



Universidade do Porto
Faculdade de Engenharia

FEUP

Otimização na Gestão de Projetos

Relatório Intercalar

Inteligência Artificial

3º ano do Mestrado Integrado em Engenharia Informática e Computação

Elementos do Grupo:

André Costa Moreira Maia – 201200674 – up201200674@fe.up.pt

Guilherme Vieira Pinto – 201305803 – up201305803@fe.up.pt

João Manuel Estrada Pereira Gouveia – 201303988 – up201303988@fe.up.pt

17 de abril de 2016

Objetivo

Neste trabalho teremos como intuito o desenvolvimento de uma aplicação, no âmbito da unidade curricular de Inteligência Artificial, para otimização do planeamento na gestão de projetos. Para tal, procuraremos associar, de forma otimizada, elementos a uma equipa de projetos, concebendo a melhor distribuição de dados para as variáveis apresentadas no contexto do problema. Tratando-se de um problema de otimização procuraremos encontrar várias soluções utilizando diferentes algoritmos de forma a podermos comparar a sua eficácia e determinar a melhor abordagem ao problema.

Descrição

Especificação

Tal como descrito anteriormente, trata-se de um problema de otimização de gestão de projetos, aos quais são associados elementos (trabalhadores) com diferentes competências, de forma a que cada tarefa seja cumprida no menor tempo possível.

Tendo em conta a complexidade da organização de recursos orientados à gestão de projetos coordenados, temos como objetivo, o *output* de um valor ótimo para o problema.

Deste modo, a alocação dos elementos mais apropriados para um dado projeto é o essencial ao sucesso dos resultados pretendidos.

Podemos, então, definir um Projeto como sendo constituído por um conjunto de Tarefas a serem concluídas por um ou mais Trabalhadores. As Tarefas podem ter precedências entre si e apresentam uma duração Trabalhador/mês. Cada tarefa não pode ser iniciada sem a totalidade das suas precedências estar completa.

Cada Trabalhador apresenta um conjunto de Competências que cobrem certas Tarefas relativas a essas áreas. Deve-se também ter em atenção, o desempenho relativo que cada Trabalhador tem sobre uma determinada Competência, de forma a minimizar a duração de cada tarefa.

Para a implementação deste projeto, recorreremos à linguagem Java que, para além de nos facultar a documentação e estruturas de dados mais importantes para a conceção do código, também suporta, de forma consistente e relativamente simples, a criação de uma interface gráfica importante para a interação entre o utilizador e o programa.

Estruturação/Planeamento do trabalho

Avaliando o objetivo do trabalho e tendo em conta a sua complexidade, o grupo delineou o seguinte plano:

- Recolha de exemplos de teste suficientemente complexos, e respetivas soluções, para comparação com resultados obtidos na execução dos algoritmos desenvolvidos;
- Estudo sobre a melhor estruturação dos cromossomas a serem processados no algoritmo genético a utilizar, tendo em conta as precedências de tarefas e distribuição dos trabalhadores segundo aptidões e respetivas performances;
- Implementação generalizada de um algoritmo genético para futura adaptação ao problema em questão;
- Implementação de uma função de adaptação (avaliação de *fitness* dos cromossomas) adequada ao caso a ser abordado, de modo a ser incorporada no algoritmo genético;
- Criação de uma interface gráfica (GUI) devidamente adaptada a receber os *inputs* fornecidos pelo utilizador e apresentando de forma simples e legível os *outputs* resultantes do algoritmo executado;
- Tendo os pontos anteriores concluídos, o grupo focar-se-á na melhoria dos algoritmos, tempos de execução ou aprimoramento da GUI.

Modelos de Representação

O tempo de execução (expresso em Trabalhadores/Mês) descreve o grau de adaptação de um dado cromossoma ao problema. No entanto, dados dois Trabalhadores com a mesma aptidão mas performances diferentes, não podemos considerar que ambos executem uma dada Tarefa com a mesma eficácia. Assim sendo, reformularemos a expressão de cálculo do tempo de execução de uma Tarefa:

$$T_{exec} = (\sum_i^n P_i) / Mes$$

Na expressão fornecida, limitamos que uma Tarefa seja concluída num mês apenas quando o somatório das performances de todos os Trabalhadores a ela associados seja igual a 100% (considerando que as performances são respetivas à aptidão da Tarefa em questão, variando de 0 a 100%). Por sua vez, se esse somatório resultasse em 50%, a Tarefa levaria dois meses até ser terminada.

O segundo dado com o qual o grupo se tem vindo a debater é a estruturação dos cromossomas. Não consideramos simples a organização de uma lista binária que satisfaça as seguintes variáveis:

- Tempo de execução de uma Tarefa;
- Distribuição de Trabalhadores por cada Tarefa, sem ocorrência de atribuição do mesmo individuo a duas Tarefas em simultâneo;
- Análise da Performance dos Trabalhadores destacados para uma dada Tarefa.

Como tal, nesta fase, assumimos que ainda não somos capazes de estabelecer uma estrutura fiável ou plausível à adaptação do problema.



No esquema apresentado está representado um possível desenho de um cromossoma a ser utilizado. Cada gene no cromossoma simboliza uma Tarefa e o valor INTEGER(X) nela contido referencia o identificador do grupo de Trabalhadores a ela associados.

Por fim, a transformação destes cromossomas, de geração para geração, será realizada com o auxilio de uma função de adaptação baseada no Algoritmo de Arrefecimento Simulado. Com este método, será possível mutar os cromossomas de modo a eleger combinações que se aproximem de um resultado que satisfaça o pretendido e, por isso, não garantindo que seja a solução ideal.

A ordenação destes mesmo genes estabelece também a respetiva execução, pelo que permitirá avaliar se as precedências de cada uma foi tida em conta.

Trabalho Realizado

No momento de submissão deste mesmo documento, o grupo tem implementada uma versão simples e generalizada de um algoritmo genético e de um algoritmo de Arrefecimento Simulado para posterior manipulação e adequação ao problema.

Apresentamos também um conjunto de classes complementares ao projeto que armazenam informação indispensável para gestão de Tarefas, Trabalhadores e Competências e respectivas Performances.

Resultados esperados e forma de avaliação

De modo a validar os resultados obtidos, após a execução dos algoritmos com os *inputs* fornecidos, procuraremos enriquecer a aplicação com testes unitários que nos garantam que um determinado não esteja associado a duas Tarefas em simultâneo e que a execução das precedências de cada Tarefa é tida em conta.

O *output* do programa também permitirá que sejam consultados os dados de cada Trabalhador numa determinada Tarefa para que se determine se possui as aptidões necessárias para a executar, assim como verificação do tempo de execução da Tarefa tendo em conta as performances dos elementos a ela atribuídos.

Conclusão

Embora o grupo se tenha interessado desde o início por eleger este problema como principal escolha, admitimos que o tínhamos considerado mais simples do que realmente ele é. Contudo, algoritmos genéticos são métodos extremamente cativantes e repletos de potencialidade, que adaptam os conhecimentos de ramos científicos distintos ao que até agora abordámos no curso para conceber resultados satisfatórios para problemas complexos com que nos deparamos.

Até esta primeira etapa, tem sido um ótimo desafio colocar as nossas capacidades à prova na conceção de uma resolução adequada ao projeto. Esperemos conseguir alcançar os objetivos pretendidos e apresentar uma aplicação robusta e eficaz no cálculo das soluções pretendidas.

Recursos Utilizados

- Apontamentos das aulas teóricas;
- “A Genetic Algorithm for Resource-Constrained Scheduling” by Matthew Bartschi Wall;
- Eclipse (<https://eclipse.org/>);
- GitHub (<https://github.com/>);
- Microsoft Office Word 2016 (<https://products.office.com/pt-pt/home>).