
PROJETO – OTIMIZAÇÃO DE ESPAÇO

1. Descrição do problema

Existem diversos processos industriais que se baseiam na distribuição, e posterior corte, de várias peças de forma otimizada numa superfície de matéria-prima. São exemplo disso as indústrias têxteis, onde é importante otimizar a disposição das peças na área de tecido a utilizar, a indústria metalomecânica, em chapas metálicas, a construção civil, ou mesmo no desenho e desenvolvimento de circuitos digitais. No caso da indústria têxtil, por exemplo, cada produto a fabricar (vestuário) é construído por diversos componentes, ou peças, recortados numa grande área de tecido. Depois de realizado o corte, todos os restos de tecido cortado são desperdiçados. Obviamente, para qualquer indústria a minimização de desperdícios é um aspeto fulcral. O problema é o mesmo com que nos deparamos em crianças quando queremos cortar e colar papel, sem o desperdiçar. Uma disposição ineficiente implica o desperdício de matéria-prima por resultar em restos que em virtude da sua forma e/ou tamanho não podem ser aproveitados.

O objetivo deste projeto é o desenvolvimento de uma aplicação que permita otimizar a disposição de peças simples numa superfície de modo a minimizar os recursos necessários ao seu fabrico.

Iremos considerar que, tanto a superfície correspondente à matéria-prima, como as peças a fabricar, são definidas com base em células quadrangulares. Assim, o problema define-se formalmente do seguinte modo:

Dados,

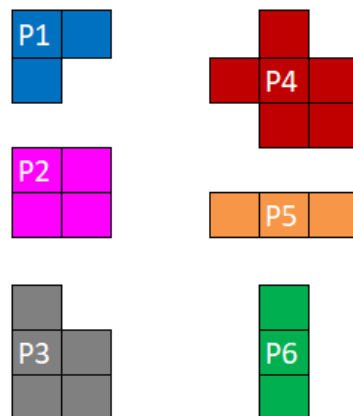
- uma superfície com uma altura de A células e um comprimento ilimitado (como se fosse um rolo sem fim);
- um conjunto de peças, sendo cada uma representada por uma matriz binária cujas dimensões dependem das dimensões da peça (considere-se que os 1s representam a peça);

pretende-se dispor espacialmente as peças na superfície de modo a otimizar a área e energia necessárias para cortar as peças sem que haja sobreposições. A solução para um problema concreto deve consistir na posição de cada peça na superfície.

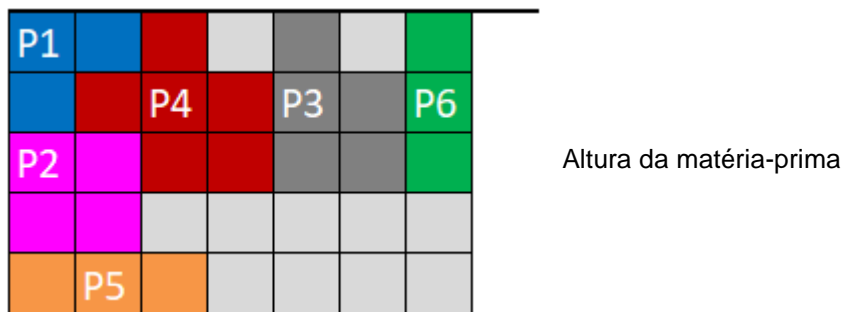
Quanto ao cálculo da área da matéria-prima necessária, vamos defini-la como sendo igual à área do retângulo mínimo capaz de envolver todas as peças, tal como ilustrado no exemplo abaixo. Ou seja, toda a área não aproveitada do retângulo envolvente é considerado como desperdício. A disposição ótima será aquela que, simplesmente, não desperdiça qualquer matéria-prima, isto é, aquela em que a área de matéria-prima necessária ao fabrico é igual à soma das áreas das peças.

Exemplo

Consideremos que se pretende fabricar o seguinte conjunto de 6 peças,



e que a matéria-prima tem uma altura de 5 células. Um possível plano de corte seria:



A área de matéria-prima necessária para o fabrico é a definida pelo retângulo a cinzento-claro (que engloba as peças). Note-se que este poderia não ser o plano ideal para este problema em particular, nomeadamente, porque as zonas a cinzento-claro seriam desperdiçadas.

O programa deve permitir que o utilizador escolha o problema a resolver. Cada problema é guardado num ficheiro de texto. A figura abaixo mostra o conteúdo do ficheiro para o exemplo ilustrado acima.

```

5 6
2 2 1 1 1 0
2 2 1 1 1 1
3 2 1 0 1 1 1 1
3 3 0 1 0 1 1 1 0 1 1
1 3 1 1 1
3 1 1 1 1

```

A primeira linha indica a altura da matéria-prima (5) e o número de peças (6). Cada uma das linhas restantes corresponde a uma peça. O primeiro e segundo valores são, respetivamente, o número de linhas e colunas da matriz necessários para definir a peça. Os restantes valores correspondem ao valor das células da matriz: 1 se a célula correspondente está preenchida e 0 em caso contrário.

2. Trabalho a realizar

O objetivo principal deste projeto é o desenvolvimento de uma aplicação capaz de resolver o problema descrito acima. Para isso, devem ser utilizados os seguintes algoritmos:

- 1) Algoritmo aleatório: este algoritmo deve gerar soluções aleatórias para o problema e no final devolver a melhor que encontrou;
- 2) Algoritmo genético.

Para cada algoritmo, deve ser possível o utilizador visualizar textualmente a melhor solução encontrada até ao momento durante a execução de cada algoritmo.

O projeto fornecido com este enunciado é baseado no projeto desenvolvido nas aulas e já contém alguns aspetos implementados como sejam a leitura dos ficheiros de texto com os problemas (datasets) e a GUI da aplicação, e outros parcialmente implementados, como sejam o algoritmo genético.

O projeto consiste na realização das tarefas seguintes:

1. Desenvolvimento de classe que representa o problema (classe `stockingproblem.StockingProblem`);
2. Desenvolvimento da classe que permite representar uma solução para o problema (classe `stockingproblem.StockingProblemIndividual`);
3. Desenvolvimento de dois operadores de recombinação adequados ao problema e à representação utilizada para as soluções (classes `ga.geneticoperators.Recombination2` e `ga.geneticoperators.Recombination3`); Nota: já é disponibilizado um operador de recombinação (classe `ga.geneticoperators.RecombinationPartialMapped`).
4. Desenvolvimento de dois operadores de mutação adequados ao problema e à representação utilizada para as soluções (classes `ga.geneticoperators.Mutation2` e `ga.geneticoperators.Mutation3`); Nota: já é disponibilizado um operador de mutação (classe `ga.geneticoperators.MutationInsert`).
5. Desenvolvimento do algoritmo aleatório (classe `random.RandomAlgorithm`);
6. Realização de testes que permitam estudar os seguintes aspetos:
 - O efeito da variação do tamanho da população e do número de gerações no algoritmo genético;

- Desempenho relativo dos diferentes operadores genéticos e das respectivas probabilidades de ocorrência;
- O efeito da variação do tamanho do torneio;
- O desempenho relativo dos diferentes algoritmos.

Do relatório do projeto deve constar:

- A descrição da representação das soluções;
- A descrição da função de avaliação utilizada;
- A descrição dos operadores genéticos desenvolvidos;
- A descrição da forma como foi implementado o algoritmo aleatório;
- A apresentação e discussão dos resultados obtidos nos testes realizados;
- Outros aspetos considerados relevantes para uma boa compreensão e avaliação do trabalho realizado.

Alguns dos fatores com mais importância na avaliação do relatório são:

- Clareza na descrição dos componentes da aplicação;
- A forma como os resultados dos testes são compilados e a clareza com que são apresentados (a utilização, mas não só, de tabelas e/ou gráficos pode ajudar);
- O significado estatístico dos resultados (que depende do número de *runs* realizados);
- A análise e discussão dos resultados.

3. Cotações

05% - Tarefa 1

35% - Tarefa 2

10% - Tarefas 3 e 4

05% - Tarefa 5

30% - Tarefa 6 e Relatório (devem ser entregues todos os ficheiros das experiências)

15% - Extras

Extras sugeridos:

- Implementar uma versão do problema em que as peças possam ser rodadas;
- Implementar um mecanismo de elitismo no algoritmo genético;
- Implementar o método de seleção por posição (*rank based selection*);
- Implementar um mecanismo de procura local no algoritmo genético;
- Implementar outros algoritmos (ex: *hill-climbing*, *simulated annealing*, *bees algorithm*, *particle swarm optimization*);
- Alterar o projeto fornecido de modo a que permita realizar experiências em *batch* com outros algoritmos que não apenas o algoritmo genético (nota: o projeto fornecido permite realizar experiências em *batch* apenas com o algoritmo genético).

4. Prazos, datas, regras e instruções

1. Data limite de entrega do projeto: **26 de junho de 2021, 23:59**.
2. Data da prova oral: **5 de julho de 2021**.
3. O projeto é realizado em grupos de 2 estudantes. Não são aceites projetos realizados por grupos com mais de 2 elementos. Os estudantes que pretendam realizar o projeto individualmente devem solicitá-lo, por escrito, ao docente responsável pela UC. Apenas em casos bem fundamentados serão autorizados projetos realizados individualmente.
4. O relatório deve ser realizado utilizando o modelo disponibilizado na secção *Projeto* do sítio da UC no Moodle.
5. O projeto deve ser entregue sob a forma de um arquivo zip, rar ou 7z que contenha todos os elementos do projeto, incluindo o relatório. O nome do arquivo deve ter o formato *IA_Projeto_#1_#2.(zip/rar/7z)*, onde #1 e #2 devem ser substituídos pelos números de estudante dos elementos do grupo. O relatório deve ser entregue em formato pdf e o seu nome deve ter a mesma estrutura do arquivo mas com extensão pdf.