

UNIVERSIDADE FEDERAL FLUMINENSE
INSTITUTO DE CIÊNCIA E TECNOLOGIA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Lucas Campos Dal Piaz de Souza

**ALGORITMO GENÉTICO COM CHAVES
ALEATÓRIAS PARA O PROBLEMA DE
LOCALIZAÇÃO DE PLATAFORMAS DE
PETRÓLEO**

Rio das Ostras-RJ

Julho, 2017

LUCAS CAMPOS DAL PIAZ DE SOUZA

**ALGORITMO GENÉTICO COM CHAVES ALEATÓRIAS PARA O PROBLEMA DE
LOCALIZAÇÃO DE PLATAFORMAS DE PETRÓLEO**

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação do Instituto de Ciência e Tecnologia da Universidade Federal Fluminense, como requisito parcial para obtenção do Grau de Bacharel. Área de Concentração: Otimização Combinatória.

Orientador: Prof. Dr. Dalessandro Soares Vianna

Rio das Ostras-RJ

2017

LUCAS CAMPOS DAL PIAZ DE SOUZA

**ALGORITMO GENÉTICO COM CHAVES ALEATÓRIAS PARA O PROBLEMA DE
LOCALIZAÇÃO DE PLATAFORMAS DE PETRÓLEO.**

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação do Instituto de Ciência e Tecnologia da Universidade Federal Fluminense, como requisito parcial para obtenção do Grau de Bacharel. Área de Concentração: Otimização Combinatória.

Aprovada em julho de 2017.

BANCA EXAMINADORA

Prof. Dr. DALESSANDRO SOARES VIANNA - Orientador

UFF

Profa. Dra. MARCILENE DE FÁTIMA DIANIN VIANNA

UFF

Prof. EDUARDO MARQUES

UFF

Rio das Ostras-RJ

2017

A Deus por ter me dado saúde e força para superar as dificuldades. A minha avó, Maria de Fátima. Ao meu avô, Antônio (*In Memoriam*). Aos meus pais, Carla e José Geraldo. A Fernanda Novaes. Aos familiares e amigos.

Agradecimentos

A Deus, meu sublime e extraordinário Mestre, minha fonte de vida, quem guia meus passos e me dá forças para prosseguir nessa caminhada. Agradeço pela oportunidade que me foi concedida.

A minha avó, Maria de Fátima, meu amor perene, heroína que me deu apoio. Ao meu avô, Antônio (*In Memoriam*), meu grande amigo e parceiro. Ambos, sempre me ampararam e creram em mim, incentivo nas horas difíceis, de desânimo e cansaço.

Aos meus pais, Carla e José Geraldo, que me apoiaram nos estudos e me aconselharam, sempre com paciência e amor.

A Fernanda Novaes, incentivadora, companheira e amiga.

A minha família e amigos, pelo amor, incentivo e apoio incondicional.

Ao meu orientador, Dalessandro Vianna, pelo suporte no pouco tempo que lhe coube, pelas suas correções e incentivos.

A esta universidade, seu corpo docente, direção e administração.

E a todos que direta ou indiretamente fizeram parte da minha formação, o meu muito obrigado.

Lista de Símbolos

AG Algoritmo Genético

ANM Árvores de Natal Molhadas

LDA lâmina d'água

RKGA *Random Key Genetic Algorithms*

SSP Sistema Submarino de produção

UEP Unidade Estacionária de Produção

Lista de Figuras

1.1	Sistema Submarino de Produção. Fonte: Adaptado de Morais (2013).	2
2.1	As três fases e suas relações de dependência. Fonte: Adaptado de Franco (2003).	6
2.2	a) Parâmetros para determinar o tipo arranjo dos poços. b) Parâmetros para determinar o tipo de <i>manifold</i> . Fonte: Adaptado de Franco (2003).	6
2.3	Tipos de poços. Fonte: Adaptado de PETROBRAS (2017).	7
2.4	a) Parâmetros para determinar o tipo de UEP. b) Parâmetros para determinar o tipo de ancoragem. c) Parâmetros para determinar tipo de <i>riser</i> . Fonte: Adaptado de Franco (2003).	8
3.1	Diagrama de Classes.	15
3.2	RKGA antes da iteração 1.	17
3.3	RKGA iteração 1.	18
3.4	RKGA iteração 2.	18
3.5	RKGA iteração final.	19
3.6	Ponto de corte variável.	20
4.1	Representação da população da instância Entrada1.txt.	23
4.2	Representação da população da instância Entrada2.txt.	23
4.3	Representação da população da instância Entrada3.txt.	24
4.4	Representação da população da instância Entrada4.txt.	25
4.5	Representação da população da instância Entrada5.txt.	25

Lista de Tabelas

4.1	Instâncias e entradas	22
4.2	Entrada1.txt.	22
4.3	Entrada2.txt.	23
4.4	Entrada3.txt.	24
4.5	Entrada4.txt.	24
4.6	Entrada5.txt.	25

Sumário

Agradecimentos	v
Lista de Símbolos	vi
Lista de Figuras	vii
Lista de Tabelas	viii
Resumo	xi
Abstract	xii
1 Introdução	1
1.1 Contextualização	1
1.2 Objetivos	3
1.2.1 Geral	3
1.2.2 Específicos	3
1.3 Justificativa	3
2 Referencial Teórico	5
2.1 Definições para um Sistema Marítimo de Produção	5
2.1.1 Primeira Fase - Arranjo dos Poços e <i>Manifold</i>	6
2.1.2 Segunda Fase - Unidade Estacionária de Produção, Ancoragem e <i>Riser</i>	8
2.1.3 Terceira fase - Tipo de transferência e estocagem de óleo	9
2.2 Algoritmo Genético	9
2.2.1 Terminologia	11
2.2.2 Representação da população	11
2.2.3 Avaliação da população	11
2.2.4 Seleção	12
2.2.5 <i>Crossover</i>	12
2.2.6 Mutação	13
2.2.7 Chaves aleatórias - RKGA	13

3	Metodologia Proposta	14
3.1	Paradigmas de programação	14
3.2	Algoritmo genético para o problema apresentado	15
3.2.1	Visão geral	15
3.2.2	População inicial	16
3.2.3	Avaliação da população	16
3.2.4	Seleção	19
3.2.5	<i>Crossover</i>	19
3.2.6	Mutação	20
4	Experimentos Realizados	21
4.1	Ambiente de desenvolvimento	21
4.2	Gerador	21
4.3	Resultados	22
5	Considerações Finais	27
5.1	Conclusões	27
5.2	Sugestões para trabalhos futuros	27

Resumo

O presente trabalho apresenta uma solução para o problema de locação das UEPs (Unidades Estacionárias de Produção) e *manifolds*. Os diversos fatores relacionados aos sistemas submarinos de produção implicam na necessidade de otimização. Quanto maior as distâncias entre poços, *manifolds* e UEPs, maior é o custo envolvido entre eles. O método proposto é baseado na metaheurística Algoritmos Genéticos (AG) associada ao conceito de chaves randômicas, ou como Bean (1994) propôs, um algoritmo genético com chaves aleatórias. A implementação é dividida em classes, sendo elas CProblema, CSolucao e CGenetico. A função de aptidão leva em conta as demandas atendidas e as capacidades envolvidas. Além destes fatores são relevantes os custos relacionados aos cabeamentos de *risers*, *flowlines* e *jumpers*. Para os experimentos, foram geradas cinco instâncias, as quais variam de acordo com número de poço, *manifolds* e UEPs. Para todas elas, é possível visualizar e comprovar que a melhora nos resultados está diretamente proporcional a quantidade de iterações. Por fim, é demonstrada a evolução da população através do gráfico de nuvem de pontos.

Palavras-chave: Algoritmo genético. Chaves aleatórias. Localização. Plataformas de petróleo. Otimização.

Abstract

The present work presents a solution for the problem of leasing UEPs and manifolds. The various factors related to underwater production systems imply the need for optimization. The greater the distances between wells, manifolds and UEPs, the greater the cost involved between them. The proposed method is based on the metaheuristic Genetic Algorithms (GA) associated with the concept of random keys, or as Bean (1994) proposed, random key genetic algorithms (RKGA). The implementation is divided into classes, which are: CProblema, CSolucao and CGenetico. The fitness function takes into account the demands met and the capacities involved. Besides these factors it is relevant the costs related to cabling of risers, flowlines and jumpers. For the experiments, five instances were generated, which vary according to number of wells, manifolds and UEPs. For all instances, it is possible to visualize and prove that the improvement in results is directly proportional to the number of iterations. Finally, the evolution of the population through the cloud chart of points is demonstrated.

Keywords: Genetic algorithm. Random keys. Location. Oil platforms. Optimization.

Capítulo 1

Introdução

1.1 Contextualização

No território brasileiro estão situadas 29 bacias sedimentares, do qual estima-se um potencial de exploração de petróleo variando entre 14 e 177 bilhões de barris. A Bacia de Campos é atualmente encarregada de 90% das reservas de petróleo e 47% das reservas de gás natural (EGLER; RIO, 2004 apud HORTA, 2002). A Bacia de Campos, posicionada na região norte do estado do Rio de Janeiro, compõe a principal área produtora de petróleo do país, abrange extensa área do mar territorial, uma influente zona costeira, com considerável valor ecológico, econômico e social, e vem apontando grande progresso em termos econômicos nos últimos anos (EGLER; RIO, 2004). O avanço tecnológico permitiu o vasto aumento do número de poços explorados em águas profundas e ultraprofundas. O campo de Enchova foi o primeiro explorado em 1977, com 124 m de profundidade. Atualmente a exploração no campo de Roncador está a 1886 m de profundidade (EGLER; RIO, 2004 apud ANP, 2003). Em seus estudos, EGLER e RIO (2004) afirma que a exploração da Bacia de Campos exige uma ampla infraestrutura de operações e instalações no mar e na terra.

De acordo com Morais (2013), as atividades de petróleo *offshore* dependem de permanentes pesquisas e inovações de processos. Alguns pontos que acabam gerando a necessidade de investimento em novas tecnologias são:

1. Condições relacionadas ao clima, no ambiente marinho e nas rochas que estão abaixo do leito;
2. As grandes distâncias entre as plataformas e o continente, e entre a plataforma e os poços localizados em áreas profundas;
3. A visibilidade é pior nas operações no mar.

Estas três especificidades tornam procedimentos operacionais nas atividades petrolíferas *offshore* dependentes do avanço tecnológico. Devem ser encontrados reservatórios de petróleo sob o leito marinho, para que em seguida seja produzido através de um Sistema Submarino de Produção (SSP).

Um Sistema Submarino de Produção é o conjunto de todos os equipamentos instalados no leito marinho. O principal propósito é a chegada de todos os fluídos na unidade de produção. Estes equipa-

mentos ajudam no escoamento dos fluídos e controlam a vazão, a pressão e a temperatura até a chegada da superfície. É importante garantir o funcionamento de todas as válvulas e sensores. Os principais equipamentos que compõem este conjunto de atividades são as Árvores de Natal Molhadas (ANM), as Linhas de Produção, os *Manifolds*, os Sistemas de Controle e as Unidades Estacionárias de Produção (UEPs) (THAMY, 2015). A Figura 1.1 ilustra tal sistema.

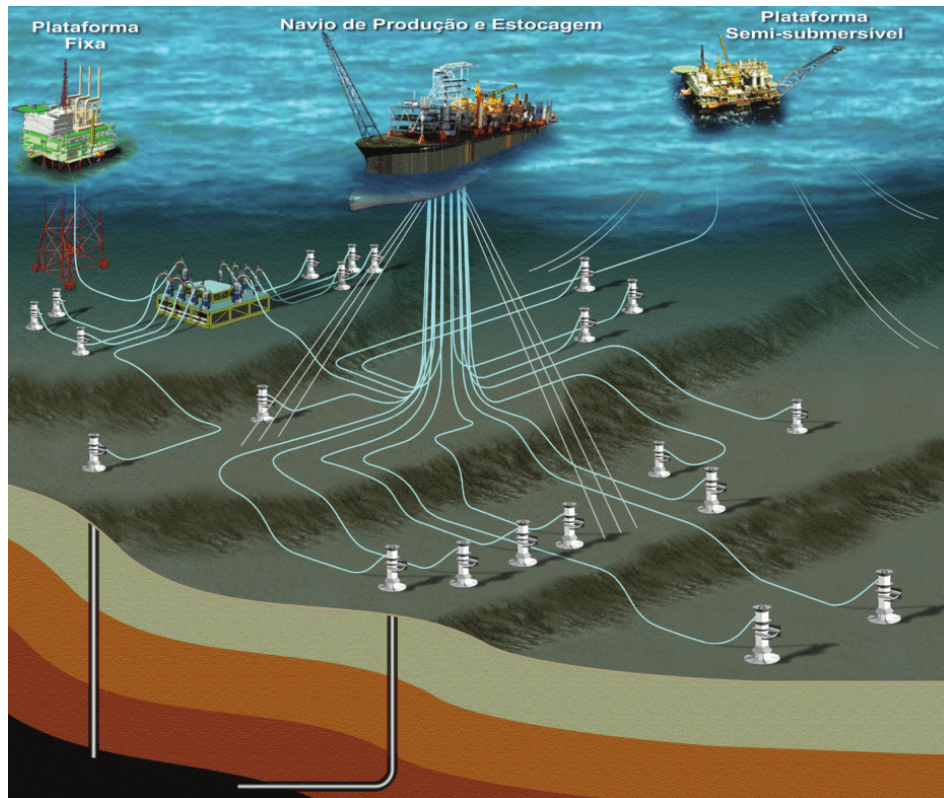


Figura 1.1: Sistema Submarino de Produção. Fonte: Adaptado de Moraes (2013).

A ANM fica na cabeça do poço, em águas profundas, e tem como função o controle da produção de hidrocarbonetos (FURTADO; FREITAS, 2009).

As Linhas de Produção são o conjunto de tubulações e interconexões que compõem o SSP, podendo ser flexíveis ou não. O escoamento do óleo produzido se dá através de *jumpers*, *flowlines* e *risers*. *Jumpers* e *flowlines* fazem a ligação do *manifold* ao poço ou a plataforma e não estão sujeitos à correntes e ações de ondas, sendo o *jumper* mais curto. Os *risers* ligam um *manifold*, por exemplo, a UEP e estão sujeitos à correntes, ações de ondas e outros movimentos condicionados no que se refere ao SSP (THAMY, 2015).

O *manifold* direciona, em uma única linha, a produção que vem de vários poços. Uma das vantagens de utilizar *manifolds* é a redução de *risers* com a UEP relacionada. Isto é um ótimo benefício levando em conta o alto custo dos *risers* (FERRARI, 2003).

O sistema de controle são cabos responsáveis por controlar equipamentos submarinos. Conexões elétricas e hidráulicas são feitas por este sistema. Um cabo deste tipo contém um conjunto de outros cabos que variam conforme a necessidade do projeto (THAMY, 2015).

As UEPs recebem e armazenam toda produção que foi escoada pelo SSP. O óleo e o gás são

transportados das UEPs para a costa. Existem diversos tipos de UEPs e a utilização de um determinado tipo varia conforme a necessidade e o projeto em questão (FERRARI, 2003).

O SSP e os equipamentos envolvidos implicam em gastos com manutenção, instalação, aquisição dos equipamentos, pesquisas, dentre outros. Além disso, o posicionamento das UEPs e dos *manifolds* é um ponto que deve ser analisado. O bom posicionamento de uma UEP pode gerar redução de gastos com equipamentos e manutenção destes. Os *risers* são fabricados a partir de muito estudo e análises. Como estes ficam submersos, é necessário que aguente a temperatura e a pressão, além de ter boa resistência e outras peculiaridades. Reduzir a quantidade de *risers* é garantia de redução de custo.

Além dos gastos envolvendo os equipamentos utilizados, deve ser considerado a questão da localização dos componentes do sistema. Projetos de SSP requisitam pontos estratégicos no que se refere aos melhores lugares possíveis para a instalação dos equipamentos, devido a possibilidade de vasta distância entre eles.

Diante desses cenários, problemas de localização e otimização surgem em todo o SSP. Minimizar custos e otimizar o projeto é, sem dúvida, relevante para o processo de extração *offshore*.

Ao invés de considerar todos os cenários possíveis com todas as restrições, é mais interessante focar em um problema mais específico que inclui a localização das UEPs e dos *manifolds*, com todos os parâmetros relevantes.

1.2 Objetivos

1.2.1 Geral

Desenvolver uma solução computacional para o melhor posicionamento das UEPs e *manifolds*.

1.2.2 Específicos

- Minimizar distâncias entre poços, *manifolds* e UEPs;
- Reduzir os custos com *risers*, *flowlines* e *jumpers*;
- Maximizar a produção, considerando capacidades e demandas;
- Oferecer uma solução computacional preenchendo lacunas na literatura.

1.3 Justificativa

Os diversos fatores relacionados aos SSPs implicam na necessidade de otimização. Na fase de planejamento a redução de custos envolve soluções computacionais para garantir a minimização do capital investido com respeito a todos os maquinários envolvidos.

Quanto maior a distância entre poços, *manifolds* e plataformas, maior é o custo envolvido entre eles. As ligações influenciam no comprimento de *risers*, *flowlines* e *jumpers*. As demandas e capacidades

são fatores que interferem em todo o processo de escolha de localização. Como todos os elementos já citados estão fortemente conectados, é indispensável o uso de mecanismos computacionais para averiguarem todas as combinações e particularidades.

Ademais, existe uma lacuna literária associada a soluções computacionais para a contextualização em questão e os conteúdos que são indispensáveis para a realização deste trabalho, fogem do escopo natural do curso de ciência da computação.

Capítulo 2

Referencial Teórico

Para que uma jazida de petróleo seja descoberta, um grande estudo deve ser feito e uma minuciosa análise de dados geofísicos e geológicos das bacias sedimentares deve ser levado em conta. Após a obtenção de resultados das camadas do subsolo, os geólogos e geofísicos decidem propor uma perfuração para aquele determinado local. Essa é uma fase que exige bastante investimento no processo de prospecção. (THOMAS, 2001).

Segundo Thomas (2001), a prospecção de petróleo, ou seja, detecção de reservas deste, foca em dois objetivos: (i) localizar situações geológicas que tenham boas condições para o acúmulo de petróleo; e (ii) verificar qual bacia sedimentar tem mais chances de conter petróleo. Não é possível uma previsão, mas sim uma análise envolvendo probabilidade.

A etapa seguinte é a realização da perfuração do poço. É extremamente importante deixá-lo em condições operáveis, levando em consideração os fatores segurança e economia. As operações destinadas a equipar o poço para produzir óleo ou gás são chamadas de completação. Na completação deve-se buscar a otimização da vazão de produção e minimizar possíveis manutenções nos poços (THOMAS, 2001).

2.1 Definições para um Sistema Marítimo de Produção

Franco (2003) propôs uma metodologia para auxiliar a escolha de um sistema para produção no mar. Esta metodologia é dividida em três fases. Na primeira fase serão determinados o arranjo dos poços e os *manifolds*. Na segunda fase serão determinados os tipos de unidade estacionária de produção, de ancoragem e de *riser*. Na terceira fase será determinado o tipo de estocagem e o tipo de sistema de transferência.

Existem relações de dependência entre as fases. Assim, para selecionar os tipos de UEP e ancoragem, determinados na segunda fase, é necessário que aconteça a seleção de arranjo dos poços inicialmente, na primeira fase (Figura 2.1).

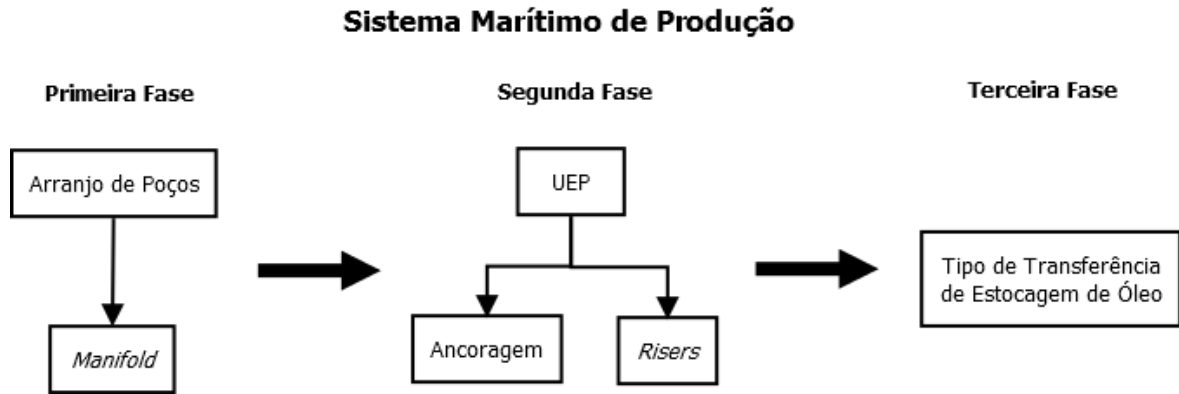


Figura 2.1: As três fases e suas relações de dependência. Fonte: Adaptado de Franco (2003).

As informações contidas nas Subseções 2.1.1, 2.1.2 e 2.1.3 tomaram como base o trabalho de Franco(2003). Franco (2003).

2.1.1 Primeira Fase - Arranjo dos Poços e *Manifold*

A primeira fase consiste em determinar o arranjo dos poços e na possível utilização do *manifold*.

Existem dois tipos de arranjo, o primeiro é chamado de satélite e o segundo de agrupamento de poços. Para determinar o arranjo dos poços, os seguintes parâmetros devem ser levados em conta: área do reservatório e sua profundidade, o tipo de poço e o número de poços. A Figura 2.2 ilustra o esquema da primeira fase.

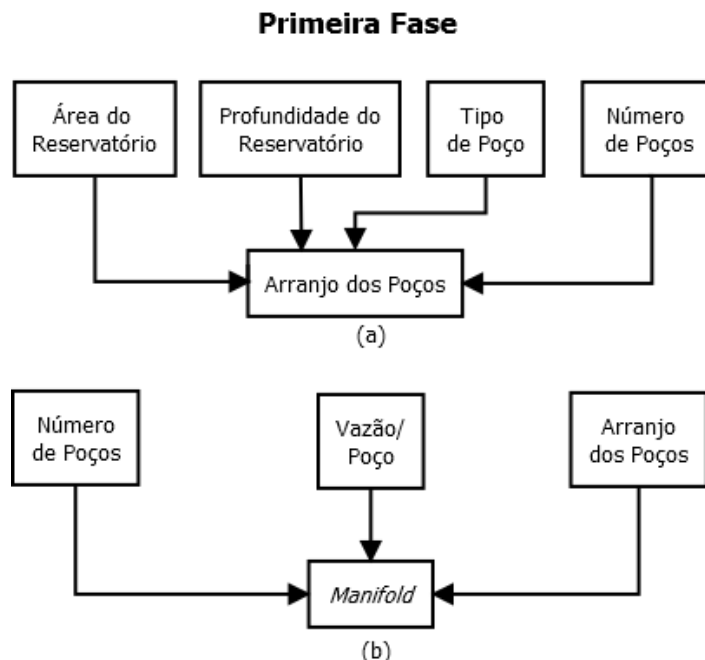


Figura 2.2: a) Parâmetros para determinar o tipo arranjo dos poços. b) Parâmetros para determinar o tipo de *manifold*. Fonte: Adaptado de Franco (2003).

Com respeito à área do reservatório, considera-se a área na direção horizontal medida em m^2 .

Caso essa área seja consideravelmente grande, o arranjo dos poços mais adequado deverá ser o satélite. Para uma área pequena, uma possível solução será a escolha de utilizar o agrupamento de poços. Outros fatores implicam na decisão do tipo de arranjo como a profundidade do reservatório, o número o tipo de poço.

A profundidade do reservatório é a distância vertical, em metros, que vai do fundo do mar até o local onde foi encontrado o petróleo. Quando o agrupamento de poços é o arranjo escolhido, a perfuração horizontal tende a ser mais favorável, pois neste caso existe mais chance de atingir pontos com maior acúmulo de óleo, não sendo isso possível com poços verticais. No caso de um reservatório não profundo, esse tipo de perfuração pode ultrapassar os limites laterais do poço.

Basicamente existem dois tipos de poços, os horizontais e os verticais (Figura 2.3). No contexto *offshore*, os poços horizontais são mais comuns. Um poço horizontal normalmente equivale a dois ou três poços verticais, no que se refere a produção e recuperação.

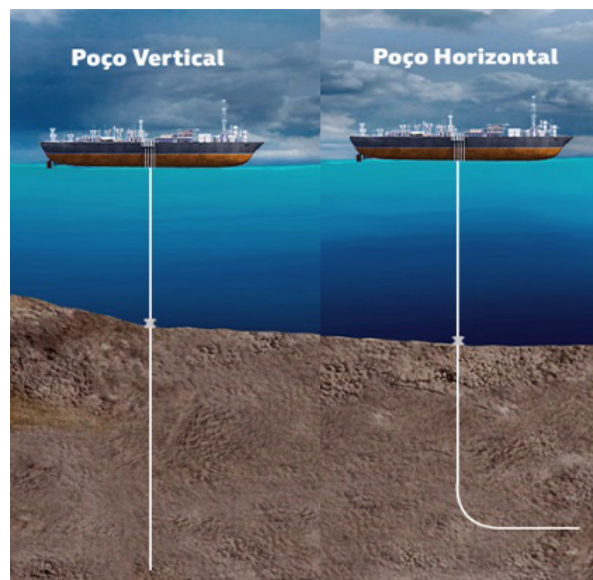


Figura 2.3: Tipos de poços. Fonte: Adaptado de PETROBRAS (2017).

Caso um poço seja perfurado horizontalmente, um possível arranjo será o agrupado, pois o ponto alvo será atingido. No entanto, se os poços forem verticais, no quesito recuperação de óleo, é mais interessante que seja o arranjo do tipo satélite podendo atingir vários locais do reservatório.

O número de poços pode ser definido como a soma de poços produtores e injetores perfurados no mesmo campo de petróleo.

Quanto maior a área, maior será a distância entre poços no caso de poços satélites. Contudo, tal parâmetro está sendo considerado de forma indireta no parâmetro área do reservatório.

Para determinar se o *manifold* será usado ou não, os parâmetros são o número de poços, a vazão diária por poço e o arranjo dos poços. É importante enfatizar que o arranjo dos poços agora é um parâmetro para a determinação do uso do *manifold*.

Quanto maior for a quantidade de poços, maior será a quantidade de *risers* necessária para atender cada um. O *manifold* reduz a quantidade de *risers* no projeto. Para o caso do arranjo ser do tipo

satélite, o comprimento de *flowlines* será grande também, para um número grande de poços. O custo destes equipamentos são bastante elevados, logo o uso do *manifold* poderá reduzir o número de *risers* e *flowlines*.

É possível definir a vazão diária por poço como o fluxo de óleo/gás escoado no período de um dia sendo medido em m^3/d . O *manifold* tem uma capacidade máxima para a realização da coleta do óleo produzido. Caso a vazão for acima da permitida, não é possível a realização da instalação do *manifold*.

2.1.2 Segunda Fase - Unidade Estacionária de Produção, Ancoragem e *Riser*

Para escolher a UEP adequada, os parâmetros utilizados são lâmina d'água, o arranjo dos poços, condições ambientais, número de poços e a vazão diária por poço. Na fase anterior, o arranjo de poços deve ter sido escolhido. Após escolha da UEP, será possível realizar escolhas relacionadas à ancoragem e *risers*. A figura 2.4 ilustra o esquema da segunda fase.

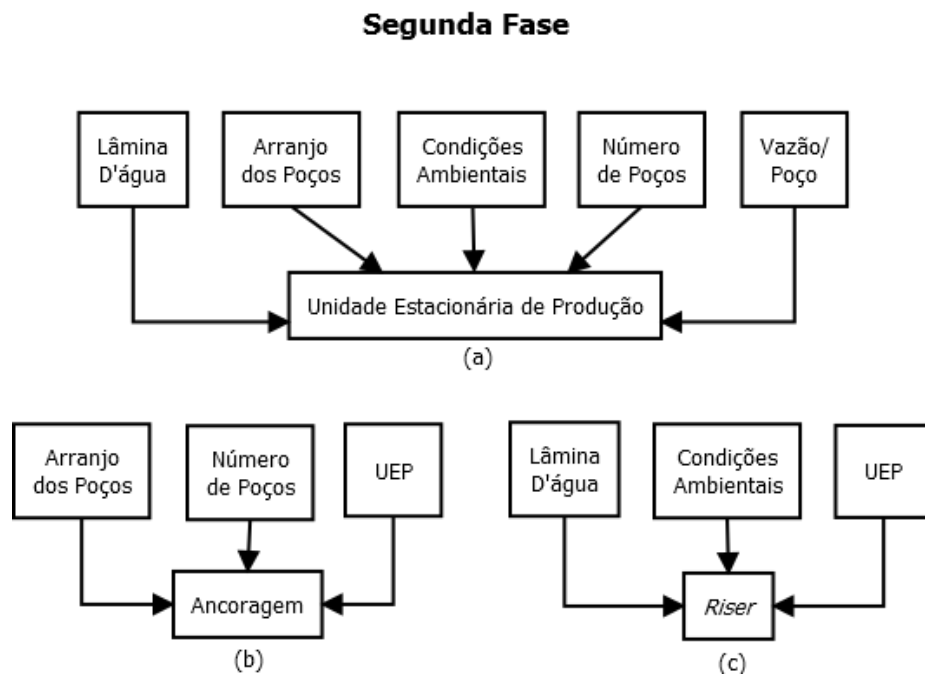


Figura 2.4: a) Parâmetros para determinar o tipo de UEP. b) Parâmetros para determinar o tipo de ancoragem. c) Parâmetros para determinar tipo de *riser*. Fonte: Adaptado de Franco (2003).

A classificação das UEPs depende da profundidade do leito oceânico e dos fatores ambientais. No entanto, de maneira geral, podem ser classificadas como fixas e ancoradas.

A lâmina d'água (LDA) é a distância da superfície até o fundo do mar e é medida em metros. As UEPs têm capacidade limitada para operar em certas profundidades. Como exemplo as plataformas fixas, as quais não podem ser utilizadas em LDA superior a aproximados 100 m.

O arranjo dos poços influencia na determinação da UEP, apenas alguns tipos são compatíveis com o sistema de agrupamento de poços.

Cada tipo de unidade estacionária pode operar com até um número máximo de poços.

Cada UEP tem restrição de vazão diária, devido a questão da capacidade máxima de processamento de petróleo de cada unidade estacionária de produção.

Depois de determinar qual o tipo da UEP, o próximo passo é decidir sobre a ancoragem e *riser*. Para realizar escolhas sobre a ancoragem é importante considerar os seguintes parâmetros: arranjo, número de poços e tipo da UEP.

Os parâmetros para a escolha do tipo de riser são lâmina d'água, as condições ambientais e a UEP. Todos mencionados nesta seção.

Em relação a lâmina d'água, quanto maior é a profundidade, maior é a pressão e a temperatura que serão exercidas no *riser*, isso pode prejudicar o processo de escoamento de óleo através do *riser*, principalmente se este for rígido.

Condições ambientais podem alterar os movimentos das UEPs. Esse movimento é transferido para os *risers*, impactando as UEPs diretamente, fazendo com que as operações fiquem prejudicadas. Este fator pode danificar os *risers* e causar acidentes ambientais e tecnológicos.

As movimentações em UEPs fixas são amenizadas, devido algumas das formas de propagação dos movimentos serem amenizados. Este fator influencia na ancoragem.

2.1.3 Terceira fase - Tipo de transferência e estocagem de óleo

Para escolher o tipo de transferência e estocagem de óleo, são adotados os parâmetros tipo de poço, vazão diária por poço, capacidade de armazenamento da unidade estacionária de produção, distância da costa e existência de infra-estrutura. Ademais, para o escopo do presente trabalho, o parâmetro "capacidade de armazenamento" será o adotado nessa fase.

2.2 Algoritmo Genético

Charles Darwin no final do século XIX, fundou a teoria da evolução, combinando a concepção de genética e seleção natural. A genética natural inclui a pluralidade entre indivíduos em uma população de organismos que se procriam. Esta variedade é feita pela combinação e pelo acréscimo de um novo material genético na população. O princípio de seleção natural favorece aos indivíduos mais aptos com maior probabilidade de reprodução. Os indivíduos com mais descendentes tem maior possibilidade de perpetuarem seu gene nas próximas gerações (MOGNON, 2004).

Segundo Medeiros (2002), existe uma ordem do processo de evolução colocada pela seleção natural, que pode ser dividida em duas partes. A primeira parte é caracterizada por obter variedade ou diversidade genética, sendo realizada aleatoriamente. Na segunda parte é feita a seleção e o controle desta diversidade, dependendo dos fatores ecológicos do ambiente. Pela seleção natural, a conservação do gene aumenta conforme este apresenta características promissoras com respeito ao ambiente. A justificativa para tal aumento é que as características genéticas são transmitidas por meio da reprodução, à medida que os indivíduos com maior aptidão possuem tais genes e maior tendência a reprodução e sobrevivência. Um gene promissor pode atribuir ao indivíduo as seguintes vantagens:

- Maior tempo de sobrevivência, levando em conta o número de reproduções do indivíduo;

- Maior fertilidade, mesmo não sobrevivendo muito, a taxa de reprodução ainda pode aumentar, resultando em maior número de filhos que outros indivíduos.

Holland (1975) iniciou a definição das bases de otimização com conceito genético. Introduzindo os AGs como técnica de otimização por meio de simulações de sistemas genéticos. David H. Goldberg, ex-aluno de Holland, desenvolveu essa metodologia mais detalhadamente (GOLDBERG; HOLLAND, 1988).

AGs são métodos com origem em eventos aleatórios cuja busca projetada para grandes espaços é realizada através da evolução de uma população de indivíduos, da qual cada indivíduo representa uma solução para o problema (GOLDBERG; HOLLAND, 1988).

Segundo Lacerda e Carvalho (1999), a otimização busca a melhor solução para um problema em questão. É necessário tentar diversas soluções e utilizar as informações resultantes para encontrar soluções cada vez melhores. As técnicas de busca e otimização, na maioria das vezes, apresentam:

- Espaço de busca: onde estão contidas todas as prováveis soluções do problema;
- Função Objetivo (ou função de aptidão): a partir dela as soluções serão avaliadas e, a cada uma, dada uma nota. Esta função é também chamada de função de aptidão.

A otimização visa achar soluções que compreenda ao ponto máximo ou mínimo. Encontrar um máximo global dentre diversos mínimos globais, dependendo da função objetivo, pode ser uma tarefa impossível utilizando métodos tradicionais, como gradiente, por exemplo. Para tais casos podemos utilizar os Algoritmos Genéticos para achar a melhor solução ou a melhor possível.

A primeira etapa gera uma população inicial de cromossomos, esta é composta por um conjunto randômico de cromossomos que representam soluções possíveis para o problema. No processo evolutivo, é dada uma nota, chamada de aptidão para a população que foi avaliada e cada cromossomo, representando a qualidade da solução. Na maioria das vezes, cromossomos com maior aptidão são escolhidos e os com menor rejeitados. Membros selecionados estão sujeitos a modificações através dos operadores, sendo eles *crossover* e mutação. Desta forma serão gerados descendentes da geração futura. Tal processo deve ser realizado até que uma solução satisfatória seja encontrada.

Holland propôs um algoritmo denominado na literatura por *Simple Generic Algorithm* ou *Standard Generic Algorithm*, também chamado de SGA. É possível descrevê-lo em seis etapas Carrijo (2004 apud DAVIS, 1991):

1. Inicializar uma população de tamanho N com valores aleatórios;
2. Avaliar a aptidão da população;
3. Criar novos cromossomos. Selecionar pais, realizar *crossover* e mutação nos respectivos pais.
4. Eliminar membros da geração anterior, inserir filhos na população, de modo a ter os mesmos N valores.
5. Avaliar a aptidão dos N indivíduos.

6. Caso o número de gerações for atingido ou a solução ideal for encontrada, deve ser retornado a melhor aptidão. Se não, voltar a etapa três.

2.2.1 Terminologia

Na biologia é mencionado na teoria da evolução que o meio ambiente seleciona os seres vivos com maior aptidão de uma determinada população para sobrevivência, isto, a cada geração. Os mais aptos conseguem fazer a reprodução e os menos são excluídos antes da reprodução. Nesta etapa acontece o *crossover* (recombinação) e a mutação que ocasiona variação do material genético contido nos cromossomos. Resultado disto é a diversidade da população. Logo os mais hábitos são os sobreviventes.(LACERDA; CARVALHO, 1999).

Como os AGs são uma metáfora desses fenômenos, é plausível que existam vários termos retirados da biologia. Os principais termos são:

- Cromossomo e Genoma: Em termos biológicos, o conjunto inteiro de genes de um organismo é denominado genoma. Um genoma é composto de vários cromossomos. Nos AGs, esses dois constituem a estrutura de dados que codificam uma solução;
- Gene: Em termos biológicos, é a unidade de hereditariedade sendo transmitida pelo cromossomo e contém as características do organismo. Nos AGs, é um elemento do vetor que corresponde o cromossomo;
- Indivíduo: Um membro da população. Nos AGs, compreende no cromossomo e na sua respectiva aptidão;
- Fenótipo: Nos AGs, representam o cromossomo decodificado.

As Subseções 2.2.2, 2.2.3, 2.2.4, 2.2.5, 2.2.6 e 2.2.7, especificam as etapas dos Algoritmos Genéticos.

2.2.2 Representação da população

Na maioria das vezes, a população inicial de indivíduos ou cromossomos é criada aleatoriamente, embora em certas situações é interessante uma seleção heurística, introduzindo indivíduos mais importantes. Entretanto, a população deve possuir uma variedade de cromossomos considerável (CASTRO, 2001).

2.2.3 Avaliação da população

Segundo Castro (2001), a avaliação da população é o retorno da função de aptidão. Deve ser especificado o custo referente a qualidade de cada indivíduo da respectiva população, que está vinculada a função objetivo que deverá ser otimizada.

2.2.4 Seleção

Lacerda e Carvalho (1999) afirmam que tendo como base a seleção natural de seres vivos, o Algoritmo Genético faz a seleção dos melhores cromossomos da população inicial. Para estes cromossomos, cuja aptidão é alta, são os gerados dos cromossomos filhos. Isto foi possível graças aos processos de *crossover* e mutação. A probabilidade dos pais serem selecionados é proporcional a sua aptidão. Em termos matemáticos:

$$p_i = \frac{f_i}{\sum_1^n f_i} \quad (2.1)$$

p_i = Probabilidade do cromossomo i ser escolhido

f_i = aptidão do cromossomo

n = Número total de indivíduos/cromossomos da população

$\sum_1^n f_i$ = somatório de todas as aptidões

Roda da roleta

É possível fazer a seleção pelo seguinte procedimento prático: Fazer o cálculo de uma coluna de aptidões acumuladas. Deve ser gerado um número aleatório r entre 0 e a soma da maior aptidão acumulada. O cromossomo selecionado será aquele cuja aptidão acumulada for primeiramente maior ou igual ao número r .

Seleção por torneio

Para o caso de aptidões negativas, o algoritmo roda da roleta não funcionará. Ademais, é possível uma convergência prematura no que se refere ao máximo ou mínimo da função, pois este algoritmo pega o maior mais próximo. Com a seleção por torneio são escolhidos n cromossomos da população, aleatoriamente e com probabilidades iguais. O cromossomo com melhor aptidão é o escolhido.

Aleatória

Dada uma probabilidade p , são escolhidos n cromossomos da população, aleatoriamente.

2.2.5 Crossover

Lacerda e Carvalho (1999) falam sobre a questão do *crossover* ser um dos principais operadores com respeito aos mecanismos de busca dos AGs para explorar regiões ainda não conhecidas do escopo de busca. O *crossover* é aplicado a um par de cromossomos oriundos da população intermediária, produzindo dois filhos. Um dos operadores clássicos de *crossover* é o operador de ponto de corte variável, no qual os cromossomos pais têm seus genes cortados em um ponto aleatório, gerando duas cabeças e duas caudas. A partir disto são trocadas as caudas, produzindo dois cromossomos novos.

A aplicação do *crossover* é realizada a partir de uma dada probabilidade. Esta probabilidade pode ser chamada de taxa de *crossover*, que compreende entre 60% e 90% (LACERDA; CARVALHO, 1999). Se não acontecer *crossover*, os filhos serão iguais aos pais, permitindo que certas soluções sejam preservadas. Uma forma de implementar esse procedimento é gerando números aleatórios no intervalo $[0,1]$. Sendo assim, o *crossover* é aplicado caso o número aleatório seja menor que sua taxa.

2.2.6 Mutação

Para fazer a manutenção da diversidade genética na população é necessário o uso do operador mutação, que altera de forma arbitrária os genes de um cromossomo escolhido. Desta forma é possível alocar novos elementos na população. A mutação é aplicada com uma probabilidade denominada de taxa de mutação. De maneira geral é utilizada uma taxa de mutação pequena, assim como acontece na genética natural, devido ser um operador genético secundário (CASTRO, 2001).

2.2.7 Chaves aleatórias - RKGA

Carvalho (2013) afirma que existem dois aspectos fundamentais ao aplicar um Algoritmo Genético: A forma de funcionamento dos operadores e como as soluções são codificadas. Bean (1994) introduziu o conceito de chaves aleatórias (*random key genetic algorithms* - RKGA) dos algoritmos genéticos. Neste tipo de codificação pode ser assumido que os vetores são compostos por indivíduos, de tamanho y . Cada gene, ou elemento, equivale a um valor aleatório compreendido entre $[0,1]$. O conjunto de soluções é analisado de forma indireta, pois estas são percorridas no conjunto das chaves aleatórias relacionando o decodificador para que possa gerar a aptidão de cada cromossomo, levando em consideração a relevância das chaves geradas. O algoritmo do decodificador deve ser determinístico, ou seja, dado um vetor de chaves aleatórias sempre terá uma única solução (CARVALHO, 2013).

Resende (2011) define um decodificador como um procedimento que faz um mapeamento do vetor de chaves aleatórias numa solução do problema em questão e gera o custo desta solução. A cada geração gerada a elite é definida. Este conceito caracteriza o RKGA. A seguir vetores de chaves aleatórias são introduzidos na geração. Estes vetores são denominados de mutantes e têm o mesmo objetivo dos operadores de mutação dos AGs clássicos, ou seja, impedir que a população de uma determinada geração convirja para um ótimo local e não global.

Capítulo 3

Metodologia Proposta

Neste capítulo, o método proposto é descrito na Seção 2.2, o qual é baseado na metaheurística Algoritmos Genéticos (AG) associados ao conceito de chaves randômicas, ou como Bean (1994) propôs, *Random Key Genetic Algorithms* (RKGA).

O AG proposto, detalhado na Seção 3.2, aborda em conjunto as três fases citadas no Capítulo 2, Seção 2.1, e tem como premissa poços já alocados em determinadas posições, para que seja definido a posição dos *manifolds* e das UEPs. Antes, porém, são apresentados os paradigmas de programação (Seção 3.1).

3.1 Paradigmas de programação

A escolha da linguagem tem como condição necessária a velocidade. A linguagem escolhida foi C/C++. Além da velocidade, esta é muito utilizada em implementações envolvendo otimização em geral.

É utilizado o paradigma de orientação a objetos. A implementação é dividida em classes, sendo elas, CProblema, CSolucao e CGenetic. O programa gerador é utilizado para criar diversas estruturas de conteúdo randômico ou não, será explicado no capítulo 4. A classe CProblema é responsável por instanciar e guardar nas estruturas de dados a entrada gerada pelo programa gerador. CSolucao tem como objetivo a criação e a avaliação da população. CGenetic tem a estrutura do RKGA e seus operadores genéticos, cuja função é instanciar e executar o RKGA.

A classe CProblema é responsável por receber as entradas fornecidas pelo programa gerador. Este procedimento é executado através do método *LerDados*, inicializando as matrizes de distâncias, os vetores de capacidade, as posições possíveis (com exceção dos poços) e as quantidades envolvidas.

A classe CSolucao utiliza os dados da classe CProblema para criar uma solução inicial pelo método *CriarSolucaoInicial*. Além disso, realiza a avaliação da população em questão pelo método *AvaliarSolucao*. A decodificação, usando o conceito de RKGA, é feita pelos métodos *AvaliarPocoManifold* e *AvaliarManifoldUEP*.

Os métodos da classe CGenetic seguem a estrutura dos RKGA. As etapas mencionadas no Capítulo 2, Seção 2.1, estão presentes nesta classe.

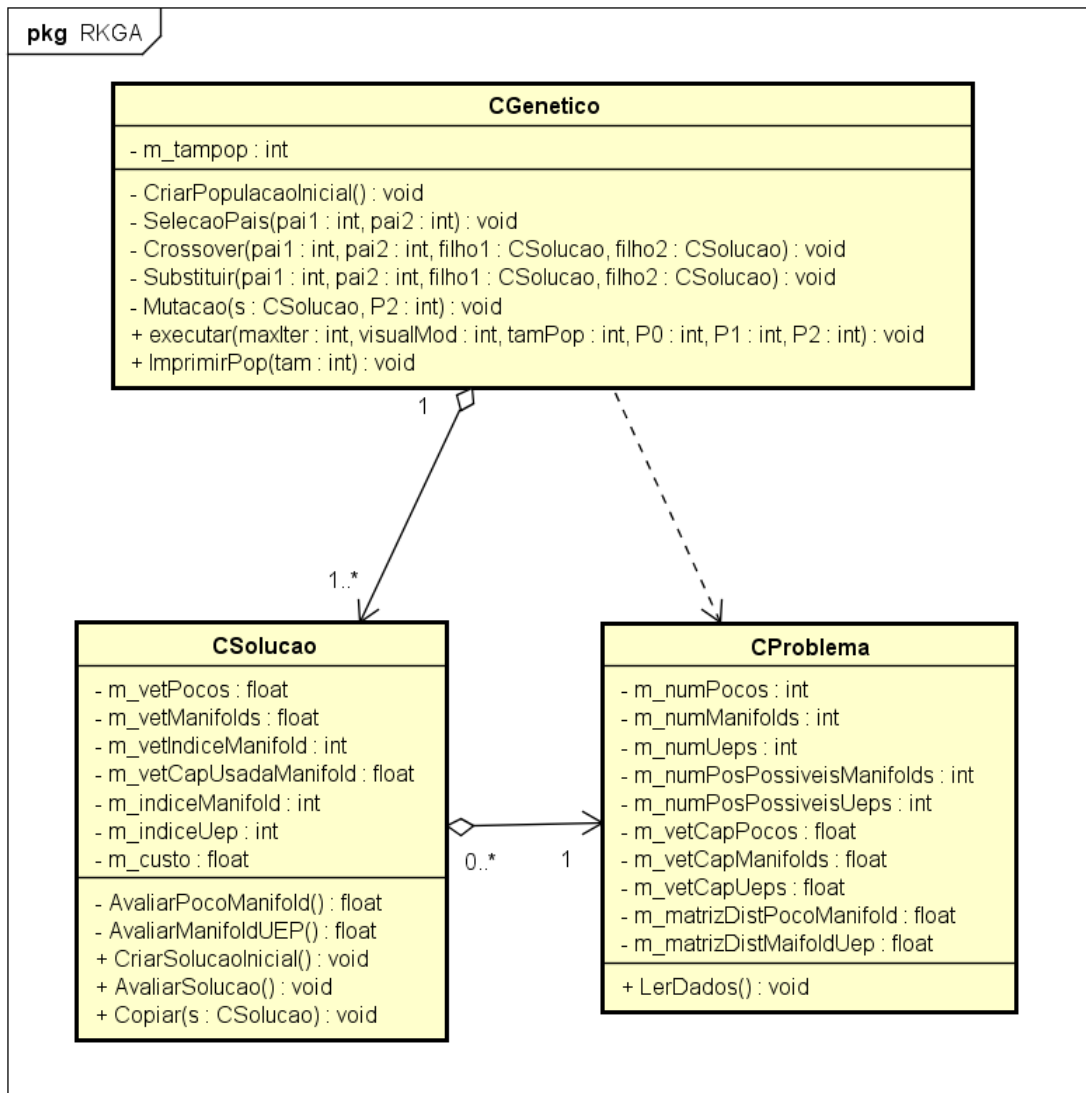


Figura 3.1: Diagrama de Classes.

3.2 Algoritmo genético para o problema apresentado

3.2.1 Visão geral

Seja um conjunto de p poços, dos quais são associados a m_1 manifolds. Para alocar os m_1 manifolds, existem m_2 posições disponíveis, tal que ($m_2 > m_1$). Após alocados os m_1 manifolds, estes devem ser associados a um conjunto de u_1 UEPs. Para alocar as u_1 UEPs, existem u_2 posições disponíveis, tal que ($u_2 > u_1$).

No Algoritmo 1 é apresentado o pseudocódigo do RKGA proposto. Os procedimentos deste algoritmo serão relatados nas seções seguintes. Na Seção 3.3.2, está definida a população inicial, como indicado na Linha 1. É definido na Seção 3.3.3, a avaliação e decodificação (Linhas 2,7,10,19 e 22). Na Seção 3.3.4 é descrita a seleção (Linha 4). O *crossover* está definido em 3.3.5 e na Linha 6 do algoritmo em questão. A mutação na Seção 3.3.6 ocorre nas Linhas 16 e 17.

Algorithm 1 RKGA

```

1: matPop ← criarPopulacaoInicial()           ▷ matPop recebe a população
2: melhor ← avaliacao(matPop)                 ▷ melhor recebe o indivíduo de melhor aptidão da população
3: para iter ← 1 até maxGeracao faça
4:   selecao(pai1,pai2)                       ▷ Realizando a seleção
5:   se  $P_0$  então                           ▷ Com probabilidade  $P_0$  de realizar crossover
6:     matPop ← crossover(pai1,pai2,filho1,filho2)
7:     se avaliaCromossomo(filho1) < melhor então
8:       melhor ← avaliaCromossomo(filho1)     ▷ Avaliação e decodificação
9:     fim se
10:    se avaliaCromossomo(filho2) < melhor então
11:      melhor ← avaliaCromossomo(filho2)     ▷ Avaliação e decodificação
12:    fim se
13:    substituir(pai1,pai2,filho1,filho2)
14:  fim se
15:  se  $P_1$  então                             ▷ Com probabilidade  $P_1$  de realizar mutação
16:    matPop ← mutacao(pai1, $P_2$ )
17:    matpop ← mutacao(pai2, $P_2$ )
18:    se avaliaCromossomo(pai1) < melhor então
19:      melhor ← avaliaCromossomo(pai1)
20:    fim se
21:    se avaliaCromossomo(pai2) < melhor então
22:      melhor ← avaliaCromossomo(pai2)
23:    fim se
24:  fim se
25: fim para
26: Escrever(melhor)

```

3.2.2 População inicial

A população de poços e *manifolds*, contém dentro de cada respectivo gene, um número aleatório contido no intervalo $[0,1]$. Desta forma os cromossomos são gerados a partir de diversas chaves aleatórias.

A quantidade de cromossomos é um dado de entrada. Cada cromossomo tem tamanho igual ao número de poços somado ao número de *manifolds*.

3.2.3 Avaliação da população

Seguindo o padrão dos RKGA, o valor de cada gene compreende a um número aleatório gerado no intervalo que é dado como entrada no algoritmo. Contudo, este intervalo pode ter um valor máximo e mínimo diferente de $[0,1]$, conforme a contextualização do sistema marítimo de produção. Desta forma,

o conjunto de chaves aleatórias é definido.

Decodificador

Enquanto não for atingindo o número total de poços, é realizada uma busca no cromossomo da geração corrente. Essa busca procura a maior chave aleatória no vetor correspondente ao indivíduo da população, e será guardado o índice k deste poço.

Para saber aonde alocar o poço p_k a um *manifold*, é preciso definir dois conjuntos: $\alpha = \{x \mid x \text{ são } manifolds \text{ usados}\}$; $\beta = \{y \mid y \text{ são os } manifolds \text{ não usados}\}$. Se α é diferente de vazio, então o *manifold* mais próximo e disponível pertencente a α , será ligado ao poço em questão. Para o próximo poço p_{k+1} , tal passo se repete até que todos os *manifolds* de α sejam analisados. Se nenhum *manifold* estiver disponível, a busca apontará para β , com o objetivo de achar o *manifold* mais próximo e disponível.

O exemplo a seguir ilustra o procedimento de codificação.

Cromossomo	0.3	0.7	0.6	0.8	0.2	Capacidade poços	30	40	50	45	35	
	1	2	3	4	5		1	2	3	4	5	
Posições utilizadas por manifolds	0	0	0	0	0	Capacidade manifolds	70	50	90	80	30	
	1	2	3	4	5		1	2	3	4	5	
Distância Poço x Manifold	55	65	90	59	41	Capacidade Usada manifolds	0	0	0	0	0	
	56	54	35	77	31			1	2	3	4	5
	73	38	31	88	59		Poços alocados	0	0	0	0	0
	90	58	46	49	76			1	2	3	4	5
	65	37	38	35	46		Índices manifolds	0	0	0	0	0
Manifolds Usados	0						1	2	3	4	5	
Custo com jumpers ou flowlines	0											

Figura 3.2: RKGA antes da iteração 1.

Antes da primeira iteração não há *manifold* usado e poço alocado. Logo, $\alpha = \beta = \emptyset$. Para este exemplo tem-se 4 *manifolds* disponíveis e foram gerados os seguintes vetores:

- Cromossomo: Contendo valores aleatórios entre $[0,1]$;
- Posições utilizadas por *manifolds*: indicando dentre as 5 posições possíveis, qual o *manifold* usado;
- Capacidade poços: Com números gerados entre 30 e 50 de maneira aleatória;
- Capacidade *manifolds*: Com números gerados entre 30 e 90, indica a capacidade de cada *manifold*;
- Capacidade usada *manifolds*: Indicando o quão cheio está determinado *manifolds*;
- Poços alocados: Indica os poços usados;
- índices *manifolds*: Será alocado de maneira sequencial os índices dos *manifolds* usados.

Foi gerada uma matriz de distâncias entre poços e o número de possíveis posições de *manifolds*, com valores entre 30 e 90. Outras duas variáveis são os *manifolds* usados e custo com *jumpers* e *flowlines*.

Cromossomo	0.3	0.7	0.6	0.8	0.2	Capacidade poços	30	40	50	45	35	
	1	2	3	4	5		1	2	3	4	5	
Posições utilizadas por manifolds	0	0	1	0	0	Capacidade manifolds	70	50	90	80	30	
	1	2	3	4	5		1	2	3	4	5	
Distância Poço x Manifold	55	65	90	59	41	Capacidade usada manifolds	0	0	45	0	0	
	56	54	35	77	31			1	2	3	4	5
	73	38	31	88	59		Poços alocados	0	0	0	1	0
	90	58	46	49	76			1	2	3	4	5
	65	37	38	35	46			1	2	3	4	5
Manifolds Usados						1						
Custo com jumpers ou flowlines						45						
						Índices manifolds	3	0	0	0	0	
							1	2	3	4	5	

Figura 3.3: RKGA iteração 1.

Na iteração 1 foi escolhido o gene com maior chave aleatória e guardado seu índice. Na matriz de distâncias, na linha 4, o menor elemento é igual a 46. Este elemento está na coluna 3. A seguir é verificado que a capacidade do poço 4, tem valor 45. Como o valor da capacidade do poço somado com o valor da capacidade usada do *manifold* é menor que a capacidade do *manifold* corrente, o vetor de capacidades usadas recebe 45 na posição 3. No índice 4 do vetor de poços alocados é atribuído o valor 1, indicando o uso do poço p_1 . Para o vetor de posições utilizadas por *manifolds*, no índice 3, deverá ser atribuído o valor 1 para indicar que o *manifold* m_3 será usado. Até o momento 1 *manifold* foi usado e o custo com cabeamentos chega a 45.

Cromossomo	0.3	0.7	0.6	0.8	0.2	Capacidade poços	30	40	50	45	35	
	1	2	3	4	5		1	2	3	4	5	
Posições utilizadas por manifolds	0	0	1	0	0	Capacidade manifolds	70	50	90	80	30	
	1	2	3	4	5		1	2	3	4	5	
Distância Poço x Manifold	55	65	90	59	41	Capacidade usada manifolds	0	0	85	0	0	
	56	54	35	77	31			1	2	3	4	5
	73	38	31	88	59		Poços alocados	0	1	0	1	0
	90	58	46	49	76			1	2	3	4	5
	65	37	38	35	46			1	2	3	4	5
Manifolds Usados						1						
Custo com jumpers ou flowlines						85						
						Índices manifolds	3	3	0	0	0	
							1	2	3	4	5	

Figura 3.4: RKGA iteração 2.

Na iteração 2, o segundo maior gene é escolhido com base na sua segunda maior chave aleatória. Na linha 2 da matriz de distâncias, o valor pertencente a coluna 3 indica a distância entre o único *manifold* anteriormente escolhido e o poço 2. O índice desta coluna representa o *manifold* já escolhido. Dessa forma $\beta = \{1\}$. Agora será verificado se a capacidade do poço p_2 somada a capacidade usada pelo *manifold* m_3 , não extrapola a capacidade total de m_3 . A verificação teve resultado positivo, devido a isto, será atribuído mais 40 para a capacidade usada pelo manifold m_3 . Em poços alocados será atribuído 1 na posição 2. Índices *manifolds* na segunda posição, também recebem o índice do *manifold* usado. Na posição 3 do vetor posições utilizadas por *manifolds*, o valor 1 continua marcado e os demais zerados. Com isto, o custo com *jumpers* e *flowlines* é igual a 85.

Cromossomo	0.3	0.7	0.6	0.8	0.2	Capacidade poços	30	40	50	45	35
	1	2	3	4	5		1	2	3	4	5
Posições utilizadas por manifolds	1	1	1	1	0	Capacidade manifolds	70	50	90	80	30
	1	2	3	4	5		1	2	3	4	5
Distância Poço x Manifold	55	65	90	59	41	Capacidade usada manifolds	55	35	85	50	0
	56	54	35	77	31		1	2	3	4	5
	73	38	31	88	59	Poços alocados	1	1	1	1	1
	90	58	46	49	76		1	2	3	4	5
	65	37	38	35	46	Índices manifolds	3	3	4	1	2
Manifolds Usados							1	2	3	4	5
Custo com jumpers ou flowlines											
							225				

Figura 3.5: RKGA iteração final.

A iteração final mostra o resultado. Todos os poços foram alocados e 1 *manifold* não precisou ser ligado, logo, $\alpha = \{1,2,3,4,5\}$ e $\beta = \{1,2,3,4\}$. O custo total com *jumpers* ou *flowlines* para esse indivíduo da população é igual a 225.

Para *manifolds* e UEPs os procedimentos anteriores são realizados de maneira análoga.

Avaliação total

Com os resultados da codificação do cromossomo, é feita a soma dos valores. Esta soma é a representação numérica da aptidão do indivíduo, ap_j , tal que j é o número em comum que representa o total de indivíduos. Para a geração corrente g_i , este processo é realizado para todos os indivíduos da população.

Depois da criação da população inicial, o indivíduo com melhor aptidão $melhor_i$ é separado e esse valor, guardado. Para a próxima geração é realizado o mesmo processo, no entanto, o melhor indivíduo $melhor_i$ da geração g_i , tem sua aptidão comparada com o melhor indivíduo, $melhor_{i+1}$, da geração seguinte g_{i+1} . O melhor indivíduo permanece na geração corrente.

3.2.4 Seleção

A forma de seleção escolhida para o primeiro pai é a seleção por torneio, conforme visto na Seção 2.2.4. São separados dois indivíduos da população, aleatoriamente e com probabilidades equivalentes. Estes dois serão avaliados e o melhor indivíduo será selecionado. Já o segundo pai é escolhido aleatoriamente para permitir que todos indivíduos tenham chance de serem selecionados como pais.

3.2.5 Crossover

Para a efetuar o *crossover*, adota-se o algoritmo de ponto de corte variável para o cromossomo em questão. Este ponto é obtido através de um número aleatório, diferente de zero e não negativo, compreendendo no máximo ao número de poços ou *manifolds*.

Feito o corte nos cromossomos nos pais da população em questão, são criados dois novos filhos. O primeiro filho constituído da primeira parte do primeiro pai, com a segunda parte do segundo pai. Outro filho originado da segunda parte do primeiro pai, com a primeira parte do segundo pai.

Seguidamente, acontece a seleção desses quatro cromossomos. Todos são avaliados e os dois melhores permanecem na população.

A ocorrência do *crossover* é dada a partir da sua taxa, que já está pré-definida. Além disso, é importante ressaltar que são realizados dois *crossovers*, ao passo que um é seguido do outro, ou seja, primeiramente o *crossover* de poços e em seguida para *manifolds*.

				Ponto de corte	
Pai 1	0.3	0.7	0.6	0.8	0.2
Pai 2	0.4	0.1	0.9	0.5	0.6
Filho 1	0.3	0.7	0.6	0.5	0.6
Filho 2	0.4	0.1	0.9	0.8	0.2

Figura 3.6: Ponto de corte variável.

Na Figura 3.6, um ponto de corte foi escolhido aleatoriamente. O Filho 1, recebe a primeira parte do Pai 1 e a segunda parte do Pai 2. O Filho 2, recebe a primeira parte do Pai 2 e a segunda parte do Pai 1.

3.2.6 Mutação

Através de uma determinada taxa de mutação, é realizada a mutação. Esta acontece apenas para os cromossomos que passaram por *crossover*.

Realizado tal procedimento, os genes escolhidos são sobrescritos por um número aleatório entre o valor máximo e mínimo dos genes da população.

Finalizado este processo, a população é avaliada.

Capítulo 4

Experimentos Realizados

4.1 Ambiente de desenvolvimento

O presente trabalho teve seus experimentos realizados no notebook Dell Vostro 3550, com processador Intel(R) Core(TM) i5-2410M CPU @2.30GHz. Tal computador, possui 8GB de memória RAM e HD com capacidade de 500GB. Tem como sistema operacional *Windows 8.1 Professional* de 64 bits. O *Code::Blocks* foi a IDE adotada. O compilador adotado é o GNU GCC *compiler*.

4.2 Gerador

O programa gerador tem a função de receber as quantidades de poços, *manifolds* e UEPs, os valores de capacidade máxima e capacidade mínima dos poços, distâncias máximas e mínimas entre poços e *manifolds* e entre *manifolds* e UEPs, o número de possíveis posições de *manifolds* e UEPs.

O arquivo de saída obedece a seguinte ordem:

1. Número de poços;
2. Vetor de capacidades dos poços;
3. Número de *manifolds*;
4. Número de possíveis posições dos *manifolds*;
5. Vetor de capacidades dos *manifolds*;
6. Número das UEPs;
7. Número de possíveis posições das UEPs;
8. Vetor de capacidades das UEPs;
9. Matriz de distâncias de tamanho número de poços por número de possíveis posições dos *manifolds*;
10. Matriz de distâncias de tamanho número de possíveis posições dos *manifolds*, pelo número de possíveis posições das UEPs.

A capacidade dos *manifolds*, é gerada a partir de um valor de folga que representa o quanto por cento esses valores são maiores que os valores das capacidades dos poços. De maneira análoga, é realizado o mesmo procedimento para *manifolds* e UEPs.

Ambas as matrizes são geradas através de valores aleatórios entre $[0,1]$. A diversidade desses valores está relacionada com os valores de distâncias de máximo e mínimo, de *manifolds* e UEPs.

4.3 Resultados

Para os experimentos, foram geradas cinco instâncias, as quais variam de acordo com número de poços, *manifolds* e UEPs. A Tabela 4.1 reúne as informações das cinco entradas, sendo os números de poços, *manifolds*, possíveis *manifolds*, UEPs e possíveis UEPs. Inicialmente, o RKGA proposto foi executado, variando o número máximo de gerações. Cada instância da Tabela 4.1 foi executada cinco vezes, tendo seu tempo médio (em segundos) e valor de aptidão médio anotados. As Tabelas 4.2, 4.3, 4.4, 4.5 e 4.6 descrevem os resultados.

Cada instância tem um domínio com tamanhos diferentes, sendo a Entrada1.txt a menor instância e a Entrada5.txt, a maior. Cada tabela de resultados demonstra o comportamento do RKGA para diferentes gerações.

Tabela 4.1: Instâncias e entradas

Instâncias	Nº Poços	Nº <i>Manifolds</i>	Nº Possíveis <i>Manifolds</i>	Nº UEPs	Nº Possíveis UEPs
Entrada1.txt	10	5	10	2	5
Entrada2.txt	20	10	15	5	10
Entrada3.txt	35	15	20	8	15
Entrada4.txt	50	20	30	10	15
Entrada5.txt	100	20	30	10	15

Tabela 4.2: Entrada1.txt.

Gerações	100	500	1000	5000	10000
Avaliação Média	8778,2	8561	8405,6	8298,2	8298,2
Tempo Médio (s)	0,041	0,091	0,153	0,686	1,3678

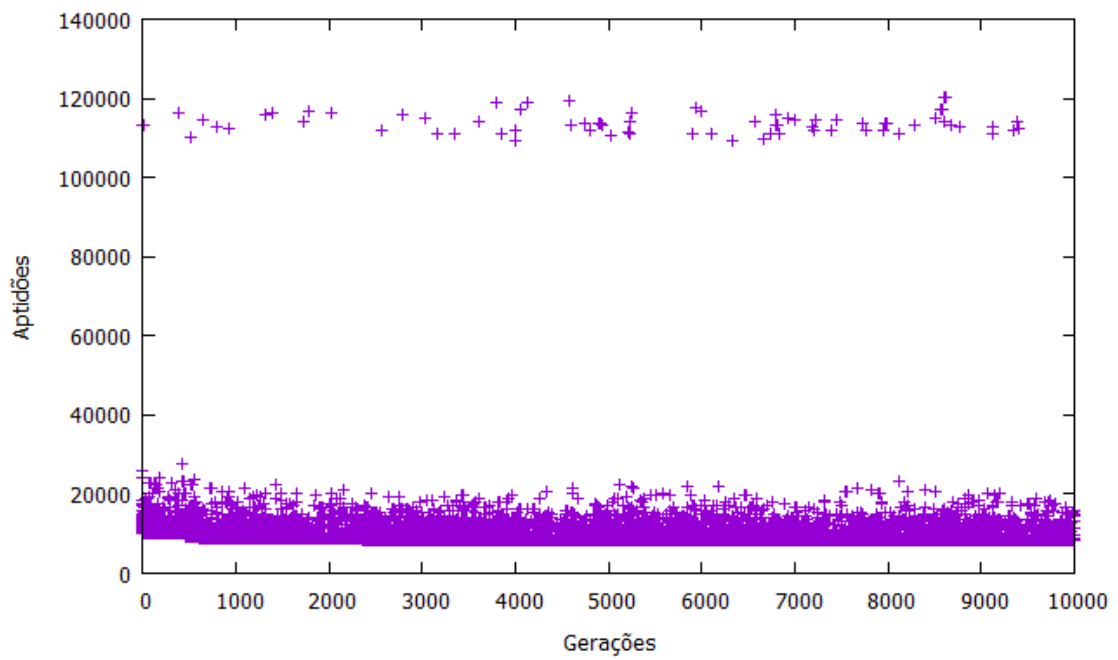


Figura 4.1: Representação da população da instância Entrada1.txt.

Tabela 4.3: Entrada2.txt.

Gerações	100	500	1000	5000	10000
Avaliação Média	18760,0	16290,0	15331,8	14110,4	13805,6
Tempo Médio (s)	0,062	0,151	0,279	1,307	2,477

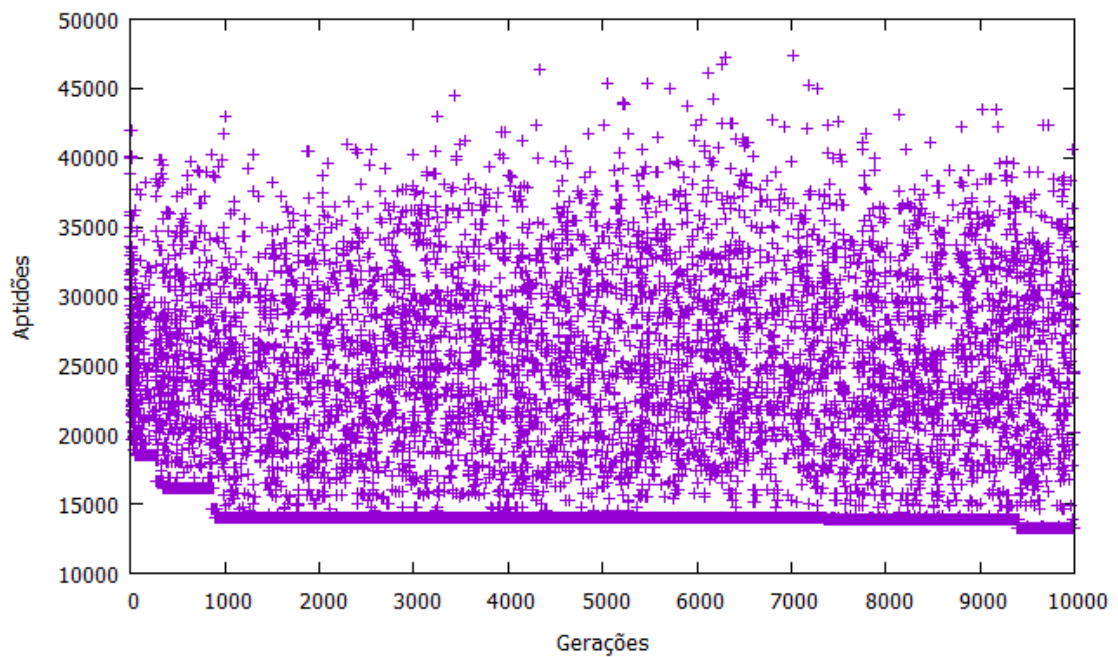


Figura 4.2: Representação da população da instância Entrada2.txt.

Tabela 4.4: Entrada3.txt.

Gerações	100	500	1000	5000	10000
Avaliação Média	29407,6	24831,4	24029,6	21481,6	20710,2
Tempo Médio (s)	0,081	0,271	0,505	2,481	4,848

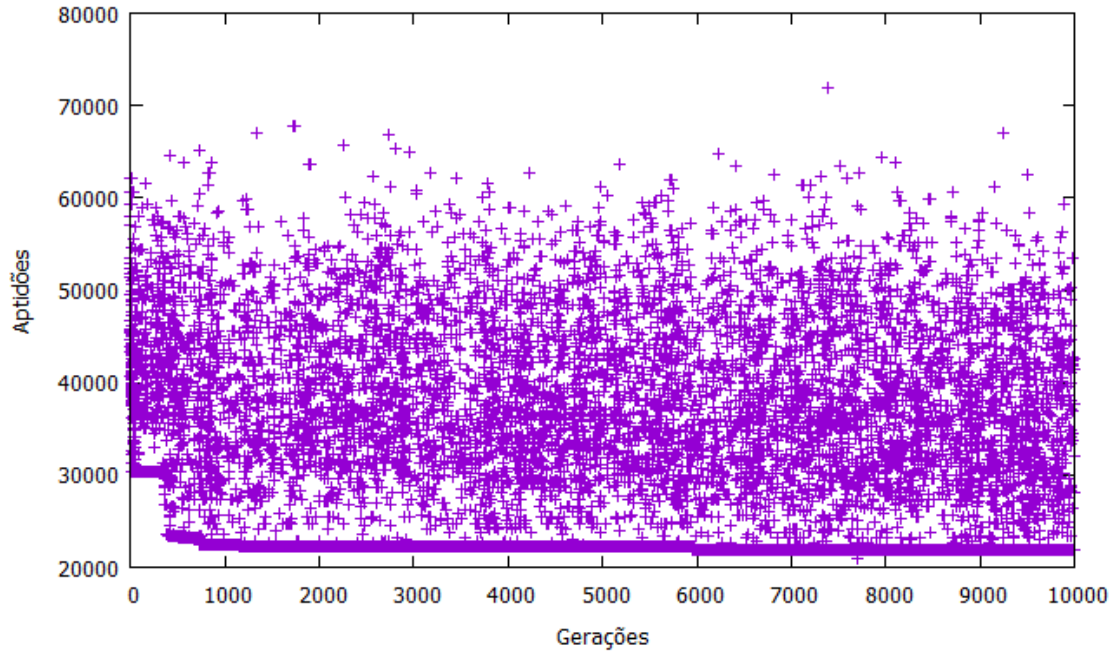


Figura 4.3: Representação da população da instância Entrada3.txt.

Tabela 4.5: Entrada4.txt.

Gerações	100	500	1000	5000	10000
Avaliação Média	42416,6	37838,8	36679,6	30533,0	29050,8
Tempo Médio (s)	0,139	0,4318	0,911	4,116	8,135

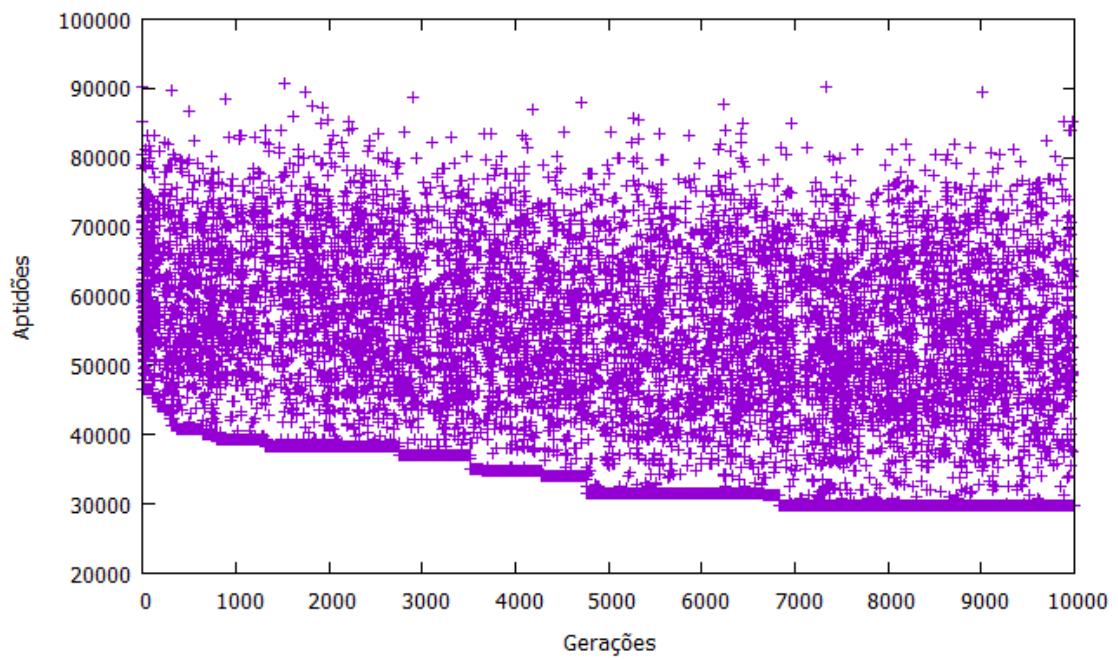


Figura 4.4: Representação da população da instância Entrada4.txt.

Tabela 4.6: Entrada5.txt.

Gerações	100	500	1000	5000	10000
Avaliação Média	101568,6	91390,6	89207,8	80871,0	78686,0
Tempo Médio (s)	0,238	1,0638	1,99	9,690	19,225

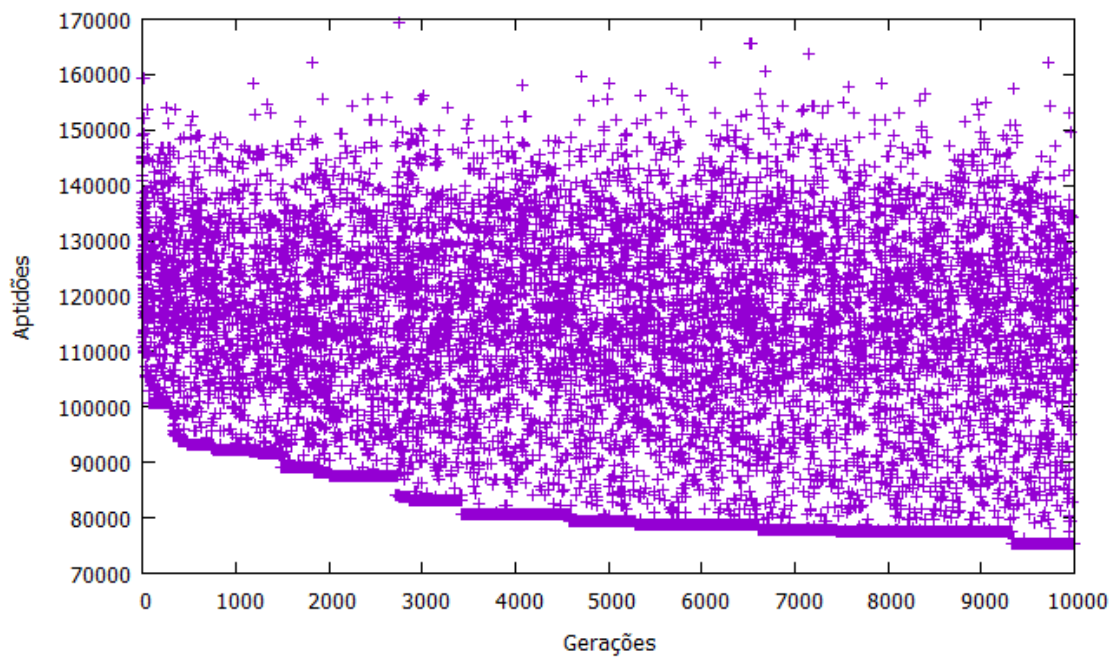


Figura 4.5: Representação da população da instância Entrada5.txt.

A Entrada1.txt, por ser uma instância menor, convergiu rapidamente. Pela Tabela 4.2 nota-se que a avaliação média entre as gerações 5000 e 10000 não variou. Os melhores indivíduos tem aptidões entre 8298,2 e 8778,2 aproximadamente.

A Entrada2.txt tem o dobro da quantidade de poços, *manifolds* e UEPs, comparado a Entrada1.txt. Os melhores indivíduos então entre aproximadamente 12805,6 e 1760,0. É possível notar que a diversidade da população foi mantida.

Na Entrada3.txt nota-se que a diversidade da população foi mantida. Entre as gerações de 5000 e 10000 houve melhora na avaliação média e maior variação de tempo médio. Os melhores indivíduos estão na faixa de 20710,2 e 29407,6.

Em comparação com a Entrada1.txt, a Entrada4.txt tem o quádruplo do tamanho. A diversidade da população foi mantida. Os melhores indivíduos variam entre 29050,8 e 42416,6.

Analisando a Entrada5.txt para 10000 gerações nota-se que o resultado melhorou conforme cada geração foi passando, tendo em vista que o objetivo é minimizar as aptidões, sendo estas equivalentes aos custos com *jumpers*, *flowlines* e *risers*. Ademais, o tempo de processamento teve um aumento de 50,4% referente as gerações de 5000 e 10000 para a Entrada5.txt. Com respeito ao resultado dessa mesma entrada, a melhora da geração 100 para a 10000 é de aproximadamente 25,5%, uma boa otimização, levando em conta em apenas aproximados 19 segundos de diferença. Os melhores indivíduos estão entre aproximadamente 78686,0 e 101568,6. Assim como nas outras instâncias, a diversidade da população foi mantida.

A construção dos gráficos levam em conta a avaliação de todos os indivíduos da população para a geração em questão da instância corrente. Desta forma, cada geração tem a quantidade de avaliações igual ao tamanho da população. Um ponto é definido obtendo o custo de um indivíduo e a geração atual do mesmo.

Pelas Figuras 4.1, 4.2, 4.3, 4.4 e 4.5, é perceptível a evolução da população ao longo das 10000 gerações. A diversidade da população é mantida ao longo das gerações. Para todas instâncias, é possível visualizar e comprovar que a melhora nos resultados está diretamente proporcional a quantidade de gerações, ou seja, o indivíduo mais apto permanece.

Capítulo 5

Considerações Finais

5.1 Conclusões

O presente trabalho apresenta uma solução para o problema de locação de UEPs e *manifolds*. Para minimizar as distâncias e custos e maximizar a produção, foi colocado em prática a metodologia proposta envolvendo RKGAs.

Em contribuição aos RKGAs, o uso de uma nova técnica foi empregado. Os cromossomos criados possuem duas características distintas: a primeira parte representa os poços; e a segunda representa os *manifolds*. A função de aptidão leva em conta as demandas atendidas e as capacidades envolvidas. Além destes fatores é relevante os custos relacionados aos cabeamentos de *risers*, *flowlines* e *jumpers*.

Os experimentos realizados para as cinco instâncias mostram a melhora nos resultados conforme as gerações são passadas. O programa que foi desenvolvido poderá ser usado na fase de planejamento dos SSPs, no entanto, diversas variáveis ainda deverão ser adicionadas dependendo do sistema submarino de produção em questão. Para o corrente problema de localização, não existe um número consideravelmente grande de soluções computacionais na literatura.

Finalmente, este trabalho contribui para o conjunto de possíveis respostas e em carga literária, para o problema de localização de plataformas de petróleo.

5.2 Sugestões para trabalhos futuros

Com base na pesquisa desenvolvida, diversas possibilidades de trabalhos futuros podem ser identificadas. Faltam ser avaliadas outras questões para que se possa ter uma melhor contextualização do problema. Exemplo disto é o custo de aquisição da UEP e seus respectivos *manifolds* envolvidos. Outro fator relevante é a ancoragem da plataforma e o escoamento, podendo ser inseridos como novas entradas.

O código poderá ser revisto considerando os novos dados. Consequência disto, é a maior abrangência do problema em questão.

Desta forma, novas pesquisas serão realizadas considerando as novas variáveis. Assim, será possível a continuidade em ambientes de pesquisa mais específicos como pós-graduação, mestrado e doutorado.

Referências Bibliográficas

- BEAN, J. C. Genetic algorithms and random keys for sequencing and optimization. *ORSA journal on computing*, INFORMS, v. 6, n. 2, p. 154–160, 1994.
- CARRIJO, I. B. *Extração de regras operacionais ótimas de sistemas de distribuição de água através de algoritmos genéticos multiobjetivo e aprendizado de máquina*. Tese (Doutorado) — Universidade de São Paulo, 2004.
- CARVALHO, A. S. F. d. *Algoritmo genético para o problema de localização de sensores*. Tese (Doutorado), 2013.
- CASTRO, R. E. de. *Otimização de estruturas com multi-objetivos via algoritmos genéticos*. Tese (Doutorado) — Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil, 2001.
- EGLER, C. A. G.; RIO, G. A. P. D. Territórios do petróleo no brasil: redes globais e governança local. *Atas do Colóquio Internacional Les intégrations régionales: quelles dynamiques transfrontalières et transnationales? Les enseignements du Bassin de la Plata dans le Mercosur*, 2004.
- FERRARI, M. *Análise da aplicação de um sistema de bombeamento multifásico submarino na produção de petróleo*. Tese (Doutorado), 2003.
- FRANCO, K. *Desenvolvimento de um Sistema Inteligente para Auxiliar a Escolha de Sistema para Produção no Mar*. Campinas: Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas. Dissertação (Mestrado) — Universidade Estadual de Campinas, 2003.
- FURTADO, A. T.; FREITAS, A. G. Nacionalismo e aprendizagem no programa de águas profundas da petrobras. *Revista Brasileira de Inovação*, v. 3, n. 1 jan/jun, p. 55–86, 2009.
- GOLDBERG, D. E.; HOLLAND, J. H. Genetic algorithms and machine learning. *Machine learning*, Springer, v. 3, n. 2, p. 95–99, 1988.
- HOLLAND, J. H. Adaptation in natural and artificial systems. an introductory analysis with application to biology, control, and artificial intelligence. *Ann Arbor, MI: University of Michigan Press*, 1975.
- LACERDA, E. G. de; CARVALHO, A. de. Introdução aos algoritmos genéticos. *Sistemas inteligentes: aplicações a recursos hídricos e ciências ambientais*, v. 1, p. 99–148, 1999.
- MEDEIROS, F. L. L. Algoritmo genético híbrido como um método de busca de estados estacionários de sistemas dinâmicos. 2002.
- MOGNON, V. R. Algoritmos genéticos aplicados na otimização de antenas. 2004.
- MORAIS, J. M. d. Petróleo em águas profundas: uma história tecnológica da petrobras na exploração e produção offshore. Instituto de Pesquisa Econômica Aplicada (Ipea), 2013.
- PETROBRAS. Disponível em: <<http://www.petrobras.com.br>>. acesso em: 27 jul. 2017.
- RESENDE, M. Introdução aos algoritmos genéticos de chaves aleatórias viciadas. 2011.
- THAMY, P. *Estudo de alternativas de arranjos submarinos de produção com o uso de manifolds e bombas multifásicas: otimização da vazão e análise financeira*. Tese (Doutorado) — Universidade Federal do Rio de Janeiro, 2015.
- THOMAS, J. E. *Fundamentos da Engenharia de Petróleo*. Rio de Janeiro: Editora Interciência Ltda, 2001.