



UNIVERSIDADE D
COIMBRA

Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Licenciatura em Engenharia Informática

Trabalho Prático Nº2

Rolling in the Hill: Evolutionary Edition

Francisco Mendes | uc2019222823@student.uc.pt | 2019222823 | PL2

Leonardo Pereira | uc2020239125@student.uc.pt | 2020239125 | PL2

Tiago Henriques | uc2020237060@student.uc.pt | 2020237060 | PL2

Fundamentos de Inteligência Artificial

2022/2023 – 2º Semestre

Índice

1. INTRODUÇÃO	2
2. IMPLEMENTAÇÃO	3
2.1 META 1 – MODELAÇÃO E DESENVOLVIMENTO DO ALGORITMO GENÉRICO.....	3
2.2 META 2 – EXPERIMENTAÇÃO E ANÁLISE.....	3
2.2.1 <i>Cenário 1: Gap Road</i>	4
2.2.2 <i>Cenário 2: Hill Road</i>	7
3. CONCLUSÃO	8

1. Introdução

A partir deste trabalho prático, irão ser estudados Algoritmos Evolucionários (AE), que são modelos computacionais inspirados na Teoria da Seleção Natural de Charles Darwin, e nos princípios de herança genética de Gregor Mendel, métodos estes que têm sido aplicados com sucesso em diversas áreas da tecnologia.

O objetivo passa pela implementação das componentes necessárias de um Algoritmo Evolucionário, de modo que este seja capaz de desenhar veículos motorizados cuja missão seja chegar o mais longe possível no percurso onde se encontram, adaptando-se à medida que vão encontrando novos obstáculos.

Cada veículo é constituído por um conjunto de formas fechadas compostas por vértices e arestas, sendo que os vértices podem ter rodas de diferentes tamanhos.

2. Implementação

2.1 Meta 1 – Modelação e desenvolvimento do Algoritmo Genérico

Para esta meta, o propósito foi adaptar e preencher as linhas de código em falta nos scripts dos quatro operadores de modo que estes ficassem funcionais. Neste projeto dá-se uso a quatro operadores:

- **Recombinação - Uniform Crossover:** Operador utilizado durante o processo de geração de novos veículos descendentes de indivíduos pais. A ideia principal por trás do uniform crossover é criar uma solução que herda propriedades de diferentes soluções passadas de modo a tentar encontrar uma solução melhor para o problema apresentado.
- **Mutação - Gaussian Mutation:** Operador responsável por introduzir mudanças aleatórias nos veículos. Para tal, adiciona a cada gene um valor aleatório com base em uma distribuição gaussiana.
- **Seleção de Pais - Rolette:** Operador cujo objetivo é determinar quais são os indivíduos da geração que devem ser selecionados para serem os progenitores da próxima geração. Para realizar esta escolha, é atribuído a cada veículo uma probabilidade de ser escolhido, proporcional ao seu valor de fitness. Após cada veículo ter a sua percentagem são escolhidos os progenitores de forma aleatória.
- **Seleção de Sobreviventes - Elitism:** Operador encarregue de garantir que os X indivíduos mais aptos da geração anterior sejam preservados para a próxima.

Na parametrização são especificados os parâmetros do algoritmo evolucionário. Para este trabalho prático foram usados os seguintes parâmetros:

- Probabilidade de mutação por gene
- Probabilidade de recombinação
- Tamanho do Elitismo
- Número de Gerações

Estas componentes servem para controlar a forma como o algoritmo evolucionário se irá comportar.

2.2 Meta 2 – Experimentação e análise

Depois da implementação dos quatro operadores, o objetivo da meta 2 é usar as suas funcionalidades de maneira a encontrar a melhor solução possível para cada um dos casos disponibilizados. Para tal, é necessário realizar experiências com diferentes

valores e o desenvolvimento de uma função de fitness que permitisse o AE gerar um veículo capaz de atingir o final do percurso.

No que toca a função de aptidão, fitness, esta existe de forma a representar os requisitos que a população deve seguir, sendo que se trata da parte mais crítica dos algoritmos genéticos. Através dela, cada indivíduo da população irá ser avaliado de forma a saber a sua posição em relação à solução desejada.

Aquando da avaliação de um indivíduo, obtém-se um valor que corresponde à qualidade do mesmo. Quanto melhor o seu desempenho, maior é a sua qualidade.

É de extrema importância ter esta função bem definida, de forma a não acontecerem exclusões erradas.

2.2.1 Cenário 1: Gap Road

Neste primeiro cenário dá-se uso aos parâmetros existentes na Tabela 1 para realizar quatro experiências e analisar os seus resultados.

	Mutação	Elitismo	Crossover	Número Gerações
Experiência 1	0.05	0	0.9	30
Experiência 2	0.2			
Experiência 3	0.05	2		
Experiência 4	0.2			
Experiência 5	0.05			

Tabela 1 NOTA: Desnecessário fazer a experiência 5 pois esta é a mesma que a experiência 3.

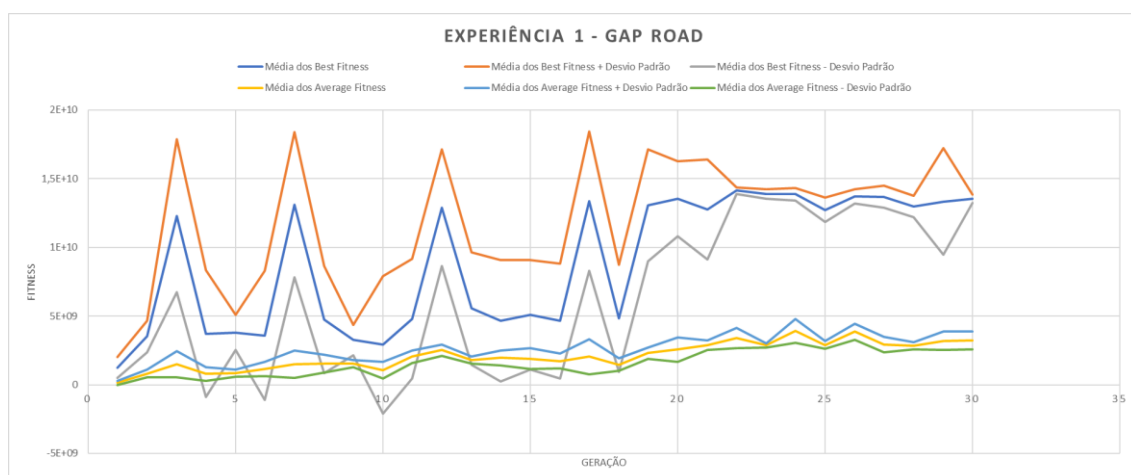
Após terem sido testadas diversas combinações, chegou-se à seguinte função de fitness, que provou ser mais eficiente face aos obstáculos postos à frente dos veículos:

$$\text{fitness} = (\text{MaxDistance} * 2.5 + \text{MaxVelocity} * 20 + \text{IsRoadComplete} * 500) ^ 3$$

Explicando agora esta escolha, os parâmetros que foram tidos em consideração para calcular a função de fitness foram: o MaxDistance (distância máxima percorrida pelo veículo), o MaxVelocity (velocidade máxima atingida pelo veículo) e o IsRoadComplete (1 ou 0 dependendo se o carro completou o trajeto ou não, respetivamente).

Esta função de fitness prioriza a maior distância percorrida e a maior velocidade atingida porque são ambos fatores importantes que necessitam de ser melhorados para que as próximas gerações consigam chegar mais longe, sendo que quanto maior for a distância e velocidade máximas, maior será a probabilidade de um agente chegar ao fim do percurso. Acima de tudo, a função prioriza veículos que já tenham sido capazes de percorrer o percurso, pois estes já têm as capacidades necessárias para chegar à meta e é crucial que passem os seus atributos para as gerações futuras.

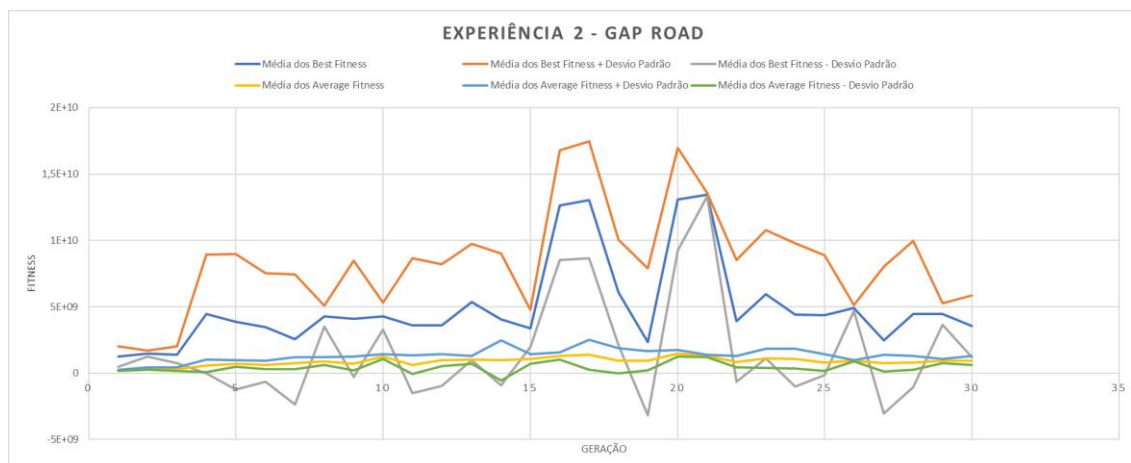
Experiência 1



Com base neste primeiro gráfico, a falta de elitismo é um dos destaques, pois não existem pontos de referência para gerações futuras, o que irá prejudicar o algoritmo a manter a estabilidade enquanto estiver a procurar veículos que terminem o percurso, ou seja, acontecer algum tipo de evolução de geração para geração.

Comprova-se então que esta experiência não seria a mais aconselhada de modo a atingir o objetivo pretendido.

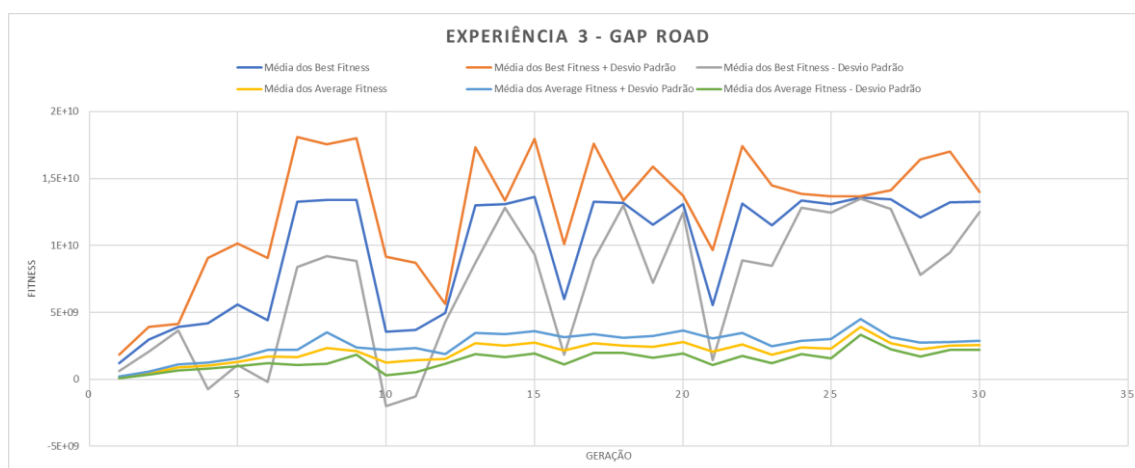
Experiência 2



Falando agora da experiência 2, também é possível observar uma grande instabilidade nos resultados. Apesar disso, existe um menor número de agentes a concluir o percurso, o que poderá dever-se ao facto de haver uma maior probabilidade de mutação, que faz com que os carros se alterem mais e não herdem tantas características.

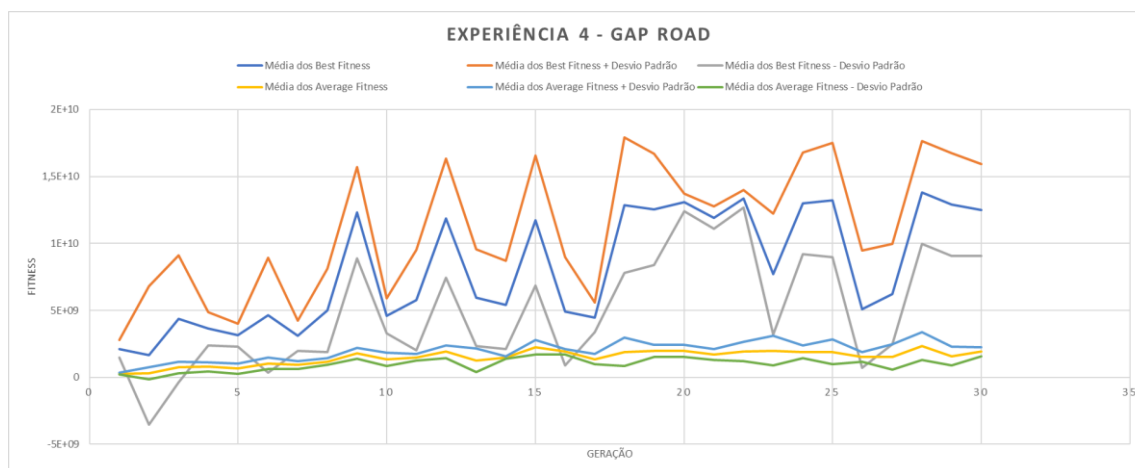
Concluindo, a experiência 1 tem melhores resultados que a experiência 2.

Experiência 3



No que concerne a experiência 3, é notória a maior estabilidade na evolução dos agentes, sendo que numa fase inicial, procura-se um veículo que consiga terminar o percurso, melhorando gradualmente a sua aptidão ao longo das gerações. Esta melhoria é consequência de existir elitismo, o que comprova que ao termos disponíveis a escolha dos pais com maior aptidão, é possível obter gerações cada vez mais eficientes. Quando um agente termina o percurso, as gerações seguintes também o fazem.

Experiência 4



Por último, na experiência 4, como a probabilidade de mutação foi alterada de 0.05 para 0.2, as alterações aos descendentes são mais significativas, o que irá prevenir o algoritmo genético de convergir para uma solução com maior rapidez, o que vai contra o objetivo que está em cima da mesa. Deste modo, observa-se um aumento no número de gerações necessárias para atingir uma solução.

Em suma, pode-se dizer que a experiência 3 é, no geral, aquela que apresentou melhores resultados. Logo, será esta a que possui o melhor conjunto de parâmetros.

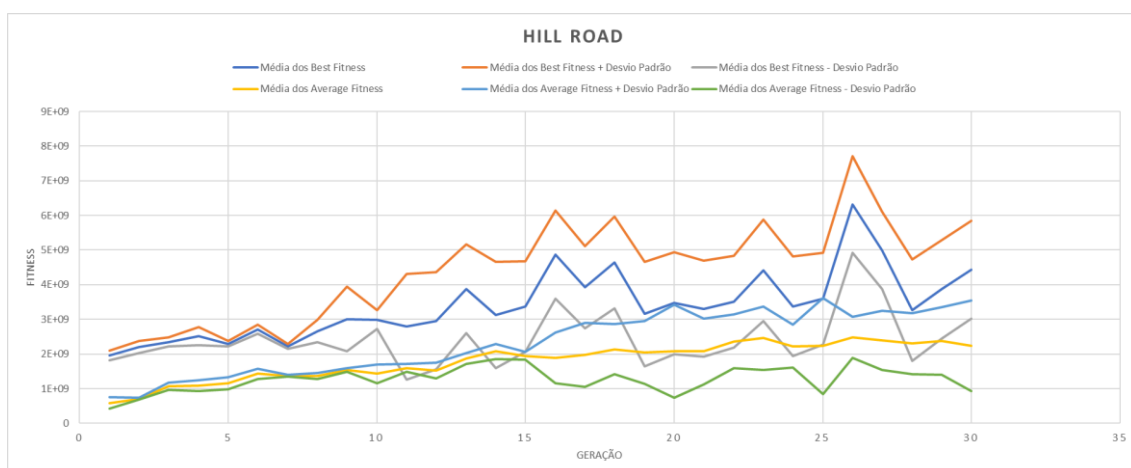
2.2.2 Cenário 2: Hill Road

Seguindo os resultados obtidos até ao momento, como foi a experiência 3 que proporcionou os melhores resultados no primeiro cenário, decidimos usar os valores dessa mesma experiência para os parâmetros de mutação e elitismo, agora no cenário Hilarodo.

Após a análise de várias opções, a seguinte função de fitness mostrou ser a mais eficiente para este cenário:

$$\text{fitness} = (1 / \text{CarMass} * 10000) ^ 2 + \text{MaxDistance} ^ 3 + (\text{MaxVelocity} * 50) ^ 3$$

Esta função tem em consideração a massa do carro, a máxima distância percorrida e a maior velocidade atingida. Ou seja, a aptidão de um carro é maior quanto mais longe for a distância percorrida, e, mais importante ainda, quanto mais leve e rápido ele for (características importantes de passar às próximas gerações, pois se forem continuamente melhoradas asseguram que os futuros carros serão capazes de acabar o percurso).



A partir destes dados, e tendo em conta a natureza estocástica do algoritmo, conclui-se que a função de fitness tem um impacto baixo na obtenção de bons resultados no cenário HillRoad, uma vez que, se os carros gerados aleatoriamente através dos pais não forem bons, não há avanços significativos ao longo das gerações, o que não permite chegar à meta em apenas trinta gerações.

Não iria existir então uma função de fitness que se enquadra de forma perfeita e que leve sempre ao resultado pretendido.

3. Conclusão

Ao longo do desenvolvimento deste trabalho prático, foram adquiridos conhecimentos e aprendizagens sobre algoritmos evolucionários e o seu funcionamento, tendo especial foco nos seus parâmetros, sendo eles mutação, elitismo, recombinação, entre outros.

Através deste algoritmo, desenvolveram-se competências sobre formulação de funções de aptidão (fitness), seja na atribuição de diferentes pesos para os componentes utilizados, ou que componentes escolher e como influenciam a evolução dos veículos.

Conclui-se também o facto do algoritmo ter uma natureza estocástica, pelo que foram necessárias diversas experiências de forma a chegar aos dados mais eficientes.

Por fim, durante as experiências dos cenários em questão, também se pôde notar que por vezes, um agente encontra-se muito dependente da sua forma, até mais do que as próprias características atribuídas, de forma a poder completar os percursos.