vital testar as funcionalidades do algoritmo. Assim, vamos tentar estabelecer os melhores parâmetros, a partir da tabela abaixo. Uma vez que, ao contrário da tabela do enunciado que contém o parâmetro do torneio, como implementámos a seleção por roleta, esse parâmetro não se encontra na nossa tabela. Não esquecendo que para cada experiência é necessário realizar 5 execuções da mesma, para obter dados mais fidedignos.

	Mutação	Elitismo	Crossover	Número Gerações
Experiência 1	0.05	0	0.5	30
Experiência 2	0.2			
Experiência 3	0.05	2		
Experiência 4	0.2			

• EXPERIÊNCIA 1

Antes da realização desta experiência, existem algumas questões teóricas que podemos presumir. Uma vez que não apresenta qualquer valor de elitismo, não se selecionam os melhores agentes da geração atual para a próxima, o que pode prejudicar a evolução, impedindo a geração de novos veículos a partir dos melhores exemplares. Uma baixa probabilidade de mutação pode ser benéfica se já existirem carros eficazes, mantendo a qualidade dos descendentes próximos ao progenitor. Porém, se não houver bons indivíduos iniciais, essa mesma baixa probabilidade de mutação dificultará a evolução. Portanto, o impacto da probabilidade de mutação depende muito dos genótipos iniciais, podendo ser vantajoso mantê-la baixa se estes forem bons, ou necessitar de aumento se forem inadequados.

A partir da nossa função de aptidão, tivemos o seguinte resultado, o gráfico mostra a relação do Best Fitness com a Average Fitness.



Com base nos resultados obtidos, e, tendo em consideração a falta de elitismo, como explicado anteriormente, é de facto notória, pois não teremos pontos de referência para gerações futuras e, isso prejudicará o algoritmo a manter a estabilidade na procura de agentes que terminem os percursos. É de realçar que para além dos nossos valores serem bastantes altos, uma vez que a nossa função de aptidão tem em conta as demais variáveis, existem uma grande instabilidade entre os melhores indivíduos de cada geração onde a evolução será de certo modo aleatória.

De notar, também as baixas *AverageFitness*, tendo em conta todas as vezes que executamos a experiência, comprovando assim, que esta não será de todo a experiência mais aconselhada para atingir o nosso objetivo.

• EXPERIÊNCIA 2

A base desta experiência é muito semelhante à anterior, apenas é alterado o valor da mutação, para um valor maior. Em contrapartida, uma alta probabilidade de mutação pode ter benefícios se já houverem carros eficazes na população, mantendo a qualidade dos descendentes próxima à do progenitor. No entanto, se os indivíduos iniciais não forem bons o suficiente, uma alta probabilidade de mutação pode dificultar a evolução. Assim, o impacto da probabilidade de mutação depende significativamente da qualidade dos genótipos iniciais.

A partir da nossa função de aptidão, tivemos o seguinte resultado, o gráfico mostra a relação do Best Fitness com a Average Fitness.



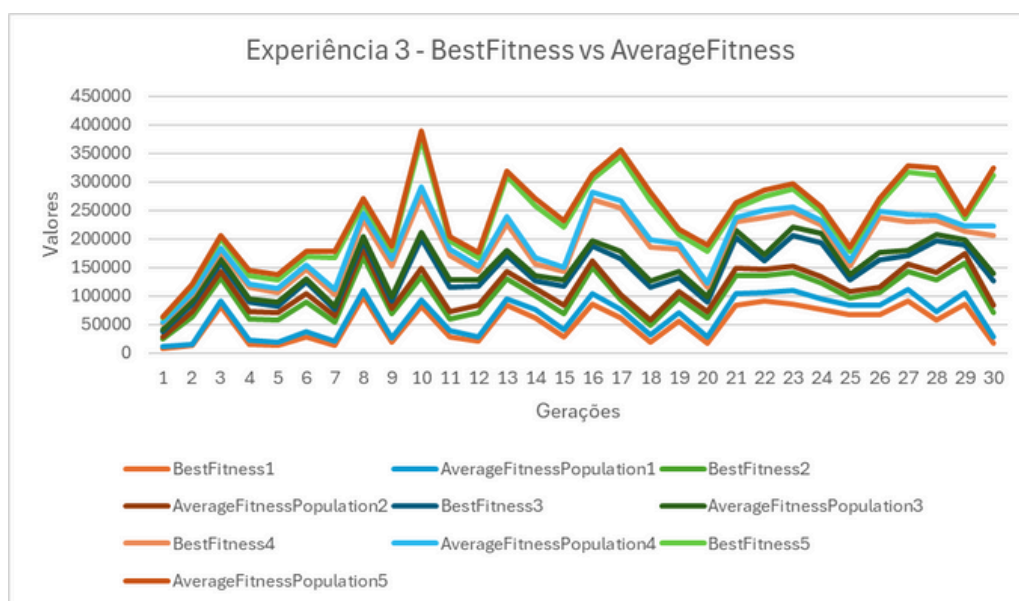
Com base nos resultados obtidos, e em comparação com a experiência anterior é de notar que os valores da experiência 2 são significativamente menores que os da experiência 1. Dado isto, percebemos que o aumento da probabilidade de mutação não é benéfica neste caso, continuando a existir uma grande instabilidade entre os melhores indivíduos de cada geração onde a evolução continua de certo modo aleatória.

Mais uma vez são bastante notórios os picos baixos da AverageFitness, tendo em conta todas as vezes que executamos a experiência, comprovando mais uma vez, que esta não é a experiência mais aconselhada para atingir o nosso objetivo.

• EXPERIÊNCIA 3

Antes da realização desta experiência, existem algumas questões teóricas que podemos presumir. Sendo que o valor de elitismo aumentou para 2, e o valor da mutação continua a ser 0.05. Ou seja, ao se manter uma probabilidade de mutação baixa poderá vir a ser benéfica caso já existam carros eficazes, mantendo assim a qualidade dos descendentes próxima à dos progenitores. E tendo em conta que o valor do elitismo aumentou, a escolha dos melhores agentes da geração atual para a seguinte será muito mais eficaz.

A partir da nossa função de aptidão, tivemos o seguinte resultado, o gráfico mostra a relação do Best Fitness com a Average Fitness.

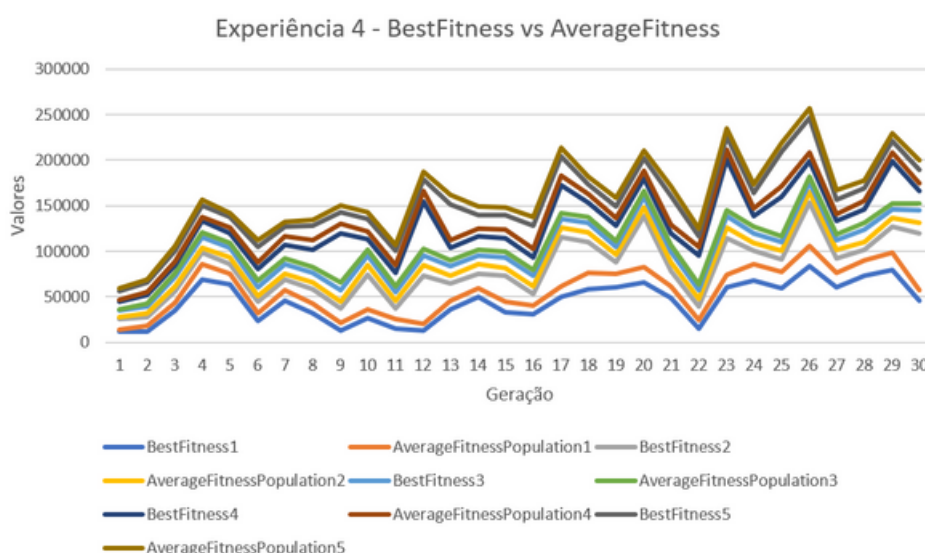


Com base nos resultados obtidos, e, tendo em consideração as suas alterações e as experiências anteriores, vemos uma estabilidade maior na evolução dos veículos, sendo que, numa primeira fase, ainda não se encontrou um veículo que consiga terminar o percurso, melhorando gradualmente o seu fitness ao longo das gerações. Isto acontece, uma vez que existe elitismo, pois ao possibilitar a escolha dos progenitores mais aptos, conseguimos gerar gerações cada vez melhores. A partir do momento, que um veículo consegue acabar a corrida, todas as gerações seguintes também o conseguem.

• EXPERIÊNCIA 4

Antes da realização desta experiência, existem algumas questões teóricas que podemos presumir. Uma vez que o valor de elitismo aumenta para 2, e a probabilidade de mutação aumenta para 0.2, é de realçar que com o aumento de probabilidade de mutação, a probabilidade de alterar progenitores benéficos para a experiência aumentará também, mas, por outro lado, a probabilidade de alterar progenitores com características menos favoráveis irá também ser superior.

A partir da nossa função de aptidão, tivemos o seguinte resultado, o gráfico mostra a relação do Best Fitness com a Average Fitness.



Com base nos resultados obtidos, e, tendo em consideração que se alterou a probabilidade de mutação de 0.05 para 0.2 (relativamente à experiência 3), o que, tecnicamente, faz com que haja alterações aos *offsprings* sejam mais significativas. Nesta experiência, reparamos que tem uma ligeira progressão. É uma progressão lenta relativamente à experiência anterior, pois só difere no valor da mutação. Isto acontece, porque quando um indivíduo tem uma boa prestação e é selecionado como progenitor da próxima geração, o seu descendente perde as suas boas características, devido à aleatoriedade do algoritmo.

CONCLUSÃO

Após a realização das experiências iniciais, concluímos, que nenhuma das soluções é ótima e vai de encontro aquilo que procuramos, no entanto, a experiência 3 apresentou os melhores resultados, isto, porque sem o elitismo, o algoritmo pode ter dificuldade em escapar de mínimos locais sub-ótimos. Se as melhores soluções não forem protegidas e mantidas, a busca pode ficar presa em regiões do espaço de solução que não são ótimas, resultando em soluções de menor qualidade

Assim, podemos concluir que:

- Alta mutação destrói boas soluções;
- Baixa mutação refina gradualmente as soluções e permite a procura local de soluções ótimas;
- Elitismo preserva boas soluções e permite a convergência mais rápido.

Sendo assim, pelas razões apresentadas iremos fixar os parâmetros da experiência 3 e explorar uma nova função de aptidão.

REFERÊNCIAS

- [1] [Geek for Geeks Website. Mutation Algorithms for Real-Valued Parameters \(GA\)](#)